Festschrift

LNCS 7160

Michael J. Dinneen
Bakhadyr Khoussainov
André Nies (Eds.)

Computation, Physics and Beyond

International Workshop on Theoretical Computer Science, WTCS 2012
Dedicated to Cristian S. Calude on the Occasion of His 60th Birthday
Auckland, New Zealand, February 2012, Revised Selected and Invited Papers

Springer

Lecture Notes in Computer Science 7160

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Michael J. Dinneen Bakhadyr Khoussainov
André Nies (Eds.)

Computation, Physics and Beyond

International Workshop
on Theoretical Computer Science, WTCS 2012
Dedicated to Cristian S. Calude
on the Occasion of His 60th Birthday
Auckland, New Zealand, February 21-24, 2012
Revised Selected and Invited Papers

Springer

Volume Editors

Michael J. Dinneen
Bakhadyr Khoussainov
André Nies
University of Auckland
Department of Computer Science
Private Bag 92019, Auckland 1142, New Zealand
E-mail: {mjd, bmk, andre}@cs.auckland.ac.nz

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-27653-8 e-ISBN 978-3-642-27654-5
DOI 10.1007/978-3-642-27654-5
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011944976

CR Subject Classification (1998): F.2, F.4.1, I.2, F.1, F.1.1, F.4.3

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The International Workshop on Theoretical Computer Science (WTCS 2012), dedicated to Cristian Calude's 60th birthday, took place during February 21–24 in 2012 in Auckland, New Zealand. This volume titled *Computation, Physics and Beyond*, based on WTCS 2012, is published in the LNCS Festschrifts Series by Springer. The volume contains contributions from invited speakers and regular papers that present either expository/survey results or original research in the following areas (in which Cristian Calude has either made significant contributions or has an interest):

- Algorithmic information theory
- Algorithms
- Automata and formal languages
- Computing and natural sciences
- Computability and applications
- Logic and applications
- Philosophy of computation
- Physics and computation
- Unconventional models of computation

Prof. Cristian (Cris) S. Calude

The following eminent researchers were invited to give lectures at the conference and contribute to the Festschrift volume.

D. Bridges (Canterbury University)
C. Câmpeanu (University of Prince Edward Island)
B. Cooper (Leeds University)
R. Freivalds (University of Latvia)
H. Jürgensen (University of Western Ontario)
G. Longo (École Polytechnique, Paris)
S. Marcus (Romanian Academy)
H. Maurer (Graz Technical University)
J. Patarin (Université Versailles)
B. Pavlov (Massey University)
G. Rozenberg (Leiden University)
A. Shen (University of Marseille)
L. Staiger (Martin Luther University)
K. Svozil (Vienna Technical University)
K. Tadaki (Chuo University, Tokyo)
S. Yu (University of Western Ontario)
L. Vîţă (NZ Customs)
H. Zenil (Wolfram Research)
M. Zimand (Towson Univ)
S. Wolfram (Wolfram Research)

Other invited contributors agreeing to contribute to this Festschrift volume dedicated to Cris include:

G. Chaitin (IBM Research, New York)
R. Downey (Victoria University, NZ)
M. Dumitrescu (University of Bucharest)
L. Kari (University of Western Ontario)
Y. Manin (Max Planck Institute)
Gh. Păun (Romanian Academy)
A. Salomaa (Turku University)
K. Salomaa (Kingston University)
I. Streinu (Smith College)
I. Tomescu (University of Bucharest)

The Program Committee consisted of B. Cooper, F. Costa, M. J. Dinneen, P. Hertling, B. Khoussainov (Chair), F. Kroon, Y. Matiyasevich, A. Nies, Gh. Păun, G. Rozenberg, K. Salomaa, L. Staiger, A. Shen, F. Stephan and M. Zimand. They appreciate the additional work done by the following referees for the conference volume:

Vasco Brattka	Rupert Hölzl	Ulrich Speidel
Elena Calude	Yun-Bum Kim	Mike Stay
Rodney Downey	Gaven Martin	Kohtaro Tadaki
Noam Greenberg	Erik Palmgren	Karl Svozil

The careers of the three editors of this book have been influenced by Cris' research in algorithmic randomness, as well as his tireless administrating and organizing work. Soon after his arrival in Auckland in the early 1990s, Cris, jointly with Douglas Bridges, who was then at the University of Waikato, established the Centre for Discrete Mathematics and Theoretical Computer Science (CDMTCS). This led to the formation of the first computer science theory group in New Zealand. With the creation of the CDMTCS and his research work, Cris put the Computer Science Department at the University of Auckland on the map. All three of us were recruited by the department with strong support from Cris. In the mid-1990s, Calude, jointly with Khoussainov, Hertling and Wang, wrote a few papers, including "Recursively enumerable reals and Chaitin Omega numbers," which was published in the *Proceedings of STACS 1998*, and later in the journal *Theoretical Computer Science*. These papers, along with early work by Chaitin, Kučera, Kurtz, Solovay and Terwijn, laid the foundation for the development of the modern theory of algorithmic randomness as expressed in the work of Downey, Hirschfeldt, Miller, Nies, Slaman, Stephan, and many others.

The paper by Calude, Khoussainov, Hertling and Wang for the first time studied the concept of Solovay reducibility (from a 1975 manuscript) on the real numbers and introduced computably enumerable presentations of reals. The authors established some fundamental properties of Solovay reducibility, such as the equivalence classes of computably enumerable reals form an upper semi-lattice. The (Chaitin) Ω numbers form an equivalence class which is the largest element in this semi-lattice. This paper proposed the problem of whether every random computably enumerable real is a Ω number, i.e., corresponds to the largest element in the semi-lattice. This problem attracted the attention of many experts in the theory of randomness and computability. Kučera and Slaman

answered the question positively in "Randomness and recursive enumerability" (*SIAM J. of Computing*) in 2001.

A related 2002 paper also inspired by the work of Cris and his collaborators is "Randomness, computability and density" by Downey, Hirschfeldt and Nies (*SIAM J. of Computing*), where the density of the semilattice is established. A further question was whether for every splitting of an Ω number as a sum of two computably enumerable reals, one of the two has to be an Ω number as well. They answered the question in the affirmative. (Curiously, later on it turned out that O. Demuth, a constructivist working in isolation in Prague, already had known this in 1975.)

Cris' work was essential for establishing the leading role of New Zealand in the area of algorithmic randomness, which is evidenced by the recent publication of Nies' book *Computability and Randomness* published by Oxford University Press in 2009, and Downey and (former Wellington postdoc) Hirschfeldt's book *Algorithmic Randomness and Complexity* by Springer in 2010.

An Ω number is simultaneously computably enumerable and random; its weak form of computable approximability (the first property) is limited by the last property which implies bi-immunity, i.e., every algorithm can compute at most finitely many exact bits of such a number (none in the case of a Solovay's number, a special type of Ω number). The work Michael Dinneen did with Cris and, initially, with their former PhD student C.-K. Shu, combined the theoretical analysis with an extensive computation to calculate exactly the values of finitely many initial bits of a natural Ω number (64 in the first case). This result—the first computation of "a glimpse of randomness"—was extensively cited and commented on (for example, in the *New Scientist*); its meaning is discussed in Chaitin's paper included in this volume. This work paved the way for a more practical and complexity-theoretic approach to randomness, which includes theoretical and experimental studies of quantum randomness (work jointly done with Cris, M. Dumitrescu and K. Svozil). Michael appreciates his cooperation with Cris in the emerging field of unconventional/natural computing, e.g., the bead search/sorting was developed with their former PhD student J. Arulanandham, and in the study of the complexity of mathematical problems (joint work with E. Calude). They organized many CDMTCS international conferences together, including most editions in the series of conferences "Unconventional Computation" that started in Auckland in 1998.

We all value our friendship with Cris and the mentoring advice he has provided over the past 16 years. Our close relationship with Cris goes beyond academic collaboration. For instance, Bakh wishes that he could play tennis at the level of Cris; Michael is envious of Cris' air gun collection; André wishes he could also organize a workshop on a boat going down the Nile river.

The local Organizing Committee at the University of Auckland wishes to acknowledge the contributions of Gill Dobbie and Bob Doran. We thank the Department of Computer Science (University of Auckland), the Faculty of Science (University of Auckland), and the New Zealand Marsden Fund for monetary support. Last but not least, it is a great pleasure to thank the fine co-operation

with the *Lecture Notes in Computer Science* team of Springer for producing this volume in time for the conference.

This book is organized as follows into themes related to Cris' research area. The first part consists of a couple of papers discussing Cris' life achievements. This is then followed by papers in the three general areas of complexity, computability and randomness; physics, philosophy (and logic) and computation; and algorithms, automata and formal models (including unconventional computing). Finally, we mention that the front cover art of this book highlights the first 40 exact bits of an Ω number recently computed by Cris and the first editor.

November 2011

Michael J. Dinneen
Bakhadyr Khoussainov
André Nies

Table of Contents

Life Story

Complexity, Computability and Randomness

Physics, Philosophy and Computation

Algorithms, Automata and Formal Models

The Art of Reaching the Age of Sixty

Solomon Marcus

Stoilow Institute of Mathematics, Romanian Academy, Bucharest, Romania
solomarcus@gmail.com

Two Key Words: 'Interaction' and 'Impact'

I agree with what some philosophers (such as Charles Sanders Peirce and Jacques Derrida) believed: We are 'defined' by the interactions with other people. Our identity is of a field type (like the identity of an atom given by its interactions with the other atoms), not of an entity type. As a corollary, the best assessment of our life achievements comes from the impact of our activity, particularly from the reaction of other people to our accomplishments. The key words here are interaction and impact. Through the glasses of this philosophy, I will try to contemplate the personality of Professor Cristian S. Calude as a scholar and as a person.

I am, in this respect, in a privileged situation. For almost forty years, I have been in a continuous interaction with Cristian. After a period of 20 years, in which if our interaction was not a direct, face-to-face communication, we used almost daily the telephone or/and the traditional mail, another period of 20 years followed, during which the Internet and the email became our permanent way of interaction, as a challenge to the enormous geographic distance between us.

I remember my first meeting with Cristian. It happened in September 1972, when he was a student in the second year of the Faculty of Mathematics of the University of Bucharest. I was then his teacher of real analysis, measure theory and general topology, typical fields of continuous mathematics. Who could have imagined, at that time, that Cristian will be able to transfer many ideas from these disciplines in the discrete fields represented by mathematical logic and algorithmic information theory?

A Message from Moisil

We became rapidly aware that we are, intellectually, in "the same equivalence class", on "the same wave length". It was a hard time for Cristian. In April 1973 his mother passed away at a very young age, and less than two months later, our common mentor, Professor Grigore C. Moisil, a pioneer of mathematical logic and computer science in Romania, passed away too. I told Cristian: I have a message for you. It was initially a message from Moisil to me, before his trip to Canada, in May 1973. On the eve of his trip, I accompanied Professor Moisil to his home in 14 Armenească Street, Bucharest. At the moment when I had

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 1–19, 2012.

to leave him, he said: "Do you know who is the author of the first example of a recursive function which is not primitive recursive?" Ackermann, I said. "No", he said; "it is Gabriel Sudan.[1] Now I am in a hurry, I will tell you more when I come back from Canada". But this never happened, he died in Ottawa. I checked all Sudan's papers and none of them refers, neither in its title, nor in its introduction, to an example of Ackermann's type. I told all these things to Cristian and I challenged him: "Would you like to engage in the detective enterprise of locating the example of a recursive function which is not primitive recursive in Sudan's texts?".

An Event Deciding Calude's Direction of Research

Cristian, an undergraduate student who was then at the very beginning of his scientific career, accepted enthusiastically this challenge and the success of this adventure (see, for more, C. Calude, S. Marcus, I. Ţevy. The first example of a recursive function which is not primitive recursive, *Historia Math.* 9 (1979), 380–384) decided his direction of scientific research. It took a long time for the community to recognise Sudan's result, but now this is almost everywhere accepted (Odifreddi's classical monographs *Classical Recursion Theory* (North-Holland 1989, 1999) were the first to cite Sudan).

Recursive Functions Faced with 'P vs. NP'

This investigation lead naturally to a couple of natural interesting problems: 1) comparing Sudan's function to other recursive but not primitive recursive functions, particularly Ackermann's function and Knuth iterated powers, 2) studying the reasons for non primitive recursiveness of these functions, 3) exploring the similarities and differences between the recursive and non-primitive recursive functions and their graphs, 4) constructing hierarchies of primitive recursive functions using fast growing recursive and non-primitive recursive functions, and 5) measuring the size of the set of recursive and non-primitive recursive functions.

The "immediate" reason for the non-primitive recursiveness of Ackermann/Sudan/Knuth (shortly ASK) functions is their huge growth. What about the time and space complexities of these functions? The time complexity (but not the space complexity) is responsible for them being not primitive recursive. This analysis suggested an iterative efficient way to compute the ASK functions (Calude, Vieru) and a criterion for a function to have its graph in the nth Grzegorczyk class (Buzeţeanu, Calude). Using this criterion, they proved that every ASK function has an elementary graph (a more elaborate argument by Calude shows that the graph is even rudimentary). Consequently, computing the value of an ASK function is difficult, but checking that a natural number is the value

[1] A Romanian mathematician, who obtained a PhD from Göttingen University, under D. Hilbert, in the twenties of the past century.

of an ASK function in a given input is easy. This is a form of the 'P vs. NP' problem proved in the negative for the class of recursive functions.

Most of the results obtained by Calude and his co-authors Ş. Buzeţeanu, N. Dima, B. Fântâneanu, S. Marcus, L. Sântean (Kari), M. Tătărâm, I. Ţevy, V. Vieru have been presented in Calude's monograph *Theories of Computational Complexity* (North-Holland, 1988), one of the frequently cited book for primitive recursiveness (even by researchers in other fields, e.g. M. Kojman, S. Shelah. Regressive Ramsey numbers are Ackermannian, *J. Comb. Theory* A, 86, 1 (1999) 177–181).

Most Recursive Functions Are Not Primitive Recursive

How "frequent" are the recursive and non-primitive recursive functions in the class of recursive functions? One possible way to give an answer is to use a topological approach, tailored for countable sets. After discovering that, from such a topological view point, most recursive functions are just in the situation of Ackermann's and Sudan's examples, they are not primitive recursive, Calude realised that this approach could lead to similar interesting and surprising results in the field of computability, in algorithmic information theory and in the field of incompleteness. His intuition proved to be right.

Consider a topological space in which Baire category theorem holds true. By constructivisation of the notions of meagre and second Baire category sets one obtains a useful tool to measure the size of countable sets. These sets are called effectively meagre and effectively second Baire category. An effectively meagre set is "smaller" than a meagre set and an effectively second Baire category set is "larger" than a second Baire category set. Many topological spaces Cristian considered are in a way or another related to Baire space.

In one of the first results in this direction, he proved that the set of recursive functions is an effectively second Baire category set while the set of measured sets in a Blum abstract space is effectively meagre. As a consequence, every complexity class in a Blum abstract space, the set of rudimentary functions, the set of Kalmár functions, each Grzergoczyk class, the set of primitive recursive functions are all effectively meagre.

The Law of Large Numbers Is False in the Sense of Baire Category

The celebrated theorem of Martin-Löf shows that the set of Martin-Löf random sequences (reals) has effectively Lebesgue measure one (here the effectivization is based on effectively null sets, that is (classical) null sets which can be covered uniformly with a computably enumerable union of intervals with rational end-points having effectively arbitrarily small measure; in contrast with the case of classical null sets, the union of all effectively null sets is itself an effectively null set, the base for Martin-Löf definition of random sequences). Is this result,

showing that the set of random reals is "large" in measure/probability, confirmed by a Baire category theorem? The answer is negative as Calude and Chiţescu proved: the set of random reals is effectively meagre with respect to the natural topology on the unit interval (the paper was reviewed by J. Oxtoby in Mathematical Reviews). In particular, the law of large numbers is false in the sense of Baire category.

Small variations operated on the unit interval topology rectify this asymmetry as Calude, Marcus and Staiger proved: in these new topological spaces the set of random reals is effectively second Baire category.

The Set of Disjunctive Infinite Sequences Is a Constructive Residual

A different path in restoring the symmetry between measure and Baire category consists in weakening the definition of randomness to disjunctivity: an infinite sequence is disjunctive (or a lexicon) if every string appears in the sequence (so it appears infinitely many times). How large is this set? Calude and Zamfirescu proved that the set of lexicons is a constructive residual. Such a set is "very large" because it has both effective measure one and is effectively second Baire category. As a consequence, the typical real number is a lexicon, i.e., constructively most numbers do not obey any probability law. To achieve their goal the authors proved a constructive version of (a weak form of) Lebesgue's density theorem, a result interesting in itself.

Other similar results for Blum abstract spaces and random objects have been obtained by K. Ambos-Spies, E. Busse, Calude, C. Câmpeanu, G. Istrate, L. Staiger, M. Zimand (some discussed in Zimand's monograph *Computational Complexity: A Quantitative Perspective*, Elsevier, Amsterdam, 2004).

A natural problem in mathematical logic is to determine how pervasive is Gödel's incompleteness theorem, i.e., how large is the set of true but unprovable sentences in a from theory subject to Gödel's incompleteness. A topological answer was given by Calude, Jürgensen and Zimand: with respect to a large class of topologies, the set of true but unprovable sentences is effectively second Baire category.

Most Situations Are Anti-intuitive

One can observe that in most of these results the "majority" is represented by objects which, with respect to our intuitive perception and expectations, appear as exceptional, singular, and, as a consequence, it is very difficult to capture them. As a matter of fact, the same phenomenon appears in the field of mathematical and functional analysis, in general topology. The chronologically first example of a category type theorem belongs to Mazurkiewcz and Banach (in the thirties of the 20th century) and states (in particular) that in the space of real continuous functions on the interval [0, 1] the functions having in at

least one point a finite derivative form a meagre set (first Baire category set). In the context of the whole history of continuity and differentiability, this result came as a shock, because it shows that, against our expectations, the examples of everywhere continuous but nowhere differentiable functions, considered from their first appearance in the 19th century, as an exceptional, singular situation, are, in the global perspective of the category theorems, just the typical case of continuous functions. This phenomenon remains valid in the category theorems in discrete mathematics: in computability theory, in algorithmic information theory and in the field of incompleteness.

This discrepancy between the global theoretical perspective, on one hand, and the status of individual entities, on the other hand, has its secret. The exceptional sets, in most theorems involving negligible sets, have a non-effective, i.e., non-constructive status because their existence is proved using a non-effective axiom/procedure, for example, the axiom of choice.[2] For most elements we have no possibility to decide whether they belong or not to the exceptional set. For instance, a monotonous real function f on the real interval $[a, b]$ is almost everywhere differentiable in $[a, b]$ (Lebesgue), but given a point x of continuity of f, we cannot decide whether f is differentiable in x. Similarly, Cantor's theorem asserting that almost all real numbers are transcendental came as a shock, taking in consideration the difficulty of describing individual transcendental numbers, as Liouville did in the first part of the 19th century. A similar situation appears in most theorems involving negligible sets in discrete mathematics.

Facing Two Major Changes

We are now, with our presentation of Cristian's scientific achievements, approximately at the moment when his life had two major changes: the fall of Romanian communism and, in short time after this, but as a consequence of the freedom acquired, his move from Romania to New Zealand. This big move, suggested and supported by his friends Douglas Bridges, Bob Doran and Hermann Maurer, was also an enormous challenge, testing his capacity to face unexpected situations. Now we can say that he faced brilliantly this challenge, transforming apparent obstacles into advantages. This evolution can be seen as an instance of the principle of Prigogine's dissipative structures: a system has a better chance to acquire a higher stage when it has to face opposite trends.

Before 1990, the main reference in Cristian's work was P. Martin-Löf randomness, and the main synthesis of his work has been the monograph *Theories of Computational Complexity* (1988). But the fields where his creativity has been at the highest level appeared in his studies only after 1990. Let us first illustrate in a quantitative way what freedom (associated with the appearance of the Internet, shortly after the fall of Romanian communism) implied for Cristian's scientific productivity, capacity of interaction, impact and creativity. We will compare the 20 years before 1991 with the 20 years after 1990. Papers in refereed journals: 56

[2] Can one prove that examples of such type cannot be constructed in Bishop mathematics?

before, 94 after. Papers in refereed conference proceedings: 16 before, 35 after. Papers in refereed collective books: 4 before, 20 after. Books published at international publishing houses: 1 before, 3 after. Editor or co-editor of collective books, in English: 1 before, 25 after. Editor of special issues of international journals: nothing before, 32 after. Needless to say, some of these statistics speak about Cristian's huge capacity to care about the global situation of the fields of his interest, to organise meetings and to attend meetings, to initiate special issues of journals, devoted to the hottest problems in the field. To give only one example: he is the initiator of the annual international conferences for which he coined the name "Unconventional Computing", bringing together hundreds of researchers.

At the Crossroad of the Hottest Contemporary Trends

In order to show the variety of his interests, their modernity and their position at the interface with some of the main paradigms of today science, I will mention here some journals where his articles have been published in the 21st century: *Fundamenta Informaticae, Applied Mathematics and Computation, Mathematical Structures in Computer Science, Journal of Foundations of Computer Science, Theoretical Computer Science, Chaos, Journal of Computer and System Sciences, Physical Review, London Mathematical Society Journal of Computer Mathematics, Complex Systems, Notices of the American Mathematical Society, Information and Computation, Advanced Science Letters, Advances in Applied Mathematics, International Journal of Quantum Information, International Journal of Bifurcation & Chaos, Journal of Mutiple-Valued Logic and Soft Computing, Annals of Applied and Pure Logic, International Journal of Theoretical Physics, Communications in Non-Linear Science and Numerical Simulation, Mathematical Logic Quarterly, Bio-Systems, The New Scientist, Journal of Universal Computer Science, Information Processing Letters, Experimental Mathematics, Minds and Machines, Journal of Artificial Intelligence, Philosophy and Cognitive Sciences, Chaos, Solitons and Fractals, Complexity, Nature, La Recherche.* In these titles we find many of the basic key words of the contemporary trends: applied mathematics, computation, foundations of computer science, chaos, complexity, system sciences, information, quantum information, fractals, multi-valued logic, soft computing, theoretical physics, nonlinear science, numerical simulation, bio-systems, experimental mathematics, minds and machines, artificial intelligence, cognitive science, philosophy.

Focus on Algorithmic Information

We can now move to the presentation of Cristian's most important directions of research, mainly developed in the last 20 years. Algorithmic information is his main area of research since early 80's. It started with a ten year cooperation with I. Chiţescu during which they developed many parts of algorithmic information theory on a general, non-binary framework. This approach seemed

at the beginning just a mathematical artificial generality, but later on it showed its utility in various contexts, for example in the study of quantum randomness. The series of early results includes the immunity of the set of random strings, a representability approach for Martin-Löf randomness tests (with I. Chiţescu and L. Staiger), the Baire category classification of random reals (with I. Chiţescu), the first proofs of what is now called Kraft-Chaitin (with E. Kurta, C. Grozea), a generalisation to arbitrary probabilistic algorithms of Chaitin-Schwartz theorem (with M. Zimand), and the limits of binary coding (with C. Câmpeanu).

An important result is the Chaitin-Calude-Hertling-Khoussainov-Wang-Kučera-Slaman theorem for left-computable random reals (they are exactly the halting probabilities (Omega numbers) of all self-delimiting universal Turing machines). This result has been extended by Calude, Hay and Stephan to left-computable ε-random reals. Calude's extension of Solovay's theorem on Omega numbers for which ZFC cannot compute any bit (ZFC cannot compute more than finitely many bits of every Omega number, as Chaitin proved) was appreciated as "the best possible result in this direction" by Hirschfeldt's review in *Mathematical Reviews* 1 923 902. In the same area Calude and Hay have studied the provability of randomness and solved a problem proposed by Solovay showing a sharp distinction between random strings and random reals: ZFC cannot prove the randomness of more than finitely many random strings (Chaitin's theorem), but can prove randomness for every left-computable random real.

Studying Omega Numbers with Strong Reducibilities

There has been a recent flowering of deep results relating classical computability and algorithmic randomness. Calude and Nies were at the forefront of this trend in their studying Omega numbers with strong reducibilities. Other results in this direction were obtained in cooperation with Coles, Hertling, Khoussainov and Wang. Two recent monographs, Nies, *Computability and Randomness* (Clarendon Press, 2009) and Downey and Hirschfeldt, *Algorithmic Randomness and Complexity* (Springer, 2010) present a synthesis of this trend. Calude's role in this direction was acknowledged in Downey and Hirschfeldt's Preface of their book:

> Though we did not know it at the time, this book genesis began with the arrival of Cris Calude in New Zealand. Cris has always had an intense interest in algorithmic information theory. The event that led to much of the recent research presented here was the articulation by Cris of a seemingly innocuous question. This question goes back to Solovay's legendary manuscript, and Downey learned of it during a visit made to Victoria University in early 2000 by Richard Coles, who was then a postdoctoral fellow with Calude at Auckland University. In effect, the question was whether the Solovay degrees of left-computably enumerable reals are dense.

A Highly Appreciated Monograph

Calude's monograph *Information and Randomness: An Algorithmic Perspective*, published in two editions, 1994 and 2002, by Springer, includes, among other topics, the first systematic study of left-computable random reals, the origin of many recent studies. Cited by virtually every researcher in algorithmic information theory the book was also used for courses in many universities around the world including University of Chicago, UCLA, UWO, University of Ulm, Siena University, Technical University of Vienna, Heidelberg University, Halle University, etc. Here are two comments about its second edition:

> This book, benefiting as it does from Cristian Calude's own research in AIT and from his experience teaching AIT in university courses around the world, has helped to make the detailed mathematical techniques of AIT accessible to a much wider audience. This vastly expanded second edition collects in one place much exciting recent work of its author and others. (G. J. Chaitin)

> The vigorous growth in the study of algorithmic information theory has continued during the past few years, which is clearly visible in the present second edition. ... The author has been directly involved in these [new] results that have appeared in the prestigious journals like *Nature, New Scientist* and *Pour la Science*. (A. Salomaa)

A Probability Space Where the Computational Time Plays a Crucial Role

I have already mentioned Omega numbers which offer a probabilistic response to the famous undecidability of the halting problem. A different probabilistic avenue in the study of the halting problem was taken by Calude and Stay who considered a more complex probabilistic space: the Lebesgue probability used for the Omega numbers was replaced with a probability space in which the computational time plays a crucial role. They proved that given an integer $k > 0$, we can effectively compute a time bound T such that the probability that an N-bit program will eventually halt given that it has not halted in time T is smaller than 2^{-k}. As consequences one gets the following facts: a) the (computably enumerable, but not computable) set of halting programs can be written as a disjoint union of a computable set and a set of effectively zero probability, so the source of undecidability is located to a small "bad" set, b) the set of times at which an N-bit program can stop after the time $2^{N+\text{constant}}$ has effectively zero density, because they are all non random times. The role of computational time is essential but the type of computational resource is not, as Y. Manin has showed: the result is true for many other computational resources and can be viewed as a cut-off type of argument developed in artificial intelligence. This can lead to a bridge between quantum field theory and classical computing. In

spite of undecidability, the halting problem can be solved probabilistically with arbitrary precision (experimental evidence was recently obtained by Delahaye and Zenil). A recent special issue of the journal *Mathematical Structures in Computer Science* edited by Calude and Cooper contains several papers that further develop these ideas.

A Decidable, Weaker Algorithmic Information Theory

A variant of algorithmic information theory in which the underlying model of computation is not the Turing machine, but a finite transducer was recently developed by Calude, K. Salomaa and Roblot. In contrast with the classical algorithmic information theory, this weaker theory is decidable, so it has more chances of practical applicability. This type of research is not atypical for Calude who paid close attention to applications of algorithmic information theory. Here are a few areas: mathematical logic (Jürgensen, Zimand), probability theory (M. Dumitrescu, M. Zimand), complex analysis (P. Hertling, B. Khoussainov), quantum physics (A. Abbott, M. Stay, L. Staiger, K. Svozil), evaluation of the complexity of mathematical problems (E. Calude, Dinneen), cellular automata (Hertling, H. Jürgensen, K. Weihrauch), image processing (J. P. Lewis).

Bridging Computation Theory with Theoretical Physics

Modern ways to bridge computation theory with theoretical physics is another central direction of Cristian's research. His first results concerned automata-theoretic models for quantum complementarity (E. Calude, M. Lipponen, C. Ştefănescu-Cuntze, K. Svozil, S. Yu). They have been followed by a series of theoretical and experimental papers trying to understand the limits of quantum computing (with Abbott, M. Cavalieri, R. Mardare, Svozil). The starting point was Calude's surprising proof that the famous Deutsch's problem can be solved classically with the same amount of resources as quantum mechanically (he coined the term "de-quantisation" for the process of extracting a classical algorithm from a quantum black-box algorithm, which solves the same problem and is as performant as the original one). Calude's student A. Abbott obtained further examples of de-quantisation including one for the quantum Fourier transform, and together with Calude, Bechmann, and Sebald proposed a nuclear magnetic resonance implementation of the de-quantisation of an instance of Deutsch-Jozsa algorithm.

Bridging Heisenberg and Gödel

Work with his student Stay revealed strong relations between Heisenberg uncertainty principle and Gödel's incompleteness phenomenon and sketched an algorithmic version of statistical mechanics based on zeta functions associated

with Turing machines (in passing, a new type of machine, the tuatara machine,[3] and a new complexity, the natural complexity, have been introduced). The later work was systematically developed by the japanese mathematician K. Tadaki.

The First Mathematical Approach to Quantum Randomness

Quantum randomness is considered to have the best quality of all types of randomness produced by nature. Is this assertion more than a belief or a hypothesis? Let us consider a spin-1 particle in a 3-dimensional Hilbert space. According to Kochen-Specker theorem, the strongest "no-go theorem" in quantum physics, it is impossible to assign definite values to all possible observables corresponding to the result of a measurement of that observable in a non-contextual way in agreement with with the predictions of quantum mechanics. As a consequence, one can either choose to accept a contextual but complete assignment of hidden variables in an attempt to maintain realism, or to give up the assertion that all "elements of physical reality" must exist at all times. Starting with two geometrical proofs for the Kochen-Specker theorem (with Hertling), Calude and Svozil proposed the first attempt, later developed in cooperation with Abbott, to understand quantum randomness from a mathematical point of view. Assume that a) all observables cannot have non-contextual hidden variables, b) contextual hidden variables are excluded, c) the prediction of the result of a measurement with certainty implies the existence of an element of physical reality/hidden variable corresponding to this prediction (EPR hypothesis). Let $\mathbf{x} = x_1 x_2 \ldots$ be the sequence of bits obtained from the concatenation of repeated state preparations and non-trivial measurements in a Hilbert space of dimension 3 or greater by discarding all but two possible outcomes. Then, under the above assumptions, $\mathbf{x}$ is bi-immune, that is, no Turing machine can compute more than finitely many bits of the sequence, and hence it is (strongly) incomputable. Bi-immunity is a weak form of randomness, weaker than Martin-Löf randomness; Omega numbers are typical examples of bi-immnune reals (because they are Martin-Löf random). With Lebesgue probability one every sequence produced as above is Martin-Löf random, a result which cannot exclude that the sequence produced can be sometimes computable: bi-immunity excludes this possibility. For example, in accord with the quantum mechanical prediction, such a sequence may contain a billion of zeroes, but, in view of the result above, it cannot consist only of zeroes.

With the achievements in these two fields, algorithmic information theory and bridging computation theory with theoretical physics, Calude is reaching his highest potential of creativity. To the results in these two fields refer most of the citations of his works. He is cited in more than 1500 papers (most of them published in the same journals where Cristian published his own papers) and 100 books, by more than 500 authors.[4] Among these authors, there are prestigious

[3] Tuatara, "peaks on the back" in Maori, is a reptile found only in New Zealand, the only survivor from the time of dinosaurs.

[4] www.cs.auckland.ac.nz/~cristian/citations.pdf.

names such as L. Accardi, K. Ambos-Spies, J. Baez, J. D. Barrow, J. Borwein, P. Borwein, D. Bridges, G. J. Chaitin, S. B. Cooper, M. Davis, M. Deza, R. Downey, R. Freivalds, E. Gelenbe, J. Gruska, Y. Gurevich, J. Hintikka, J. van Leeuwen, G. Longo, Yu. Manin, Yu. V. Matiyasevich, A. Nerode, H. Niederreiter, P. Odifreddi, G. Rozenberg, A. Salomaa, T. A. Sebeok, J. Shelah, M. Sipser, T. A. Slaman, C. Smorynski, J. F. Traub, V. A. Uspenski, S. Wolfram, A. Zeilinger.

Cristian's work has attracted the attention of people outside his fields of interests. Here are three examples. The infinite real time composition for computer-controlled piano "Lexikon-Sonate" composed by K. Essl in 1992 http://www.essl.at/works/Lexikon-Sonate.html illustrates Calude-Zamfirescu notion of lexicon[5]. R. M. Chute's "Poem on Ω" published in *Beloit Poetry Journal Spring* Vol. 50, No. 3 (2000), 8 was inspired by Calude and Chaitin note "Randomness everywhere" published in *Nature* 400, 22 July (1999), 319–320. A character in the CBS (US drama) TV show Numb3rs (season 5; episode 5; scene 6, http://www.cbs.com/primetime/numb3rs) recites the Omega number bits computed by Calude-Dinneen-Shu.

Experimental Mathematics and Physics

There are a few more directions of research in Cristian's biography. One of them, symptomatic for the new face of exact sciences as a consequence of the impact of computer science, is "experimental mathematics and physics". Incomputability is an asymptotic property and tests to evidence it are difficult to find. In an attempt to produce experimental evidence of the incomputability of quantum randomness, very large samples of bit-strings (2^{32}) produced with Mathematica and Maple (cyclic, so computable), π (computable but not cyclic), Quantis (quantum random bits generated with the University of Geneva commercial device), and quantum random bits produced in the A. Zeilinger's lab at the University of Vienna have been analysed with a large battery of randomness tests, most of which proved powerless to produced the desired results. The best test capable of differentiating between the quantum and pseudo-randomness generators was based on the transposition of Borel normality property from infinite sequences to strings (Calude): it showed a clear separation between the two classes of randomness generators (Calude, M. Dinneen, M. Dumitrescu, K. Svozil).

Computing Exact Bits of a Natural Omega Number

Cristian's first involvement in computer experiments concerned computing initial bits of a natural Omega number[6] (with M. Dinneen and C. Shu) and proving

[5] A real in the unit interval is disjunctive in base b in case its b-expansion contains all possible strings over the alphabet $\{0, 1, \ldots, b-1\}$. A lexicon is a real which is disjunctive in any base.

[6] Every Omega number is invariant under the change of finitely many bits, so adding any prefix to the sequence of bits of an Omega number produces also an Omega number.

facts about automata recognising no words (with C. Câmpeanu and M. Dumitrescu). Recall that an Omega number is both computably enumerable (the limit of a computable, increasing, converging sequence of rationals) and Martin-Löf random. As a consequence, an Omega number is bi-immune, a property shared with sequences of quantum random bits discussed above; in some cases, more precisely, when Omega is given by a Solovay machine, no bit of its binary expansion can be calculated and certified. With a combination of mathematical analysis and very large scale programming, the halting problem for all programs up to 88 bits was solved for a natural universal self-delimiting Turing machine and the first exact 64 bits of the associated Omega number calculated (with M. Dinneen and C. Shu); improved versions of these computer experiments were later reported by Calude and Dinneen. These results have been extensively discussed and cited by many papers and books, including books in programming like M. Trott, *The Mathematica GuideBook for Programming* (Springer, 2004) and experimental mathematics (S. R. Finch, *Mathematical Constants*, Cambridge University Press, 2003 and D. Bailey, J. Borwein, *The Experimental Mathematician*, A. K. Peters, 2003). The importance and significance of these results are discussed in Chaitin's paper included in this volume.

Formal Proofs, under the Advent of Computer Technology and Programming

Another direction, to which he devoted a very interesting study, is related to formal proofs, as they were conceived by Hilbert. Recall that Hilbert's concept of formal proof is an ideal of rigour for mathematics which has important applications in mathematical logic, but due to Gödel's incompleteness theorem, was considered irrelevant for the practice of mathematics. This situation is no longer valid, not because of theoretical results, but because of the advent of computer technology and programming. Indeed, in the last twenty years, many deep mathematical theorems have been formally proved using specialised programming languages—proof-assistants—like Isabelle or Coq. With formal proof, which has become practically achievable, correctness reaches a standard that no pen-and-paper proof can match, but an essential component of mathematics, the insight and understanding, is in danger to be lost. Calude and C. Müller have analysed a list of symptoms of understanding for mathematical proofs and then proposed an environment in which users can write and check formal proofs as well as query them with reference to the symptoms of understanding. In a nutshell, the proposed environment reconciles the main features of proof, correctness and understanding. Using the proof-assistant Isabelle, Calude and Hay developed the first formal proofs in algorithmic information theory, in particular, a proof for the representability of left-computable ε-random reals.

Unconventional Computing

His work on unconventional computing was briefly mentioned before. Unconventional computing roughly refers to any computational paradigm which at some point in time is different from main stream approaches: this may be due to a different type of hardware (quantum computing or molecular computing) or because of a completely different way to use a conventional paradigm (for example, using finite automata to compute real functions). Calude's own contributions to this area include results concerning strategies to compute the incomputable. With his co-authors he proved both negative results—like the impossibility of breaking the Turing barrier with time-travel (with M. Dinneen, K. Svozil), axiomatic versions of the incompleteness theorem (with Rudeanu), or the necessity to use an infinite computational space when computing an incomputable function even on an accelerated Turing machine (with Staiger), and positive ones as the use of an infinite dimensional quantum method to compute the halting problem (with V. Adamyan and B. Pavlov), the use of gravity to perform fast sorting and searching (the so-called bead-sort proposed with J. Arulanandham and M. Dinneen), and the use of accelerated membranes (with Păun) and quantum random oracles (with Abbott and Svozil) as methods for breaking the Turing barrier. The book *Computing with Cells and Atoms*, (Taylor & Francis, London, 2001) by Calude and Păun, one of the first monographs in the area of unconventional computing, has been frequently cited and used as textbook in many universities around the world.

Constructivity, Mathematical Logic and Philosophy

From computability and complexity to constructive mathematics is not a long way and Cristian crossed it several times. He proposed constructive approaches to Hilbert's basis theorem (with Vaida), the inverse function theorem (with D. Bridges, B. Pavlov, D. Ştefănescu) and Poincaré-Hardy inequality on the complement of a Cantor set (with Pavlov); he also obtained recursive bounds for the exceptional values in Blum's speed-up theorem in abstract complexity spaces (with Bridges).

Cristian's interests in mathematical logic problems naturally lead him to philosophy of mathematics and physics. A series of papers with E. Calude, Chaitin and Marcus reflect on the role and power of proof in mathematics. A new perspective on the evolution and history of the idea of mathematical proof was proposed in a 3-dimensional analysis: syntactical, semantical and pragmatical. Proofs by computers and proof-assistants, and proofs "allowed" in various hypothetical physical universes (proofs exist because of the mathematical and logical determination, but also because the laws of the universe allow them to be thought and communicated). The lawlessness of the universe was argued in papers with W. Meyerstein and A. Salomaa, a formal model of God was proposed in a joint paper with Marcus and D. Ştefănescu, and graphical illustrations of randomness were produced with A. Gardner and M. Dinneen.

The Metric Method

We started this article with Moisil's problem which lead to Cristian life's long interest in computability and complexity. More or less at the same time, motivated by Marcus' interests, Cristian was attracted to problems in mathematical linguistics, like the morphology of programming languages and contextual analysis. The metric method—his preferred approach in this area—was used to study formal languages (with K. Salomaa and S. Yu), but also to obtain an algorithmic solution to multi-criteria aggregation problems (with E. Calude) and to the construction of a multi-criteria metric algorithm and recommender system (with A. Akhtarzada and J. Hosking). The discrete metrics introduced by C. Calude and E. Calude are presented in the *Encyclopedia of Distances* (M. Deza and E. Deza, Springer, 2009).

Popular Articles and Reviews

Cristian wrote articles for wider audiences which have appeared in prestigious journals or science magazines like *Nature, Notices of AMS, The New Scientist* (Calude, J. Casti and Chaitin), *Singularité, Complexity, Pour La Science, La Recherche* (Calude). He also wrote prefaces to books by Chaitin and L. Vîţă, more than 550 reviews in *Mathematical Reviews, Zentrablatt für Mathematik, Computing Reviews*, and 55 columns "News from New Zealand" in the *Bulletin of EATCS*.

Invited Lectures and Seminars

Cristian was an invited lecturer to many prestigious international conferences and workshops including International Conference on Discrete Mathematics (Dortmund, Germany, 1991), The Foundational Debate. Complexity and Constructivity in Mathematics and Physics (Vienna, Austria, 1994), Constructivity and Complexity in Analysis (Dagstuhl, Germany, 1997), Conference 'Integrability and Chaos in Discrete Systems' (Brussels, Belgium, 1997), Millennial Symposium 'Defining the Science of Stochastics' (Würzburg, Germany, 2000), Second Pacific Rim Conference on Mathematics, Institute of Mathematics (Taipei, Taiwan, 2001), NZ Mathematical Colloquium (Auckland, NZ, 2002), Workshop on Natural Processes and Models of Computation (Bologna, Italy, 2005), Kolmogorov Complexity and Applications (Dagstuhl, Germany, 2006), Workshop on Information Theories (Münchenwiler, Switzerland, 2006), Significant Advances in Computer Science (Graz, Austria, 2007), Workshop on Automata, Formal Languages and Algebraic Systems (Kyoto, Japan, 2008), New Kind of Science (Bloomington, USA, 2008), Science and Philosophy of Unconventional Computing (Cambridge, UK, 2009), Conference on Logic, Computability and Randomness (Notre Dame, USA, 2010), Workshop Developments in Computational Models (Edinburgh, UK, 2010), Semantics and Syntax: A Legacy of Alan Turing (Cambridge, UK, 2012), The Incomputable (London, UK, 2012).

He gave 119 invited seminars in many universities and reputable IT companies around the world, including Brussels Free University, École Normale Supérieure, Paris, Heidelberg University, Imperial College London, Joint Institute for Nuclear Research (Dubna), Mathematical Institute (Belgrade), Mathematical Institute (Bucharest), Martin-Luther-Universität Halle, Open University Hagen, Oxford University, Université de Bourgogne, University of Leeds, University of Rome "La Sapienza', University of S. Petersburg, University Sorbonne Paris (in Europe), Cornell University, Google (Mountainview), IBM (New York), Microsoft (Trento), National Sandia Laboratories (Albuquerque), Queens University, Rutgers University, Schrödinger International Institute for Mathematical Physics (Vienna), Technical University of Vienna, Turku University, Universidad de Chile, University of California at Berkeley, University of California at San Diego, University of Chicago, University of Massachusetts at Boston, Université Paris Sud, University of Toronto, University of Waterloo, University of Western Ontario, Wesleyan University (in Americas), Academia Sinica, Taipei, Canterbury University, Hong Kong University of Science & Technology, Kyoto Sangyo University, National University of Singapore, University of Newcastle, Victoria University (Melbourne) (in Australasia), University of Capetown, University of South Africa (Pretoria) (in Africa).

Attracting Students to Research

Cristian started "advising" when he was in the last undergraduate year: two of his colleagues have worked "with him" for their Diploma Theses (officially, the supervisor was one of their professors). He then "un-officially" co-supervised with me a few PhD students, till he was granted himself the habilitation to supervise PhD students.[7] Overall, he has supervised 4 post-doc fellows, 15 PhD students, 27 MSc students, and 18 research/visiting students.

There is no space to discuss in detail his students achievements, so I will make only some global comments. Most of his students, today well-known experts in their fields, work in prestigious academic, research institutions or major companies, scattered all around the world: University California at San Diego, Towson University, Universität der Bundeswehr München, University of North Carolina at Charlotte, University of Western Ontario, IBM Research, Google Mountainview, Wolfram Research. Cristian keeps in touch regularly with most of them: he knows details about their careers—results, awards, promotions—, but also about their families. They meet at conferences or universities, and he continues to collaborate with quite a few of them, not only with the youngest ones. Some of his graduate students and post-docs have been cited above (in alphabetical order): A. Abbott, A. Akhtarzada, J. Arulanandham, Ş. Buzeţeanu, C. Câmpeanu, R. Coles, N. Dima, B. Fântâneanu, C. Grozea, N. Hay, P. Hertling, G. Istrate, E. Kurta, C. Müller, M. Lipponen, L. Sântean (Kari), T. Roblott, C. Shu, M. Stay, M. Tătărâm, L. Vîţă, Vieru, Y. Wang, H. Zenil, M. Zimand.

[7] In Romania, the habilitation is granted by the Ministry of Science and Education.

Editorial Activity

Calude has and is serving in many editorial boards of book series (*Discrete Mathematics and Theoretical Computer Science*, Springer-Verlag, London (from 1996 to 2004) and *European Association for Theoretical Computer Science*, Springer-Verlag, Heidelberg, (from 2004 on)) and international journals (founding editor–in–chief, *Journal of Universal Computer Science*, Springer-Verlag (from 1994 to 2009) and member of the Editorial Board (from 2009 on), *Analele Universităţii Bucureşti, Matematică–Informatică* (from 1988 to 2006), *Bulletin of the European Association of Theoretical Computer Science* (from 1993 on), *Grammars* (from 1997 to 2003), *Fundamenta Informaticae* (from 1997 on), *Romanian Journal of Information Science and Technology* (from 1998 on), *Natural Computing Journal* and *Contributions to Discrete Mathematics* (from 2005 on), *International Journal of Foundations of Computer Science* and *Mathematics Applied in Science and Technology* (from 2006 on), *unoMolti, Modi della Filosofia* and *Revista de Filosofie Analitica* (from 2007 on), *The Open Software Engineering Journal* (from 2008 on), *Theoretical Computer Science, International Journal of Nanotechnology and Molecular Computation* (from 2009 on), *Mathematical Structures in Computer Science, International Journal of Unconventional Computing* (from 2010 on)). He was an associate–editor of the *Handbook of Formal Languages*, (Springer-Verlag, 1997) and a member of the Advisory Board of the *Handbook of Natural Computing: Theory, Experiments, and Applications*, (Springer, 2011).

Awards and Distinctions

Calude was awarded prizes and distinctions from prestigious academic organisations and learned societies: Visiting Fellow of Isaac Newton Mathematical Institute (2012) and London Mathematical Society (2010), Hood Fellow (2008–2009), Member of Academia Europaea (2008; member of the Informatics Section Committee, 2010–2013), Dean's Award for Excellence in Teaching, University of Auckland (2007), Award for Excellence in Research, University of Bucharest (2007), "Gheorghe Lazăr" Mathematical Prize of the Romanian Academy (1988), Computing Reviews Award of the Association for Computing Machinery, New York (1986), and Mathematical Student Prize, University of Bucharest (1975).

A Few Words about Cristian's Family and Childhood

Rarely I have seen such a highly professional curriculum vitae as that posted by Cristian Calude on his website.[8] However, something is missing: details about his family and his first years of life. A person with his remarkable achievements in research and education deserves to be better known, in particular, his early years of education in family and at school.

[8] `www.cs.auckland.ac.nz/ cristian/criscv.pdf.`

Despite the huge distance between New Zealand and his place of birth, the city of Galaţi, Romania, he goes there almost every year. A mathematical competition, started there ten years ago, bears his name.[9] In the late seventies I met his maternal grandmother Sultana Bobulescu and, when she had a car accident, I visited her in the hospital. I met his father Constantin Calude several times. He was a lawyer of excellent repute who was awarded the national order "Star of Romania"—Romania's highest civil order—for exceptional military services during WW2. His wife Elena studied mathematics at the University of Bucharest and she wrote her Diploma Thesis *Mathematical Analysis of the Drama "Long Days Journey Into Night" by E. O'Neill* under my guidance. I followed the evolution of their daughter Andreea: after having obtained a PhD in linguistics and a BSc in mathematics from the University of Auckland, she is now a Post-Doc Fellow at Reading University in UK.

Here are several interesting facts, explaining the roots of his scientific career; I learned them from Cristian, but also from his recent interviews.

His grandfather (from the mother side) Marcel Bobulescu was a very successful investor (before the communist regime put an end to this type of activity) and a chess passionate; he published chess problems and solutions for a local newspaper. His grandfather taught him chess him when he was five or six years old; through chess Cristian first met formal rules. Playing chess with his grandfather was not fun, so they switched to very simple chess problems. Later in life Cristian enjoyed playing chess, first with his father in law, Petre Anghel, then with Andreea and Elena, and (more competitively) with a colleague from the university, Peter Shields.

His mother Jeanette was Cristian's first teacher at home. She supervised his readings, writings (mostly letters and short abstracts of the books he read) and French studies (he fondly remembers his tutors, Miss Angelique and Miss Jenny). "I learned from her to keep my 'space' (toys, books, pencils, notes) tidy, to be polite and prompt. I still have piles of letters written to and received from her during my student years in Bucharest (till her untimely death); much later, when I left Bucharest for Auckland, I kept a regular correspondence with my father, for about eight years", he remembers. His father being a lawyer, Cristian learned from him the value of a logical argument and the relativity of truth (juridically, truth is only what you can prove in court). His father was also a regular contributor to law journals and Cristian remembers how proud he was when as a ten-year old kid he saw his family name printed in a journal; later, father and son jointly wrote a paper on mathematical modelling in juridical sciences. G. Stoenescu, a lawyer and friend of the family, was a role model and close friend (in spite of being 50 years older) for about 35 years; he tried in vain to teach him to play violin.

[9] http://mategl.com/concurs.pp.

His First Teachers

In the pre-high school years Cristian had two very special teachers: Irina Botezatu[10] (Romanian grammar) and Adrian Ropotă (mathematics). At the National College Vasile Alecsandri (his grandfather Ştefan and father have been students of the College; all three attended the centenary celebration of the College in 1967) he had excellent teachers, especially Dana Bogatu (Romanian literature), Feya Brener (French), Victor Necula (physics), Radu Rotta (English), and, most importantly for him, Ionel Decu (mathematics). Decu recognised Cristian's "mathematical mind" when he proposed a problem in elementary geometry with no solution (this was in his first year in high-school). His colleagues showed that various triangles do not satisfy the required condition, but he was the one to point out that a general proof was required—no triangle can satisfy the condition—and to propose an algebraic solution (not really the solution his teacher expected). Decu helped him a lot; in particular he supported Cristian's passion for popular mathematics books, a literature which was partially incomprehensible to him, but had a huge impact on his career. During that time Cristian discovered the books by Moisil (mathematical logic) and Marcus (mathematical analysis), which attracted him in an irresistible and definitive way to mathematics. The mathematics presented in these books was very different from the school mathematics: there he discovered infinity.

Cristian didn't excel in written examinations, where he had to solve problems in a limited time. However, in the third high-school year he qualified for the National Mathematics Olympiad in Bucharest. Not unexpectedly, he didn't do too well, but he solved a problem in an unconventional way. The algebraic argument attracted the attention of Gr. C. Moisil, who invited him to his house and guided his mathematical education and research from that moment till Moisil's death four years later. Under Moisil's guidance Cristian wrote his first paper and started tutoring (for Moisil's course of logic for students in law). Over the years Cristian has written fondly about Moisil.

His Life as a University Student

At the University of Bucharest Cristian was impressed and influenced by the following professors (in the order he met them): Ioan Tomescu, Dragoş Vaida, Solomon Marcus, Ion Chiţescu, Sergiu Rudeanu, and Mircea Maliţa. He has joint papers with each of them, and also with two others who were not his direct professors, Virgil Căzănescu and Monica Dumitrescu. In Auckland he continued to cooperate with his department colleagues, in both theory and applied computer science and mathematics; it seems that he has the highest number of joint papers with colleagues in his department. He had, and continue to have, a strong interaction with colleagues from his generation, particularly, G. Păun and S. Istrail.

[10] The mother of his life-long friend Dan Botezatu, a distinguished medical doctor.

A Man of His Time

From New Zealand, Cristian travels often to other parts of the world, to meet other scholars, to attend scientific meetings, to interact with people having common scientific interests. He is an example of optimal use of the Internet and email. But to them, he adds, as one of his basic human needs, the face-to-face interaction with his potential or already existing partners. His joint work with 134 authors from 29 countries illustrate clearly his synergetic capacity. He also has a need to spread his ideas beyond the community of specialists in his fields, to address a public, to argue, to face controversies. He enjoys interaction and you can see in his eyes his pleasure in doing research; his teaching is an organic part of his creativity process. He likes to point out hidden aspects, unexpected things, to reveal delicate points requiring further investigation. He alternately doubts and wonders during his oral presentations; you really feel that science is for him a great opportunity for satisfaction and joy. Using the computer presentation facilities, he does not become their victim, as it often happens: he knows how to remain alive behind the contemporary technology.

All these things are, to a large extent, mirrored in those parts of his curriculum vitae, usually considered, by the university bureaucracy, as secondary, if not negligible. I refer to sections such as Varia, Popular articles, Demos, Web sites, and to citations of similar types. You discover his capacity to reveal the meaning of mathematical and computational facts, their philosophical and artistic face. In order to realise his deep consciousness of belonging to a community of scholars and his capacity to serve and organise research—please stop a moment and think what does it mean to edit 28 collective books at international publishing houses and 33 special issues of international academic journals; what does it mean to be selected as a member of 84 Program Committees.

By all his accomplishments, Professor Cristian Calude is recognised today as a remarkable scholar and professor, as an intellectual with a wide cultural horizon, bridging mathematics, logic and computer science with physics, biology, philosophy and art. His evolution has followed the rhythm of his time, his interests focused at each moment on some of the hottest scientific issues. He is reaching the age of sixty with the same freshness of mind he showed forty years ago, at the beginning of his scientific career. Contemplating his personality at this anniversary moment gives a real satisfaction.

Calude as Father of One of the Computer Science Journals*

Hermann Maurer

Graz University of Technology, Graz, Austria
hmaurer@iicm.edu

Abstract. The Journal of Universal Computer Science ("universal" to indicate that no area is excluded) was one of the first (if not **the** first) journal that published refereed papers on the Internet, yet also provided a printed "archive" version of all papers published during a year after the end of that year. It is also one of the few truly open access journals in computer science: both publication and access is free for everyone. In this paper we describe how the journal started, what problems it was confronted with, and how they were solved.

1 Introduction

The Internet was initially seen as tool for emails and for allowing the dissemination of scientific publications rapidly, much faster than using traditional refereed and printed journals.

Before the Web took off in a large way, three systems to handle information on the Web had been developed: (1) The WWW by a group of four at CERN (where only the alphabetically first, although by no means the most important one, is still mentioned—as consequence that in scientific contributions in computer science we tend to mention authors alphabetically, independent of who was most important); (2) The Gopher System, developed by Marc McCahill at the University of Minnesota and (3) The Hyper-G (later Hyperwave) System developed by the author and his team in Graz. First publications on those three systems appeared in 1990, and first prototypes where available in 1991. Gopher was leading for a while, until a graphic browser "Mosaic" was developed

* This paper is dedicated to Cris Calude on the occasion of his 60th birthday. I got to know Cris in person when I took on the position of Full Professor for Computer Science at the University of Auckland in the early nineties. At that time Cris and his family had left their home country Romania a short while ago. It was a pleasure to make friends with such a multi-talented scientist and generous person. We have been good friends ever since. If I have two wishes I have one for him: to continue to enjoy his life and to contribute to science in essential ways as he has done in the past many times (and one facet not many will know about is what is reported in this paper); and I have one wish for myself: that Cris and I meet more often. However, let me also mention one historic bit I am proud of: when I came to Auckland, Cris was still in an unacceptable junior position. I did my bit to help change this, with Cris being offered an endowed chair soon thereafter.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 20–30, 2012.

for WWW at the University of Illinois. Now Gopher started to disappear, and Hyperwave withdrew to niche applications like Intranet or scientific publishing. During my tenure at Auckland I had many chats with Cris Calude and he understood immediately that Hyperwave was a powerful tool to handle large amounts of data. In 1992 he suddenly surprised me with the idea to use Hyperwave as basis for an electronic journal: "You know", he told me, "Arto Salomaa is willing to join us as foundation editor. So why don't you show how good Hyperwave is, by allowing the publication of refereed material in various formats, by choosing a new way of refereeing, by allowing the addition of comments to contributions, by providing full text search and search by various categories (like title, author, ...), and by preparing a printed version at the end of each year. And then let us discuss how over time we can add more and more innovative features to our journal." This is how *J.UCS*, the *Journal of Universal Computer Science* (meaning: covering all aspects of computer science) `http://www.JUCS.org` was born, given birth to by Cris who is top theorist, yet has his mind wide open for other issues whose significance he often realizes before the people deeply involved in technical details (as I was) do.

During my tenure as Professor at the University of Auckland my friend and college Cris Calude saw three interesting points with surprising clarity: (1) Why not use the Internet also for rapid dissemination of refereed material, if one can assure to speed up the refereeing process; (2) Once "published" in electronic form, no changes would be allowed to a paper (like in a printed version), but authors or readers would have the possibility to add comments (to correct typos, to add references, to clarify points in the paper, etc.); (3) To add "academic credibility", material would also have to be published in printed form, with ISSN or ISBN number, and would have to enter the list of high-quality journals by being indexed by ISI and similar institutions.

In discussion between the three of us: Cris Calude, Arto Salomaa and myself it became clear that many additional powerful features could be integrated in a journal as planned. It was me (who would after all be responsible for the implementation) who had to argue to first get started on the basis of the three points mentioned, and postpone further developments depending on the success of *J.UCS*.

After all, there was more to solve than just the implementation!

2 The Start of *J.UCS*

To get *J.UCS* [J.UCS 2012] off the ground (in 1994!) each of the three issues mentioned had to be solved, and in doing so new problems surfaced.

2.1 Issue 1 (The Refereeing Process)

It was clear that to have a high-quality journal we would need 2–3 referees for each paper submitted. Covering all areas of computer science (using the ACM Categories with permission of ACM) required well over 200 referees: we started

with 176 in 1994, and are now at over 335, and growing, despite the sad fact that we have lost a few due to health problems or old age. Due to the special involvement of referees we call them members of the editorial board: they may delegate actual refereeing to colleagues, but are responsible for the quality of reports obtained.

The real challenge was, however, how to speed up the refereeing process. We decided to use an entirely new approach: rather than leaving it up to the editors-in-chief or the managing editor to select for each paper submitted suitable referees (members of the editorial board) we would send the abstract of each paper to all members of the editorial board and members would choose which one they will review. Thus, no member of the editorial board will ever receive a paper for refereeing that is a total misfit, nor will a member receive a paper when too busy to do the refereeing fast, with a maximum of four weeks allowed. This does indeed shorten the refereeing process dramatically if three members of the editorial board are willing to look at a paper. Unfortunately, there are still papers were referees are not found "automatically" and the managing editor together with the assistant to the managing editor has to intervene, potentially prolonging the reviewing process quite a bit. Basically, it is by now clear that some 300 members in the editorial board would suffice, if all were reasonably active. Not surprisingly, only a small percentage falls in this category, however. Thus, a further expansion of referees to catch enough active ones seems necessary.

The system of "voluntary reviewing" with a large number of members on the editorial board does create one potential problem: an author may contact three friends on the editorial board and ask them for a favourable review. While a conflict of interest can never be eliminated even with traditional reviewing systems, the "danger" in our system is clearly higher, hence we had to invent new techniques for handling such situations. This will be described in Section 3 of this paper.

Another issue was which kinds of file formats we would allow for submission, and which formats we would use for publishing. PDF was not as omnipresent as it is today, so we had to be more lenient: we allowed a number of different file formats for submission and published each paper in three formats: HTML, Postscript and PDF. We have retained the last two, but given up on HTML simply because formulae in HTML are a real headache (they have to be inserted as in-line images). As far as submissions are concerned, we are still very liberal and accept most common formats like PostScript, PDF, MS Word, RTF and LaTeX.

2.2 Issue 2 (Freezing Contributions But Allowing Comments)

It was clearly desirable that an accepted and published paper should be "frozen", i.e. no later changes should be possible, much like in a printed journal. The temptation was not to be quite strict, but to allow corrections of typos or such. However, our decision was and remains that even such minor corrections are not possible "after the fact" in *J.UCS*, providing a source of stable, high-quality contributions.

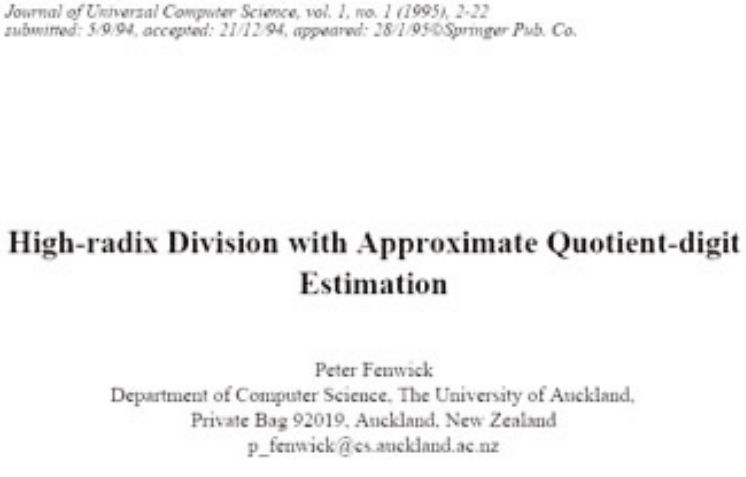
However, in printed journals it is sometimes possible to correct errors in a later issue in a specific column "corrections" or such. In J.UCS we decided on a more modern way (indeed we believe it was the first electronic journal to allow this): anyone can add comments after a paper: the author can add corrections, readers can ask questions or voice criticism, the author can reply by writing another comment, etc. Thus, *J.UCS* allows an arbitrary intensive discussion of papers already published. In the past this feature has not been widely used, somewhat to our disappointment: we believe that in 1994 when the journal started the user community was not yet used to what later would be called the "interactive Web" or Web 2.0.

We remain proud to be one of the pioneers in this area.

2.3 Issue 3 (The Printed Version)

When starting *J.UCS* the idea was, see [Calude at all 1994], to first of all publish a CD at the end of every year with all papers of the preceding year, and to have not one server for J.UCS but a number of mirror servers, so that access would be fast in all parts of the world. We ended up with a printed version of all papers of one year, the "archival version", rather than producing CDs. Further, the mirroring concept turned out to be superfluous, with the increase of bandwidth world-wide.

The challenge remained to have papers published electronically with exactly the same pagination and format as they would later appear in printed version. Indeed if you look at the PDF version of the very first paper in the very first volume published in 1995 you find what is shown in Fig. 1.:


Journal of Universal Computer Science, vol. 1, no. 1 (1995), 2-22
submitted: 5/9/94, accepted: 21/12/94, appeared: 28/1/95 ©Springer Pub. Co.

High-radix Division with Approximate Quotient-digit Estimation

Peter Fenwick
Department of Computer Science, The University of Auckland,
Private Bag 92019, Auckland, New Zealand
p_fenwick@cs.auckland.ac.nz

Fig. 1. Sample *J.UCS* front page

Thus, printing this paper when it has appeared electronically looks exactly like a reprint from the printed version of the journal which is likely to appear only up to a year later: note that the above paper appeared January 1995, but the volume containing all papers of 1995 only appeared in April 1996!

We believe that this idea to have a fast electronic publication which is fully citable and indeed also appears in printed form is one reason why *J.UCS* became as successful as it is (for details see Section 5). It is only fair to mention that we had one other main advantage going for us: Springer was sponsoring J.UCS under one condition: *J.UCS* would not be entirely free, but there would be an annual subscription fee of $ 100,00 per university, i.e. for just $ 100,00 arbitrarily many members of the University could read and print each paper in the journal. Although this was not quite the idea of open access, it was close enough (it seemed to us) and did help tremendously to make *J.UCS* better known in computer science circles. When we later decided to drop the subscription fee Springer did generously permit us to do so, signed over all remaining rights to us (at that point already Graz University of Technology) but did not support/sponsor it any more.

The paper on electronic publishing [Odlyzko 1994] is still excellent reading: it is fairly accurate in forecasting what would happen to scientific journals (many would go electronic by 2010) and that the costs, even with copy editing but cutting down on reviewing will be low. We quote:

> My general conclusion is that it should be possible to publish scholarly journals electronically for well under $1.000,00 per article, and probably under $500,00, without losing much quality.

It is interesting to note that in *J.UCS* we initially published some 50–80 papers a year, and now are up to about 150. Multiplying this by $500,00 gives $75.000,00 per year, or some €60.000,00. And this is indeed the budget that we need for *J.UCS*. Of this, close to €20.000,00 is used for continued improvement and for formatting and printing; the remainder is used for running the server, but mainly for one editorial assistant. During the first years, the somewhat smaller costs where covered by Graz University of Technology alone, then later the Institute for Knowledge Management stepped in, still later UNIMAS in Kuching. And as of 2012 it is now a consortium of 8 groups, see [J.UCS 2012] that are jointly carrying the total cost that has gone up a bit to around €70.000. This allows the operation of *J.UCS* as open access journal: no fee for submission or publication, no fee for reading.

There are two further points to be mentioned: First, [Odlyzko 1994] was already recommending 18 years ago to do away with reviewing, replacing it by just copy-editing and comments/dialogues that would follow after publication, possibly leading to revisions, etc. *J.UCS* is more conservative and is still sticking to the refereeing process, although a number of collaborative undertakings on the Web based on the idea "Wisdom of Crowds" [Surowiecki et al 2005] suggest that a journal on that basis might be quite feasible, and indeed require fewer resources than a refereed journal. However, such journals would not be

recognized by the more important citation indices at this point in time, and this might be an obstacle to obtain some top papers. Second, there are many "open access" journals around today. However, in most cases (*J.UCS* is one of the few exceptions) reading is free, but authors have to pay for papers accepted, usually hundreds of US dollars, or even more. We are violently opposed to this approach: the temptation to accept a moderately good paper just because it will help financially is likely to reduce the overall quality of such journals.

3 Problems and Partial Solutions

In this section we want to address some of the problems we have encountered in *J.UCS*, and some attempts how to deal with them.

As explained in the previous section the already large number of members of editorial board (some 340 at the time of writing) is still not enough to assure short refereeing times, simply because only a fraction of potential candidates have time and are willing to take on a reviewing job at any particular moment.

This has a "trivial" consequence concerning the submission system: sending the abstract of an incoming paper to all members of the editorial board is not realistic any more. Rather, a match between the area of the paper submitted and the area of expertise/ knowledge of reviewers has to be established and only "suitable" editors are notified that a new submission in their area has been received. Although this does sound easy we will argue below that unfortunately it is not.

Further, with the very large number of members of the editorial board a more serious problem arises: how does one avoid a paper submitted being reviewed only or mainly by friends of the author(s)?

Of course conflict of interest (COI) situation do arise also in other areas like when employing or promoting a person on the basis of letters of recommendations, or such. Thus, we have tried to pose the following general problem: given two persons A and B, can one—by using information on the Web (including social networks)—make an educated guess whether A and B are friends? We have not finished exploring all avenues in this direction but we have tried to tackle the problem by (a) considering closeness of location (surely if a paper comes from a certain department and all reviewers come from the same department one cannot really expect objective reviews) and by (b) considering closeness of persons on the basis of citations, co-authorship, etc.

Let us discuss these two issues in turn. We have tackled (a) by implementing mash-ups. Indeed there are three types: (i) those that show where papers (over a certain period in a certain area) come from; (ii) those that show where editors come from; and (iii) where editors of a certain paper come from.

Note that while (i) and (ii) are openly available, (iii) is only available to the managing editor in chief. It is clear that (iii) allows the managing editor to check if there is a potential COI situation due to co-location. That we implemented also (i) and (ii) has two reasons: (ii) allows editors to check if enough editors in some field are available. It also allows authors to check if enough editors

are available that are considered "objective" for whatever reason. Like, would a Pakistani feel comfortable if all referees are from India, an Arab if all are from Israel? Mash-ups under (i) allow the managing editor in chief to find out in which part of the world certain areas are considered particularly important, and how the importance of areas might change over time. Or at least this was the idea behind those mash-ups: although the implementation has worked flawlessly now for years, the results are often less helpful than expected. Let us explain this by means of a particular example (which is, fortunately, not typical for most areas).

Fig. 2 shows all editors that have indicated that their area is knowledge management. Note that not a single person in the USA shows up. This allows three interpretations: (i) it is an area not of interest in the USA; (ii) *J.UCS* has failed to attract editors in this field in the USA (iii) something else is amiss.

Fig. 2. A mash-up showing editors in the area "Knowledge management"

Looking at Fig. 3 the situation becomes even more puzzling: the only papers that where written in the category knowledge management come from the USA! This is particularly surprising since one of the major conferences in knowledge management of which top papers always appear in *J.UCS* is [I-KNOW 2011]. Using the search function of *J.UCS* we can identify 120 papers that contain the word "knowledge"!

Fig. 3. A mash-up showing all papers that have been written with category knowledge management specified

Putting this together means: Papers that should be tagged as "knowledge management" are not tagged that way. Editors that should indicate that their area includes knowledge management forget to do so. Thus categories/indices /tagging have to be more carefully studied to turn mash-ups into universally usable tools. However, the example given is a bit arbitrary: knowledge management is not an ACM category and not used as term very widely. Areas such as tagging, knowledge discovery, data mining, e-Learning etc. might all be considered as parts of knowledge management, but are often not considered in this context both by editors and authors. Thus, the mash-up concept does indeed yield valuable results in other areas, but to use it in general, good ontologies or at least synonym dictionaries will still have to be developed.

The second approach to COI—in the sense that a referee may be a friend of an author whose paper is under review—has been studied in depth in the thesis of [Khan 2011a]. The most relevant work in our context is published in [Khan 2011b]. Without going into detail it is clear that persons who have co-authored a paper are likely to know each other; if A often cites B, and B often cites A, then even if they don't know each other they seem to be on the same wavelength. If A and B have never co-authored a paper but both are often cited together in papers, again a certain relationship is likely.

We believe that based on information on the Web the question "are A and B likely to be friends" can still be studied from many additional other angles (that might be valuable for journal editors, but also in case of promotions, or hiring staff, etc.). Indeed, even the (more difficult) question "are A and B likely to be enemies" can probably be tackled using material on the Web to some extent. One point seems to be important: answers obtained in this fashion can never be trusted, and hence should not be "black and white". I.e. systems should report a percentage of likelihood such as "A and B are friends with a likelihood of 92%".

4 Additional Functions in *J.UCS*

We have noted earlier that papers once published in *J.UCS* cannot be changed any more. However, notes (comments) can be added to an arbitrary extent. This has led us early on to the idea of "Links into the future" (© H. Maurer). The basic idea is most easily explained by means of an example. Suppose a paper was written in the year 1998. A paper in 2006 extends some results of the earlier one, hence will have the 1998 paper in the list of references. So why not add as comment to the *J.UCS* paper of 1998 a remark that further work in this area has been done in another *J.UCS* paper in 2006, with a direct link to it, i.e. a link from 1998 to 2006, i.e. a link into the future?

The challenge of this is to also have links to papers in the future that have appeared elsewhere, or more generally, to even link to related papers written later, even if they do not explicitly refer to the earlier *J.UCS* paper (but do so implicitly by dealing with the same algorithm; or by pointing to a paper that itself points to the earlier *J.UCS* paper, etc.).

Over the years this basic idea has been expanded more and more, and there is still much room for further improvement. The classical paper in this area is probably [Afzal et al., 2007], although earlier and/or simpler versions exist, such as [Maurer, 2001], [Krottmaier, 2003], or [Dreher et al., 2008].

One of the powerful functions of *J.UCS* is searching. It can be restricted to a certain time-area, to the names of authors in titles or in full-text, allows logical connectives, etc.

5 Outlook

J.UCS is more than a journal that has some 85.000 distinct readers, and over 650.000 PDF files downloaded a year, and a five-year impact factor close to 0.8. It is one of the few journals that is truly open access: no charge for publishing, no charge for reading. It has sophisticated functions like complex searching, links into the future, a novel refereeing system, and is in the process of further expansion by allowing readers to find for each paper published "similar ones", authors working in "a similar area", profiles of such authors, and much more.

It also has to come to grips with an overwhelming flood of incoming papers, simply because many young researchers cannot afford to pay for a publication, nor can they accept long waiting times until their paper is or is not published.

We have indicated that *J.UCS* needs further high quality and active editors, yet we cannot send to all of them all abstracts of papers received, nor can we easily decide what to send to whom since the categories used by editors and by authors often disagree more than expected, as we showed in the discussion of Fig. 2 and Fig. 3.

J.UCS may have to switch to a two stage submission process: If a paper comes in from an author who is not yet an established scientist (measured by citation count or Hirsh index?) maybe we have to put the paper into an area for public comments, and let the community decide whether a serious reviewing should take place or not. All readers who are fans of Wikipedia, of crowd sourcing, and of the Wisdom of the Crowds [Surowiecki et al 2007] will be delighted to read this and will be disappointed to learn that it does not work in general: a great idea by a young scientist put to the public in an unintelligible way may well be snapped up by someone, and turned into a top-notch paper, never giving credit to the one who had the original idea.

Thus, *J.UCS* is by now more than a well accepted journal (the five year impact factor is now close to 0.8, quite high for computer science); it is also a great research and publishing project. I want to thank Cris Calude and Arto Salomaa for their pioneering vision that has not just provided me, but many, with much food for thought and further research. I wish all members of the *J.UCS* team continuing success: I have headed that team till end of 2011, but have handed it over to the new group of editors-in-chief, in particular the managing editor in chief as mentioned in [J.UCS 2012].

References

[Andrews et al., 1995] Andrews, K., Kappe, F., Maurer, H.: The Hyper-G Network Information System. Journal of Universal Computer Science, J.UCS 1(4), 206–220 (1995)

[Afzal et al., 2007] Afzal, M., Maurer, H.: Creating links to the future. Journal of Universal Computer Science, J.UCS 13(9), 1234–1245 (2007)

[Berners-Lee et al. 1994] Berners-Lee, T., Cailliau, R., Luotonen, A., Nielsen, H.F., Secret, A.: The World Wide Web. Communications of the ACM 37(8), 76–82 (1994)

[Calude et al., 1994] Calude, C., Maurer, H., Salomaa, A.: Journal of Universal Computer Science, J.UCS 0(0), 109–116 (1994)

[Dreher et al., 2008] Dreher, H., Krottmaier, H., Maurer, H.: What we expect from digital libraries. J.UCS 10(9), 1110–1122 (2004)

[I-KNOW 2011] http://www.i-know.at

[J.UCS 2012] www.jucs.org/www.J.UCS.org

[Khan et al., 2008] Khan, M.S., Kulathuramaiyer, N., Maurer, H.: Applications of mash-ups for a digital journal. Journal of Universal Computer Science, J.UCS 14(10), 1695–1716 (2008)

[Khan 2011a] Khan, M.S.: Aspects of Content Quality, Management in Digital Libraries of Scholarly Publications, Ph.D. Thesis Graz University of Technology (2011)

[Khan 2011b] Khan, M.S.: Exploring citations for conflict of interest detection in peer-review system. International Journal of Computer Information Systems and Industrial Management Applications 3 (to appear, 2011)

[Krottmaier, 2003] Krottmaier, H.: Links To The Future. Journal of Digital Information Management 1(1), 3–8 (2003)

[Maurer, 2001] Maurer, H.: Beyond Classical Digital Libraries. In: Chen, D.-C. (ed.) Proc. of Conference on Global Digital Library Development, pp. 165–173. Tsinghua University Press, Beijing (2001)

[McCahill et al., 1995] McCahill, M.P., Anklesaria, F.X.: Evolution Of Internet Gopher. Journal of Universal Computer Science, J.UCS 1(4), 235–246 (1995)

[Odlyzko 1994] Odlyzko, A.M.: Tragic loss or good riddance? The impending demise of traditional scholarly journals. J.UCS 0(0), 3–53 (1994)

[Surowiecki et al., 2007] Surowiecki, J.: The Wisdom of Crowds. Anchor Pocket Books (2005)

Random Semicomputable Reals Revisited

Laurent Bienvenu[1] and Alexander Shen[2,⋆]

[1] LIAFA, CNRS & Université Paris Diderot,
Paris 7, Case 7014, 75205, Paris Cedex 13, France
Laurent.Bienvenu@liafa.jussieu.fr
[2] LIRMM, CNRS & Université Montpellier 2,
161 rue Ada, 34095 Montpellier Cedex 5,
France, on leave from IITP RAS, Bolshoy Karetny, 19, Moscow
alexander.shen@lirmm.fr, sasha.shen@gmail.com

To Cristian Calude on the occasion on his 60th birthday

Abstract. The aim of this expository paper is to present a nice series of results, obtained in the papers of Chaitin [3], Solovay [8], Calude et al. [2], Kučera and Slaman [5]. This joint effort led to a full characterization of lower semicomputable random reals, both as those that can be expressed as a "Chaitin Omega" and those that are maximal for the Solovay reducibility. The original proofs were somewhat involved; in this paper, we present these results in an elementary way, in particular requiring only basic knowledge of algorithmic randomness. We add also several simple observations relating lower semicomputable random reals and busy beaver functions.

1 Lower Semicomputable Reals and the $\preceq_1$-Relation

Recall that a real number α is *computable* if there is a computable sequence of rationals a_n that converges to α computably: for a given $\varepsilon > 0$ one may compute N such that $|a_n - \alpha| \leqslant \varepsilon$ for all $n > N$. (One can assume without loss of generality that the a_n are increasing.)

A weaker property is lower semicomputability. A real number α is *lower semicomputable* if it is a limit of a computable increasing sequence of rational numbers. Such a sequence is called *approximation of α from below* in the sequel.

Equivalent definition: α is lower semicomputable if the set of all rational numbers less than α is enumerable. One more reformulation: if $\alpha = \sum_{i \geqslant 0} d_i$ where d_i is computable series of rational numbers, and all d_i with $i > 0$ are non-negative. (We let d_0 be negative, since lower semicomputable α can be negative.)

It is easy to see that α is computable if and only if α and $-\alpha$ are lower semicomputable. There exist lower semicomputable but non-computable reals. Corresponding sequences of rational numbers have non-computable convergence. (Recall that convergence of a sequence a_i to some α means that for every rational $\varepsilon > 0$ there exist some integer N such that $|a_i - \alpha| < \varepsilon$ as soon as $i > N$. Noncomputable convergence means that there is no algorithm that produces some N with this property given ε.)

⋆ Supported by NAFIT ANR-08-EMER-008-01, RFBR 0901-00709-a grants.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 31–45, 2012.

We want to classify computable sequences according to their convergence speed and formalize the intuitive idea "one sequence converges better (i.e., not worse) than the other one".

Definition 1. *Let $a_i \to \alpha$ and $b_j \to \beta$ be two computable strictly increasing sequences converging to lower semicomputable α and β (approximations of α and β from below). We say that $a_n \to \alpha$ converges "better" (not worse) than $b_n \to \beta$ if there exists a total computable function h such that*

$$\alpha - a_{h(i)} \leqslant \beta - b_i$$

for every i.

In other terms, we require that for each term of the second sequence one may algorithmically find a term of the first one that approaches the limit as close as the given term of the second sequence. Note that this relation is transitive (take the composition of two reducing functions).

In fact, the choice of specific sequences that approximate α and β is irrelevant: *any two increasing computable sequences of rational numbers that have the same limit, are equivalent with respect to this quasi-ordering*. Indeed, we can just wait to get a term of a second sequence that exceeds a given term of the first one. We can thus set the following definition.

Definition 2. *Let α and β be two lower semicomputable reals, and let (a_n), (b_n) be approximations of α and β respectively. If (a_n) converges better than (b_n), we write $\alpha \preceq_1 \beta$ (by the above paragraph, this does not depend on the particular approximations we chose).*

This definition can be reformulated in different ways. First, we can eliminate sequences from the defintion and say that $\alpha \preceq_1 \beta$ if there exists a partial computable function φ defined on all rational numbers $r < \beta$ such that

$$\varphi(r) < \alpha \text{ and } \alpha - \varphi(r) \leqslant \beta - r$$

for all of them. Below, we refer to φ as the *reduction function*.

The following lemma is yet another characterization of the order (perhaps less intuitive but useful).

Lemma 1. *$\alpha \preceq_1 \beta$ if and only if $\beta - \alpha$ is lower semicomputable (or said otherwise, if and only if $\beta = \alpha + \rho$ for some lower semicomputable real ρ).*

Proof. To show the equivalence, note first that *for every two lower semicomputable reals α and ρ we have $\alpha \preceq_1 \alpha + \rho$*. Indeed, consider approximations (a_n) to α, (r_n) to ρ. Now, given a rational $s < \alpha + \rho$, we wait for a stage n such that $a_n + r_n > s$. Setting $\varphi(s) = a_n$, it is easy to check that φ is a suitable reduction function witnessing $\alpha \preceq_1 \alpha + \rho$.

It remains to prove the reverse implication: *if $\alpha \preceq_1 \beta$ then $\rho = \beta - \alpha$ is lower semicomputable*. Indeed, if (b_n) is a computable approximation (from below) of β and φ is

the reduction function that witnesses $\alpha \preceq_1 \beta$, then all terms $b_n - \varphi(b_n)$ are less than or equal to $\beta - \alpha$ and converge to $\beta - \alpha$. (The sequence $b_n - \varphi(b_n)$ may not be increasing, but still its limit is lower semicomputable, since all its terms do not exceed the limit, and we may replace nth term by the maximum of the first n terms.) □

A special case of this lemma: let $\sum u_i$ and $\sum v_i$ be computable series with non-negative rational terms (for $i > 0$; terms u_0 and v_0 are starting points and may be negative) that converge to (lower semicomputable) α and β. If $u_i \leqslant v_i$ for all $i > 0$, then $\alpha \preceq_1 \beta$, since $\beta - \alpha = \sum_i (v_i - u_i)$ is lower semicomputable.

The reverse statement is also true: if $\alpha \preceq_1 \beta$, one can find computable series $\sum u_i = \alpha$ and $\sum v_i = \beta$ with these properties ($0 \leqslant u_i \leqslant v_i$ for $i > 0$). Indeed, $\beta = \alpha + \rho$ for lower semicomputable ρ; take $\alpha = \sum u_i$ and $\rho = \sum r_i$ and let $v_i = u_i + r_i$.

In fact, a stronger statement is also true; each of the series can be chosen in an arbitrary way. We have already seen how to choose v_i when u_i are given. The other direction: assume that $\alpha \preceq_1 \beta = \sum v_i$ for some $v_i \geqslant 0$. We need a decomposition $\alpha = \sum u_i$ where $u_i \geqslant 0$ and $u_i \leqslant v_i$ for $i > 0$. Indeed, we can construct u_i sequentially using the following invariant: the current approximation $A = \sum_{j<i} u_j$ to α should be below α and at least as close (to α) as the current approximation $B = \sum_{j<i} v_j$ (to β). Initially we choose u_0 applying reduction function to v_0. When the current approximation becomes $B' = B + v_i$, we apply reduction function to get A' which is at least as close to α as B' is to β. Then there are several cases:

(1) if $A' < A$, we let $u_i = 0$, and the next approximation is A (it is close enough by assumption);

(2) if $A \leqslant A' \leqslant A + v_i$, we let $u_i = A' - A$; the condition guarantees that $u_i \leqslant v_i$;

(3) finally, if $A' > A + v_i$, we let $u_i = v_i$ (the invariant remains valid since the distances to α and β are decreased by the same amount).

2 The Solovay Reducibility and Complete Reals

Let α be a lower semicomputable but not computable real. By the results of the previous section, one has

$$\alpha \preceq_1 2\alpha \preceq_1 3\alpha \preceq_1 \ldots$$

because for all k the difference $(k+1)\alpha - k\alpha = \alpha$ is lower semicomputable (so Lemma 1 applies). The reverse relations are not true, because $k\alpha - (k+1)\alpha = -\alpha$ is not lower semicomputable (if it were, then α would be computable).

One may argue that this relation is therefore a bit too sharp. For example, α and 2α have essentially the same binary expansion (just shifted by one position), so one may want α and 2α to be equivalent. In other words, one may look for a less fine-grained relation. A natural candidate for this is *Solovay reducibility*.

Definition 3 (Solovay reducibility). *We say that $\alpha \preceq \beta$ if $\alpha \preceq_1 c\beta$ for some positive integer $c > 0$.*

(A convenient notation: we say, for some positive rational c, that $\alpha \preceq_c \beta$ if $\alpha \preceq_1 c\beta$. Then $\alpha \preceq \beta$ if $\alpha \preceq_c \beta$ for some c.)

Like for lower semicomputable semimeasures in algorithmic information theory (see, e.g., [7]), one can easily prove the existence of maximal elements [8].

Theorem 1. *There exists a $\preceq$-biggest lower semicomputable real.*

Proof. Indeed, we can enumerate all lower semicomputable reals α_i in $[0,1]$ and then take their sum $\alpha = \sum w_i\alpha_i$ with computable positive weights w_i such that $\sum w_i$ converges. This α can be represented as $w_i\alpha_i$ plus some lower semicomputable real, so $\alpha_i \preceq_1 (1/w_i)\alpha$. □

The biggest elements for the $\preceq$-preorder are also called *(Solovay) complete* lower semicomputable reals. They have an alternative description [8,2]:

Theorem 2. *Complete semicomputable reals in $[0,1]$ are sums of universal semimeasures on $\mathbb{N}$ and vice versa.*

Recall (see [7] for details) that lower semicomputable semimeasures on $\mathbb{N}$ are lower semicomputable functions $m\colon \mathbb{N} \to \mathbb{R}$ with non-negative values such that $\sum_i m(i) \leqslant 1$. (For a function m lower semicomputability means that $m(i)$ is lower semicomputable uniformly in i: there is an algorithm that gets i as input and produces an increasing sequence of rationals that converges to $m(i)$.) Universal semimeasures are the maximal (up to a constant factor) lower semicomputable semimeasures.

Proof. Any lower semicomputable real α is a sum of a computable series of rationals; this series (up to a constant factor that does not matter due to the definition of the Solovay reducibility) is bounded by a universal semimeasure. The difference between the upper bound and the series itself is a lower semicomputable semimeasure, and therefore α is reducible to the sum of the universal semimeasure.

We have shown that sums of universal semimeasures are complete. On the other hand, let α be a Solovay complete real in $[0,1]$. We need to show that α is a sum of some universal semimeasure. Let us start with arbitrary universal semimeasure $m(i)$. The sum $\sum m(i)$ is lower semicomputable and therefore $\sum m(i) \preceq_1 c\alpha$, so $\alpha = \sum m(i)/c + \tau$ for some integer $c > 0$ and some lower semicomputable τ. Dividing m by c and then adding τ to one of the values, we get a universal semimeasure with sum α. □

Chaitin denoted the sum of a universal semimeasure by Ω. Since there is no such thing as *the* universal semimeasure, it is better to speak about *Ω-reals* defined as sums of universal semimeasures. We have shown therefore that the class of Ω-reals coincides with the class of Solovay complete lower semicomputable reals in $[0,1]$.

It turns out that this class has one more characterization [3,2,5]:

Theorem 3. *A lower semicomputable real is complete if and only if it is Martin-Löf random.*

(See, e.g., [7] for the definition of Martin-Löf randomness.) We provide the proof of this result below, starting with one direction in the next section 3 and finishing the other direction in section 5.

3 Complete Lower Semicomputable Reals Are Random

The fact that lower semicomputable reals are random, is Chaitin's theorem (randomness of Ω). It is usually proved by using complexity characterization of randomness.

However, there is a direct argument that does not involves complexity (it is in the footnote in Levin's "Forbidden information" paper [6]; this footnote compressed the most important facts about lower semicomputable random reals into few lines!).

First, we prove that *there exists a lower semicomputable random real*. For that we consider an effectively open set U of measure less than (say) $1/2$ that covers all non-random reals in $[0,1]$. (The definition of Martin-Löf randomness guarantees that for every $\varepsilon > 0$ one can find an effectively open set that has measure less than ε and covers all non-random reals. We need only one such set for some $\varepsilon < 1$, say, $\varepsilon = 1/2$.) Then take the minimal element α in a closed set $[0,1] \setminus U$. This number is random (by definition) and lower semicomputable: compactness implies that any segment $[0,r]$ with rational $r < \alpha$ is covered by finitely many intervals of U and thus all such r's can be enumerated.

Second, we prove that *randomness is upward-closed*: if $\alpha \preceq \beta$ and α is random, then β is random. We may assume without loss of generality that $\alpha \preceq_1 \beta$ (randomness does not change if we multiply a real by a rational factor).

So let $b_i \to \beta$ be a computable increasing sequence of rational numbers that converges to β. Assume that somebody gives us (in parallel with b_i) a sequence of rational intervals and guarantees that one of them covers β. How to transform it into a sequence of intervals that covers α (i.e., one of the intervals covers α) and has the same (or smaller) total length? If an interval appears that is entirely on the left of the current approximation b_i, it can be ignored (since it cannot cover β anyway). If the interval is entirely on the right of b_i, it can be postponed until the current approximation b_j enters it (this may happen or not, in the latter case the interval does not cover β). If the interval contains b_i, we can convert it into the interval of the same length that starts at a_j, where a_j is a rational approximation to α that has the same or better precision as b_i (as an approximation to β): if β is in the original interval, α is in the converted interval.

So randomness is upward-closed and therefore complete lower semicomputable reals are random.

Remark. The second part can be reformulated: if α and β are lower semicomputable reals and at least one of them is random, then the sum $\alpha + \beta$ is random, too. The reverse is also true: if both α and β are non-random, then $\alpha + \beta$ is not random. (We will see later different proofs of this statement.)

4 Randomness and Prediction Game

Before proving the reverse implication, let us make a digression and look more closely at the last argument. Consider the following game: an observer watches an increasing sequence of rationals (given one by one) and from time to time makes predictions of the following type: "the sequence will never increase by more than δ" (compared to its current value). Here δ is some non-negative rational. The observer wins this game if (1) one of the predictions remains true forever; (2) the sum of all numbers δ used in the predictions is small (less that some rational $\varepsilon > 0$ which is given to the observer in advance).

It is not required that at any moment a valid prediction exists, though one could guarantee this by making predictions with zero or very small (and decreasing fast) δ at

each step. Note also that every prediction can be safely postponed, so we may assume that the next prediction is made only if the previous one becomes invalid. Then at any moment there is only one valid prediction.

Theorem 4. *Let a_i be a computable increasing sequence of rational numbers that converges to some* (*lower semicomputable*) *real α. The observer has a computable winning strategy in the game if and only if α is not random.*

Proof. A computable winning strategy gives us a computable sequence of prediction intervals of small total measure and guarantees that one of these (closed) intervals contains α. On the other hand, having a sequence of intervals that covers α and has small total measure, we may use it for predictions. To make the prediction, we wait until the current approximation a_i gets into the already discovered part of the cover (this will happen since the limit is covered). Then for our prediction we use the maximal δ such that $(a_i, a_i + \delta)$ is covered completely at the moment, and then wait until this prediction becomes invalid. Then the same procedure is used again. At some point α is covered by some interval in the sequence and the current approximation enters this interval; the prediction made after this moment will remain valid forever. The total length of all prediction interval is bounded by the measure of the cover (the prediction intervals are disjoint and all are covered). □

A reformulation of the same observation that does not use game terminology:

Theorem 5. *Let a_i be a computable increasing sequence of rational numbers that converges to α. The number α is non-random if and only if for every rational $\varepsilon > 0$ one can effectively find a computable sequence $h_0, h_1, \ldots$ of non-negative rational numbers such that $\sum_i h_i < \varepsilon$ and $\alpha \leqslant a_i + h_i$ for some i.*

(Here the predictions h_i are made on every step; it does not matter since we may use zeros.)

There is a Solovay criterion of randomness (a constructive version of Borel–Cantelli lemma): a real number α is non-random if and only if there exists a computable sequence of intervals that have finite total measure and cover α infinitely many times. It can also be reformulated in the style of our previous theorem:

Theorem 6. *Let a_i be a computable increasing sequence of rational numbers that converges to α. The number α is non-random if and only if there exists a computable sequence $h_0, h_1, \ldots$ of non-negative rational numbers such that $\sum_i h_i < \infty$ and $\alpha \leqslant a_i + h_i$ for infinitely many i.*

Proof. If α is non-random, we apply the preceding result for $\varepsilon = 1, 1/2, 1/4, 1/8, \ldots$ and then add the resulting sequences (with shifts $0, 1, 2, \ldots$ to the right). Each of them provides one value of i such that $\alpha \leqslant a_i + h_i$, and these values cannot be bounded due to shifts. On the other hand, if $\alpha \leqslant a_i + h_i$ for infinitely many i, we get a sequence of intervals with finite sum of measures that covers α infinitely many times (technically, we should replace closed intervals by slightly bigger open intervals). It remains to use Solovay's criterion (or recall its proof: the effectively open set of points that are covered with multiplicity m has measure at most $O(1/m)$). □

The randomness criterion given in this section implies the following observation (which may look strange at first). Consider a sum of a computable series of positive rational numbers. *The randomness of the sum cannot change if all summands are changed by some $\Theta(1)$-factor*. Indeed, all h_i can be multiplied by a constant.

Now let us prove that *if α and β are non-random lower semicomputable reals, their sum $\alpha+\beta$ is non-random, too*. (See the discussion in the previous section). The natural idea to prove this is the following: make predictions in the games for α and β, and then take their sum as prediction for $\alpha+\beta$. But this simple argument does not work. The problem is that the same prediction for α can be combined with many predictions for β and therefore will be counted many times in the sum.

The solution is to make predictions for α and β of the same size. Let a_i and b_i be computable increasing sequences that converge to α and β. Since α and β are non-random, they are covered by sequences of intervals that have small total measure. To make a prediction for the sequence a_i+b_i (after the previous prediction became invalid) we wait until the current approximations a_i and b_i become covered by the intervals of those sequences. We take then the maximal h and k such that (a_i, a_i+h) and (b_i, b_i+k) are entirely covered (by the unions of currently appeared intervals). The prediction interval is declared to be $(a_i+b_i, a_i+b_i+\delta)$ where $\delta = 2\min(h,k)$.

Let us show that one of the predictions will remain valid forever. Indeed, the limit values α and β are covered by some intervals. These intervals appear in the sequences at some point and cover α and β with some neighborhoods, say, σ-neighborhoods. If the prediction is made after a_i and b_i enter these neighborhoods, δ is greater than 2σ and the prediction is final: a_i+b_i never increases more than by δ.

It remains to estimate the sum of all δs used during the prediction. It can be done using the following observation: when a prediction interval $(a_i+b_i, a_i+b_i+\delta)$ becomes invalid, this means that either a_i or b_i has increased by $\delta/2$ or more, so the total measure of the cover on the right of a_i and b_i has decreased at least by $\delta/2$. (Here we use that $(a_i, a_i+\delta/2)$ and $(b_i, b_i+\delta/2)$ are covered completely because $\delta/2$ does not exceed both h and k: it is important here that we take the minimum.)

Let us return to the criterion for randomness provided by Theorem 5. The condition for non-randomness given there can be weakened in two aspects: first, we can replace computable sequence by a semicomputable sequence; second, we can replace h_i by the entire tail $h_i+h_{i+1}+\ldots$ of the corresponding series:

Theorem 7. *Let a_i be an increasing computable sequence of rational numbers that converges to α. Assume that for every rational $\varepsilon>0$ one can effectively find a lower semicomputable sequence h_i of non-negative reals such that $\sum_i h_i < \varepsilon$ and $\alpha \leqslant a_i+h_i+h_{i+1}+\ldots$ for some i. Then α is not random.*

Proof. Assume that for every i there is a painter who get h_i units of paint and the instruction to paint the line starting at a_i, going to the right and skipping the parts already painted by other painters (but making no other gaps). (Since h_i is only semicomputable, the paint is provided incrementally.) The painted zone is an effective union of intervals of total measure $\sum_i h_i$. If $\alpha < a_i+h_i+h_{i+1}+\ldots$, then α is painted since we cannot use $h_i+h_{i+1}+\ldots$ paint starting between a_i and α (recall that all a_k are less than α) and not

crossing α. (In the condition we have $\leqslant$ instead of $<$, but this does not matter since we can increase all h_i to, say, twice their original value.) □

This result implies one more criterion of randomness for lower semicomputable reals:

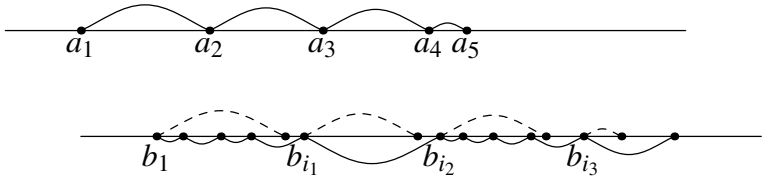
Theorem 8. *Let $\alpha = \sum d_i$ be a computable series of non-negative rational numbers. The number α is non-random if and only if for every $\varepsilon > 0$ one can effectively produce an enumerable set $W \subset \mathbb{N}$ of indices such that* (1) $\sum_{i \in W} d_i < \varepsilon$ *and* (2) W *is co-finite, i.e., contains all sufficiently large integers.*

Proof. If α is not random, it can be covered by intervals with arbitrarily small total measure. It remains to consider the set W of all i such that $(d_0 + \ldots + d_{i-1}, d_0 + \ldots + d_{i-1} + d_i)$ is entirely covered by one of those intervals. In the other direction the statement is a direct consequence of Theorem 7, just let $a_i = d_0 + \ldots + d_{i-1}$ and $h_i = d_i$ for $i \in W$ (and $h_i = 0$ for $i \notin W$). □

This result shows again that the sum of two non-random lower semicomputable reals is not random (take the intersection of two sets W_1 and W_2 provided by this criterion for each of the reals).

5 Random Lower Semicomputable Reals Are Complete

To prove the completeness of random lower semicomputable reals, let us start with the following remark. Consider two lower semicomputable reals α and β presented as limits of increasing computable sequences $a_i \to \alpha$ and $b_i \to \beta$. Let $h_i = a_{i+1} - a_i$ be the increases in the first sequence. We may use h_i to construct a strategy for the prediction game against the second sequence in the following way. We shift the interval $[a_1, a_2]$ to get the (closed) interval of the same length that starts at b_1. Then we wait until b_i at the right of this interval appears; let it be b_{i_1}. Then shift the interval $[a_2, a_3]$ to get the interval of the same length that starts at b_{i_1}; let b_{i_2} be the first b_i on the right of it, etc.



There are two possibilities: either

(1) the observer wins in the prediction game, i.e., some of the shifted intervals covers the rest of b_i and the next b_{i_k} is undefined, or

(2) this process continues indefinitely.

In the second case $\alpha \preceq_1 \beta$ since the difference $\beta - \alpha$ is represented as a sum of a computable series ("holes" between neighbor intervals; note that the endpoints of the shifted intervals also converge to β).

One of these two alternatives happens for arbitrary lower semicomputable reals α and β. Now assume that β is not Solovay complete; we need to prove that β is not

random. Since β is not complete, there exists some α such that $\alpha \not\preceq \beta$. In particular, $\alpha \not\preceq_1 \beta$. Therefore, for these α and β the second alternative is impossible, and the observer wins. In other terms, we get a computable sequence of (closed) intervals that covers β. Repeating the same argument for $\alpha/2$, $\alpha/4$,... (we know that $\alpha/c \not\preceq_1 \beta$ for every c, since $\alpha \not\preceq \beta$) we effectively get a cover of β with arbitrary small measure (since the sum of all h_i is bounded by a integer constant even being non-computable), therefore β is not random.

Remark. This argument probably gives some quantitative connection between randomness deficiency of a random lower semicomputable real and another parameter that can be called *completeness deficiency*. It can be defined as follows: fix some complete α and for every β consider the infimum of all c such that $\alpha \preceq_1 c\beta$.

6 Slow Convergence: Solovay Functions

We have seen several results of the following type: the limit of an increasing computable sequence of rationals is random if and only if the convergence is slow. In this section we provide one more result of this type.

Consider a computable converging series $\sum r_i$ of positive rational numbers. Note that r_i is bounded by $O(m_i)$ where $m\colon i \mapsto m_i$ is a universal semimeasure (m_i is also called *a priori probability* of integer i). Therefore prefix complexity $\mathrm{K}(i) = -\log_2 m_i$ is bounded by $-\log_2 r_i + O(1)$ (see, e.g, [7]). We say that the series $\sum r_i$ *converges slowly in the Solovay sense* (has the *Solovay property*) if this bound is tight infinitely often, i.e., if $r_i \geqslant \varepsilon m_i$ for some $\varepsilon > 0$ and for infinitely many i. In other terms, the series does *not* converge slowly if $r_i/m_i \to 0$.

In [1,4] the name *Solovay function* was used for a computable bound $S(i)$ for prefix complexity $\mathrm{K}(i)$ that is tight infinitely often, i.e., $\mathrm{K}(i) \leqslant S(i) + O(1)$ for every i and $\mathrm{K}(i) \geqslant S(i) - c$ for some c and for infinitely many values of i. Thus, a computable series $\sum a_i$ of positive rational numbers has the Solovay property if and only if $i \mapsto -\log_2 a_i$ is a Solovay function [1].

Theorem 9. *Let $\alpha = \sum_i r_i$ be a computable converging series of positive rational numbers. The number α is random if and only if this series converges slowly in the Solovay sense.*

In other terms, the sum is non-random if and only if the ratio r_i/m_i tends to 0.

Proof. Assume that $r_i/m_i \to 0$. Then for every ε we can let $h_i = \varepsilon m_i$ and get a lower semicomputable sequence that satisfies the conditions of Theorem 7. Therefore α is not random.

We can also prove that α is not complete (thus providing an alternative proof of its non-randomness). Recall the argument used in the proof of Theorem 2: if $r_i \leqslant m_i$, then $\sum r_i \preceq_1 \sum m_i$. And if $r_i \leqslant cm_i$, then $\sum r_i \preceq_c \sum m_i$. This remains true if the inequality $r_i \leqslant cm_i$ is true for all sufficiently large i. So for a fast (non-Solovay) converging series and its sum α we have $\alpha \preceq_c \sum m_i$ *for arbitrarily small* c. If α were complete, we would have also $\sum m_i \preceq_d \alpha$ for some d and therefore $\alpha \preceq_{cd} \alpha$ for some d and all $c > 0$. For

small enough c we have $cd < 1/2$ and therefore $\alpha \preceq_{1/2} \alpha$ i.e., $2\alpha \preceq_1 \alpha$. Then, as we saw on page 33, α should be computable.

It remains to show the reverse implication. Assuming that $\alpha = \sum r_i$ is not random, we need to prove that $r_i/m_i \to 0$. Consider the interval $[0,\alpha]$ split into intervals of length $r_0, r_1, \ldots$. Given an open cover of α with small measure, we consider those intervals (of length $r_0, r_1, \ldots$, see above) that are completely covered (endpoints included). They form an enumerable set and the sum of their lengths does not exceed the measure of the cover. If the cover has measure 2^{-2n} for some n, we may multiply the corresponding r_i by 2^n and their sum remains at most 2^{-n}. Note also that for large enough i the ith interval is covered (since it is close to α and α is covered). So for each n we get a semimeasure $M^n = M^n_0, M^n_1, \ldots$ such that $M^n_i/r_i \geqslant 2^n$ for sufficiently large i and $\sum_i M^n_i < 2^{-n}$. Taking the sum of all M^n, we get a lower semicomputable semimeasure M such that $r_i/M_i \to 0$. Then $r_i/m_i \to 0$ also for the universal semimeasure m. □

This result provides yet another proof that a sum of two non-random lower semicomputable reals is non-random (since the sum of two sequences that converge to 0 also converges to 0).

It shows also that Solovay functions exist (which is not immediately obvious from the definition). Moreover, it shows that there exist computable *non-decreasing* Solovay functions: take a computable series of rational numbers with random sum and make this series non-increasing not changing the sum (by splitting too big terms into small pieces).

It also implies that slow convergence (in the Solovay sense) is not a property of a series itself, but only of its sum. It looks strange: some property of a computable series (of positive rational numbers), saying that *infinitely many terms come close to the upper bound provided by a priori probability*, depends only on the sum of this series. At first it seems that by splitting the terms into small parts we can destroy the property not changing the sum, but it is not so. In the next section we try to understand this phenomenon providing a direct proof for it (and as a byproduct we get some improvements in the result of this section).

7 The Solovay Property as a Property of the Sum

First, let us note that the Solovay property is invariant under computable permutations. Indeed, computable permutation π changes the a priori probability only by a constant factor: $m_{\pi(i)} = \Theta(m_i)$. Then let us consider grouping. Since we want to allow infinite groups, let us consider a computable series $\sum_{i,j} a_{ij}$ of non-negative rational numbers. Then

$$\alpha = \sum_{i,j} a_{ij} = (a_{00} + a_{01} + \ldots) + (a_{10} + a_{11} + \ldots) + \ldots = \sum_i A_i,$$

where $A_i = \sum_j a_{ij}$.

We want to show that A_i and a_{ij} are slowly converging series (in the Solovay sense) at the same time. Note that slow convergence is permutation-invariant, so it is well defined for two-dimensional series.

However, some clarifications and restrictions are needed. First, $\sum A_i$ is not in general a computable series, it is only a lower semicomputable one. We can extend the definition

of the Solovay property to lower semicomputable series, still requiring $A_i = O(m_i)$, and asking this bound to be $O(1)$-tight infinitely often. Second, such a general statement is not true: imagine that all non-negative terms are in the first group A_0 and all $A_1, A_2, \ldots$ are zeros. Then $\sum A_i$ does not have the Solovay property while $\sum a_{ij}$ could have it.

The following result is essentially in [4]:

Theorem 10. *Assume that each group A_i contains only finitely many non-zero terms. Then the properties $A_i/m_i \to 0$ and $a_{ij}/m_{ij} \to 0$ are equivalent.*

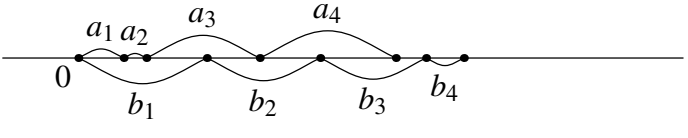
Here m_{ij} is the a priori probability of pair $\langle i, j\rangle$ (or its number in some computable numbering, this does not matter up to $O(1)$-factor). The convergence means that for every $\varepsilon > 0$ the inequality $a_{ij}/m_{ij} > \varepsilon$ is true only for finitely many pairs $\langle i, j\rangle$.

Proof. Let us recall first that $m_i = \sum_j m_{ij}$ up to a $O(1)$-factor. (Indeed, the sum in the right hand side is lower semicomputable, so it is $O(m_i)$ due to the maximality. On the other hand, already the first term m_{i0} is $\Omega(m_i)$.) So if a_{ij}/m_{ij} tends to zero, the ratio $A_i/\sum_j m_{ij}$ does the same (only finitely many pairs have $a_{ij} > \varepsilon m_{ij}$ and they appear only in finitely many groups).

It remains to show that $A_i/m_i \to 0$ implies $a_{ij}/m_{ij} \to 0$. Here we need to use that only finitely many terms in each group are non-zero. For this it is enough to construct some lower semicomputable $\tilde{m}_{ij}$ such that $a_{ij}/\tilde{m}_{ij} \to 0$, somehow using the fact that $A_i/m_i \to 0$. The natural idea would be to split m_i between $\tilde{m}_{ij}$ in the same proportion as A_i is split between a_{ij}. However, for this we need to know how many terms among $a_{i0}, a_{i1}, \ldots$ are non-zero, and in general this is a non-computable information. (For the special case of finite grouping this argument would indeed work.)

So we go in the other direction. For some constant c we may let $\tilde{m}_{ij}$ to be ca_{ij} while this does not violate the property $\sum_j \tilde{m}_{ij} \leqslant m_i$. (When m_i increases, we increase $\tilde{m}_{ij}$ when possible.) If indeed $A_i/m_i \to 0$, for every constant c we have $cA_i \leqslant m_i$ for all sufficiently large i, so $a_{ij}/\tilde{m}_{ij} \leqslant 1/c$ for all sufficiently large i (and only finitely many pairs $\langle i, j\rangle$ violate this requirement, because each A_i has only finitely many non-zero terms). So we are close to our goal ($a_{ij}/\tilde{m}_{ij} \to 0$): it remains to perform this construction for all $c = 2^{2n}$ and combine the resulting $\tilde{m}$'s with coefficients 2^{-n}. □

As a corollary of Theorem 10 we see (in an alternative way) that the Solovay property depends only on the sum of the series. Indeed, if $\sum_i a_i = \sum_j b_j$, these two series could be obtained by a different grouping of terms in some third series $\sum_k c_k$. To construct c_k, we draw intervals of lengths $a_1, a_2, \ldots$ starting from zero point, as well as the intervals of lengths $b_1, b_2 \ldots$; combined endpoints split the line into intervals of lengths $c_1, c_2, \ldots$ (as shown):



In this way we get not only the alternative invariance proof, but also can strengthen Theorem 9. It dealt with computable series of rational numbers. Now we still consider series of rational numbers but the summands are presented as lower semicomputable

numbers and each has only finitely many different approximations. (So $r_i = \lim_n r(i,n)$ where r is a computable function of i and n with rational values which is non-decreasing as a function of n and for every i there are only finitely many different values $r(i,n)$.)

Now the result of [4] follows easily:

Theorem 11. *Let $\alpha = \sum_i r_i$ be a converging semicomputable series of rational numbers in the sense explained above. The number α is random if and only if this series converges slowly in the Solovay sense (i.e., r_i/m_i does not converge to 0).*

Proof. Indeed, each r_i is a sum of a computable series of non-negative rational numbers with only finitely many non-zero terms. So we can split $\sum r_i$ into a double series not changing the sum (evidently) and the Solovay property (due to Theorem 10). □

In particular, we get the following corollary: *an upper semicomputable function $n \mapsto f(n)$ with integer values is an upper bound for $\mathrm{K}(n)$ if and only if $\sum_n 2^{-f(n)}$ is finite; this bound is tight infinitely often if and only if this sum is random.*

Now we can show an alternative proof that all complete reals have the Solovay property. First we observe that the Solovay property is upward closed with respect to Solovay reducibility. Indeed, if $\sum a_i$ and $\sum b_i$ are computable series of non-negative rational numbers, and a_i converges slowly, then $\sum(a_i + b_i)$ converges slowly, too (its terms are bigger). So it remains to prove directly that at least one slowly converging series (or, in other terms, computable Solovay function) exists. To construct it, we watch how the values of a priori probability increase (it is convenient again to consider a priori probability of pairs):

$$\begin{array}{ccccc} m_{00} & m_{01} & m_{02} & m_{03} & \ldots \\ m_{10} & m_{11} & m_{12} & m_{13} & \ldots \\ m_{20} & m_{21} & m_{22} & m_{23} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \end{array}$$

and fill a similar table with rational numbers a_{ij} in such a way that $a_{ij}/m_{ij} \not\to 0$. How do we fill this table? For each row we compute the sum of current values $m_{i,*}$; if it crosses one of the thresholds $1/2, 1/4, 1/8 \ldots$, we put the crossed threshold value into the a-table (filling it with zeros from left to right while waiting for the next threshold crossed). In this way we guarantee that a_{ij} is a computable function of i and j; the sum of a-values is at most twice bigger than the sum of m-values; finally, in every row there exists at least one a-value that is at least half of the corresponding m-value. Logarithms of a-values form a Solovay function (and a_{ij} itself form a slowly convergent series).

Note that this construction does not give a *nondecreasing* Solovay function directly (it seems that we still need to use the arguments from the preceding section).

8 Busy Beavers and Convergence Regulators

We had several definitions that formalize the intuitive idea of a "slowly converging series". However, the following one (probably the most straightforward) was not considered yet. If $a_n \to \alpha$, for every $\varepsilon > 0$ there exists some N such that $|\alpha - a_n| < \varepsilon$ for all $n > N$. The minimal N with this property (considered as a function of ε, denoted

by $\varepsilon \mapsto N(\varepsilon)$) is called *modulus of convergence*. A sequence (or a series) should be considered "slowly converging" if this function grows fast. Indeed, slow convergence (defined as the Solovay property) could be equivalently characterized in these terms (see Theorem 13 below).

First we define a prefix-free version of busy beaver function:

Definition 4. *Let m be a natural number. Define $BP(m)$ as the minimal value of N such that $\mathrm{K}(n) > m$ for all $n > N$.*

In other terms, $BP(m)$ is the maximal number n whose prefix complexity $\mathrm{K}(n)$ does not exceed m. Let us recall a well-known natural interpretation of $BP(m)$ in terms of "busy beavers":

Theorem 12. *Fix an optimal prefix-free universal machine M. Let $T(m)$ be the maximal time needed for termination of (terminating) programs of length at most m. Then*

$$BP(m-c) \leqslant T(m) \leqslant BP(m+c)$$

for some c and all m.

Proof. First we prove that for all $t > T(m)$ the compexity of t is at least $m - O(1)$, thus showing that $T(m) \geqslant BP(m-c)$. Indeed, let $\mathrm{K}(t) = m - d$. Appending the shortest program for t to the prefix-free description of d, we get a prefix free description of the pair $\langle t, m\rangle$. Indeed, we can reconstruct t and $m - d$ from the shortest program of t (the second is its length) and then add d and get m. Then, knowing t and m, we run t steps of all programs of length at most m, and then choose the first string that is not among their outputs. This string has by construction prefix complexity greater than m, and it is (prefix-freely) described by $m - d + O(\log d)$ bits, so $d = O(1)$.

On the other hand, $T(m)$ can be (prefix-freely) described by most long-playing program of size at most m (program determines its execution time), so $\mathrm{K}(T(m)) \leqslant m + O(1)$ and therefore $T(m) \leqslant BP(m + O(1))$. □

Now we can prove the equivalence of two notions of "slow convergence":

Theorem 13. *The computable series of non-negative rational numbers $\sum r_i$ has the Solovay property ($\Leftrightarrow$ has a random sum) if and only its modulus of convergence satisfies the inequality $N(2^{-m}) > BP(m-c)$ for some c and for all m.*

Proof. Let $\alpha = \sum r_i = \lim a_i$, where $a_i = r_0 + \ldots + r_{i-1}$. Assume that α is random. We have to show that $|\alpha - a_i| < 2^{-m}$ implies $\mathrm{K}(i) > m - O(1)$; this shows that $N(2^{-m}) \geqslant BP(m - O(1))$. Since $\mathrm{K}(i) = \mathrm{K}(a_i) + O(1)$, it is enough to show that every rational 2^{-m}-approximation to α has complexity at least $m - O(1)$. This is a bit stronger condition than the condition $\mathrm{K}(\alpha_0 \ldots \alpha_{m-1}) \geqslant m - O(1)$ (used in prefix complexity version of Schnorr–Levin theorem) since now we consider *all* approximations, not only the prefix of the binary expansion. However, it can be proven in a similar way.

Let c be some integer. Consider an effectively open set U_c constructed as follows. For every rational r we consider the neighborhood around r of radius $2^{-\mathrm{K}(r)-c}$; the set U_c is the union of these neighborhoods. (Since $\mathrm{K}(r)$ is upper semicomputable, it is indeed an effectively open set.) The total length of all intervals is $2 \cdot 2^{-c} \sum_r 2^{-\mathrm{K}(r)} \leqslant 2^{-(c-1)}$.

Therefore, U_c form a Martin-Löf test, and random α does not belong to U_c for some c. This means that complexity of 2^{-m}-approximations of α is at least $m - O(1)$.

In the other direction we can use Schnorr–Levin theorem without any changes: if $N(2^{-m}) \geqslant BP(m-c)$, then $\mathrm{K}(i) \geqslant m - O(1)$ for every i such that a_i is a 2^{-m}-approximation to α. Therefore, the m-bit prefix of α has complexity at least $m - O(1)$, since knowing this prefix we can effectively find an a_i that exceeds it (and the corresponding i). □

Question. Note that this theorem shows equivalence between two formalizations of an intuitive idea of "slowly converging series" (or three, if we consider the Solovay reducibility as a way to compare the rate of convergence). However, the proof goes through Martin-Löf randomness of the sum (where the series itself disappears). Can we have a more direct proof? Can we connect the Solovay reducibility (not only completeness) to the properties of the modulus of convergence?

Reformulating the definition of $BP(m)$ in terms of a priori probability, we say that $BP(m)$ is the minimal N such that all $n > N$ have a priori probability less than 2^{-m}. However, in terms of a priori probability the other definition looks more natural: let $BP'(m)$ be the minimal N such that the total a priori probability of all $n > N$ is less than 2^{-m}. Generally speaking, $BP'(m)$ can be greater that $BP(m)$, but it turns out that it still can be used to characterize randomness in the same way:

Theorem 14. *Let a_i be a computable increasing sequence of rational numbers that converges to a random number α. Then $N(2^{-m}) \geqslant BP'(m-c)$.*

Proof. Since all $i > N(2^{-m})$ have the same a priori probability as the corresponding a_i (up to some $O(1)$-factor), it is enough to show that for every m the sum of a priori probabilities of all rational numbers in the 2^{-m}-neighborhood of a random α is $O(2^{-m})$ (recall that for all $i > N(2^{-m})$ the corresponding a_i belong to this neighborhood).

As usual, we go in the other direction and cover all "bad" α that do not have this property by a set of small measure. Not having this property means that for every c there exists m such that the sum of a priori probabilities of rational numbers in the 2^{-m}-neighborhood of α exceeds $c2^{-m}$. For a given c, we consider all intervals with rational endpoints that have the following property: *the sum of a priori probabilities of all rational numbers in this interval is more than $c/2$ times bigger than the interval's length*. Every bad α is covered by an interval with this property (the endpoints of the interval $(\alpha - 2^{-m}, \alpha + 2^{-m})$ can be changed slightly to make them rational), and the set of intervals having this property is enumerable. It is enough to show that the union of all such intervals has measure $O(1/c)$, in fact, at most $4/c$.

It is also enough to consider a finite union of intervals with this property. Moreover, we may assume that this union does not contain redundant intervals (that can be deleted without changing the union). Let us order all the intervals according to their left endpoints:

$$(l_0, r_0), (l_1, r_1), (l_2, r_2), \ldots$$

where $l_0 \leqslant l_1 \leqslant l_2 \leqslant \ldots$ It is easy to see that right endpoints go in the same order (otherwise one of the intervals would be redundant). So $r_0 \leqslant r_1 \leqslant r_2 \leqslant \ldots$ Now note that $r_i \leqslant l_{i+2}$, otherwise the interval (l_{i+1}, r_{i+1}) would be redundant. Therefore, intervals with even numbers $(l_0, r_0), (l_2, r_2), (l_4, r_4) \ldots$ are disjoint, and for each of them the

length is $c/2$ times less than the sum of a priori probabilities of rational numbers inside it. Therefore, the total length of these intervals does not exceed $2/c$, since the sum of all priori probabilities is at most 1. The same is true for intervals with odd numbers, so in total we get the bound $4/c$. □

Question: We see that both BP and BP' can be used to characterize randomness, but how much could BP and BP' differ in general?

Acknowledgments. The authors are grateful to the organizers of the conference for the opportunity to present this work, and to L. Staiger and R. Hölzl for comments.

References

1. Bienvenu, L., Downey, R.: Kolmogorov complexity and Solovay functions. In: Symposium on Theoretical Aspects of Computer Science (STACS 2009). Dagstuhl Seminar Proceedings, vol. 09001, pp. 147–158. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany Internationales Begegnungs- und Forschungszentrum fuer Informatik (IBFI), Schloss Dagstuhl, Germany (2009), http://drops.dagstuhl.de/opus/volltexte/2009/1810
2. Calude, C., Hertling, P., Khoussainov, B., Wang, Y.: Recursively Enumerable Reals and Chaitin Omega Numbers. In: Meinel, C., Morvan, M. (eds.) STACS 1998. LNCS, vol. 1373, pp. 596–606. Springer, Heidelberg (1998)
3. Chaitin, G.: Information-theoretical characterizations of recursive infinte strings. Theoretical Computer Science 2, 45–48 (1976)
4. Hölzl, R., Kräling, T., Merkle, W.: Time-Bounded Kolmogorov Complexity and Solovay Functions. In: Královič, R., Niwiński, D. (eds.) MFCS 2009. LNCS, vol. 5734, pp. 392–402. Springer, Heidelberg (2009)
5. Kučera, A., Slaman, T.: Randomness and recursive enumerability. SIAM Journal on Computing 31, 199–211 (2001)
6. Levin, L.: Forbidden information. In: The 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS 2002), p. 761 (2002)
7. Shen, A.: Algorithmic Information theory and Kolmogorov complexity. Technical report TR2000-034. Technical report, Uppsala University (2000)
8. Solovay, R.: Draft of a paper (or series of papers) on Chaitin's work. Unpublished notes, 215 pages (1975)

Constructing the Infimum of Two Projections

Douglas S. Bridges and Luminita S. Vîţă

Department of Mathematics & Statistics,
University of Canterbury, Christchurch, New Zealand
d.bridges@math.canterbury.ac.nz, luminita.vitza@gmail.com

Abstract. An elementary algorithm for computing the infimum of two projections in a Hilbert space is examined constructively. It is shown that in order to obtain a constructive convergence proof for the algorithm, one must add some hypotheses such as Markov's principle or the locatedness of a certain range; and that in the finite-dimensional case, the existence of both the infimum and the supremum of the two projections suffices for the convergence of the algorithm.

1 Introduction

Consider a separable complex Hilbert space H. We define the **infimum of the projections** E and F of H to be the unique projection P (if it exists) that satisfies the following two conditions:

(a) $P \leqslant E$ and $P \leqslant F$.
(b) If Q is a projection with $Q \leqslant E$ and $Q \leqslant F$, then $Q \leqslant P$,

where $\leqslant$ is the usual ordering of projections on H. We then denote the infimum by $E \wedge F$. Classically, $E \wedge F$ always exists, and is the projection on the intersection of $\operatorname{ran} E$ (the range of E) and $\operatorname{ran} F$ ([11], page 111). However, within Bishop-style constructive mathematics, **BISH**,[1] the projection on a closed linear subset S of a Hilbert space H exists if and only if S is **located**: that is,

$$\rho(x, S) \equiv \inf\{\rho(x, s) : s \in S\}$$

exists for each $x \in H$ (see [3] page 366, Theorem (8.7)). Since there is no guarantee that the intersection of two located sets is also located, the infimum of two projections may not exist.

It can be shown classically that the decreasing sequence $\left((EFE)^n\right)_{n \geqslant 1}$ of projections converges strongly to a projection P on H, in the sense that

$$Px = \lim_{n \to \infty} (EFE)^n x$$

[1] **BISH** is, in essence, just mathematics carried out with intuitionistic logic and some suitable set- or type-theoretic foundation such as the Aczel-Rathjen-Myhill **CST** [1,2,13] or Martin-Löf's type theory [12]. We also accept the principles of countable and dependent choice as part of **BISH**. More information about analysis in **BISH** can be found in [3,6].

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 46–58, 2012.

for each $x \in H$; it then follows that P satisfies conditions (a) and (b), and is therefore the infimum of the projections E and F ([9], page 257). Thus in classical mathematics there is an analytic characterisation of the infimum of two projections. In this paper we examine, within **BISH**, the connection between the existence of P and the strong convergence of the sequence $((EFE)^n)_{n\geqslant 1}$.

2 The Algorithm

For future reference, we first note:

Proposition 1. *The projection $E \wedge F$ exists if and only if $E(H) \cap F(H)$ is located, in which case $E \wedge F$ is the projection on $E(H) \cap F(H)$.*

Proof. Suppose that $K \equiv E(H) \cap F(H)$ is located, and let G be the projection of H onto K. Then, by Proposition 2.5.2 of [11], $G \leqslant E$ and $G \leqslant H$. Also, by the same proposition, if P is a projection, $P \leqslant E$ and $P \leqslant F$, then $P(H) \subset E(H)$ and $P(H) \subset F(H)$, so $P(H) \subset K$ and therefore $P \leqslant G$. Thus $G = E \wedge F$.

Now suppose that $G \equiv E \wedge F$ exists, and let $K \equiv G(H)$. Then $G \leqslant E$, so $K \subset E(H)$, and similarly $K \subset F(H)$; so $K \subset E(H) \cap F(H)$. On the other hand, if $x \in E(H) \cap F(H)$ and P is the projection of H on $\mathbf{C}x$, then, by Proposition 2.5.2 of [11], $P \leqslant E$ and $P \leqslant F$, so $P \leqslant G$ and therefore, again by that proposition, $x \in P(H) \subset G(H) = K$. Hence $E(H) \cap F(H) = K$, which is located, and G is the projection of H onto $E(H) \cap F(H)$.

A famous theorem of Specker shows that the monotone convergence theorem in **R** is false in the recursive constructive mathematics (**RUSS**) of the Markov School; see [15] or [5] (Chapter 3). It follows that the monotone convergence theorem for projections of a Hilbert space ([11], Lemma 5.1.4) is also false in **RUSS**, and therefore, since **RUSS** is consistent with **BISH**, not constructively provable. In consequence, Halmos's classical proof of the statement

$$(E \wedge F)\, x = \lim_{n\to\infty} (EFE)^n\, x \quad (x \in H) \tag{1}$$

fails constructively. However, we can adapt Halmos's argument to obtain the following constructive result.

Proposition 2. *Let E, F be projections on H such that the strong limit P of the sequence $((EFE)^n)_{n\geqslant 1}$ exists. Then $P = E \wedge F$. Moreover, the sequence $((FEF)^n)_{n\geqslant 1}$ converges strongly to $E \wedge F$.*

Proof. It is easily seen that P is a linear mapping of H into itself. For each $x \in H$ and each n,

$$\langle (EFE)^n\, x, x\rangle \geqslant \left\langle (EFE)^{2n}\, x, x\right\rangle = \langle (EFE)^n\, x, (EFE)^n\, x\rangle \geqslant 0,$$

so

$$\langle Px, x\rangle = \lim_{n\to\infty} \langle (EFE)^n\, x, x\rangle \geqslant 0.$$

Hence P is a positive operator. On the other hand, if $m > n$, then $(EFE)^m \leqslant (EFE)^n$, so for all $x \in H$,

$$\begin{aligned}\langle (EFE)^m x, (EFE)^n x\rangle &= \langle (EFE)^n (EFE)^m x, x\rangle \\ &= \langle (EFE)^m x, x\rangle\end{aligned}$$

and therefore

$$\begin{aligned}\left\langle P^2 x, x\right\rangle &= \langle Px, Px\rangle \\ &= \lim_{m,n\to\infty} \langle (EFE)^m x, (EFE)^n x\rangle \\ &= \lim_{m\to\infty} \langle (EFE)^m x, x\rangle = \langle Px, x\rangle .\end{aligned}$$

Thus P is idempotent and therefore a projection. Since $(EFE)^n x \in E(H)$ for each n, we have $P \leqslant E$. If $x \in E(H) \cap F(H)$, then $Ex = x$ and $Fx = x$, so $(EFE)^n x = x$ for each n; whence $Px = x$. For all m, n we have

$$(EFE)^m F(EFE)^n = (EFE)^m (EFE)(EFE)^n = (EFE)^m (EFE)^{n+1} .$$

Letting $n \to \infty$, we obtain $(EFE)^m FP = P$. Now letting $m \to \infty$, we obtain $PFP = P$. Hence

$$\begin{aligned}0 &= P - PFP = P(I - F)P \\ &= P(I - F)(I - F)P = P(I - F)(P(I - F))^* ,\end{aligned}$$

so

$$\|P(I - F)x\|^2 = \left|\left\langle P(I - F)(P(I - F))^* x, x\right\rangle\right| = 0$$

for each $x \in H$, and therefore $P(I - F) = 0$. Thus $P = PF$ and therefore $P \leqslant F$. We now have $P \leqslant E$ and $P \leqslant F$, so for each $x \in H$, $Px \in E(H) \cap F(H)$. On the other hand, if $x \in E(H) \cap F(H)$, then $Ex = x$ and $Fx = x$, so $(EFE)^n x = x$ for each n; whence $Px = x$. It follows that $x \in E(H) \cap F(H)$ if and only if $Px = x$; whence P is the projection $E \wedge F$ on $E(H) \cap F(H)$. Finally, for $n \geqslant 2$ we have $(FEF)^n = F(EFE)^{n-1} F$, so the sequence $((FEF)^n)_{n\geqslant 1}$ converges strongly to FPF; but $FPF = P = E \wedge F$.

Can we prove, conversely, that if $E \wedge F$ exists, then the sequence $((EFE)^n x)_{n\geqslant 1}$ converges for each $x \in H$? To see that the answer is "no",[2] consider the case where[3] $H = \mathbf{R}^2$, E is the projection of H on the subspace $\mathbf{R}e$, where $e = (0, 1)$, and F is the projection on $\mathbf{R}(\cos\theta, \sin\theta)$, where $\neg(\theta = 0)$. (Note carefully the distinction between "$\neg(\theta = 0)$" and "$\theta \neq 0$": the latter means that $|\theta| > 0$.) In this situation we have $E \wedge F = 0$. Also,

$$(EFE)^n e = \left(\cos^{2n}\theta, 0\right)$$

[2] This Brouwerian example was originally presented in [8].

[3] In this example, which recurs in the paper, the Hilbert space is over $\mathbf{R}$ for convenience. The Hilbert space H used outside this example is a complex one.

for each n; but this converges to $(0,0)$ if and only if $\cos\theta \neq 1$ (that is, $|1-\cos\theta| > 0$) and therefore $\theta \neq 0$. Thus if the answer to our last question were "yes", we could prove that

$$\forall_{\theta\in\mathbf{R}}\ (\neg(\theta=0) \Rightarrow \theta \neq 0)\,,$$

which is easily shown to be equivalent to **Markov's Principle**:

> **MP:** For each binary sequence $(a_n)_{n\geqslant 1}$, if it is impossible that $a_n = 0$ for all n, then there exists n such that $a_n = 1$.

This principle, equivalent to an unbounded search, is not a part of **BISH** (though it is consistent with **BISH** and is accepted **RUSS**).[4]

The estimates in the next lemma will enable us to make progress with the question posed at the start of the preceding paragraph.

Lemma 1. *Let H be a Hilbert space, and E, F projections on H such that $E\wedge F$ exists. Then for each $x \in H$, there exists a strictly increasing sequence $(n_k)_{k\geqslant 1}$ of positive integers such that*

$$\|(E-F)\,(EFE)^{n_k}\,x\| < 2^{-k} \quad (k \geqslant 1)\,. \tag{2}$$

Proof. For each positive integer n,

$$\begin{aligned}
&(EFE)^{2n} - (EFE)^{2n+1}\\
&= (EFE)^{2n} - (EFE)^n F (EFE)^n\\
&= (EFE)^n\,(I-F)\,(EFE)^n\\
&= (EFE)^n\,(I-F)^2\,(EFE)^n\\
&= ((I-F)\,(EFE)^n)^*\,(I-F)\,(EFE)^n\,.
\end{aligned}$$

Hence

$$\begin{aligned}
\|(I-F)\,(EFE)^n\,x\|^2 &= \langle (I-F)\,(EFE)^n x, (I-F)\,(EFE)^n\,x\rangle\\
&= \left\langle ((I-F)\,(EFE)^n)^*\,(I-F)\,(EFE)^n\,x, x\right\rangle\\
&= \left\langle \left((EFE)^{2n} - (EFE)^{2n+1}\right) x, x\right\rangle.
\end{aligned}$$

Given $\varepsilon > 0$, pick N such that $N\varepsilon > \left\langle (EFE)^2\,x, x\right\rangle$. Suppose that

$$\left\langle \left((EFE)^{2n} - (EFE)^{2n+1}\right) x, x\right\rangle > \varepsilon$$

and therefore

$$\left\langle (EFE)^{2n}\,x, x\right\rangle \geqslant \left\langle (EFE)^{2n+1}\,x, x\right\rangle + \varepsilon \geqslant \left\langle (EFE)^{2n+2}\,x, x\right\rangle + \varepsilon$$

[4] It is an exercise to prove that, in the notation of our example, if $\theta \neq 0$, then $((EFE)^n\,x)_{n\geqslant 1}$ converges to 0 for each $x \in H$.

for $1 \leqslant n \leqslant N$. Then

$$\begin{aligned}\left\langle (EFE)^2 x, x\right\rangle &> \left\langle (EFE)^4 x, x\right\rangle + \varepsilon \\ &> \left\langle (EFE)^6 x, x\right\rangle + 2\varepsilon \\ &> \cdots \\ &> \left\langle (EFE)^{2(N+1)} x, x\right\rangle + N\varepsilon \geqslant N\varepsilon,\end{aligned}$$

a contradiction. Hence there exists $n \leqslant N$ such that

$$\begin{aligned}\|(E-F)(EFE)^n x\| &= \|(I-F)(EFE)^n x\|^2 \\ &= \left\langle \left((EFE)^{2n} - (EFE)^{2n+1}\right) x, x\right\rangle < 2\varepsilon.\end{aligned}$$

Since $\varepsilon > 0$ is arbitrary, it is now straightforward to construct, inductively, a strictly increasing sequence $(n_k)_{k\geqslant 1}$ of positive integers such that (2) holds.

For our first theorem we need to know that an operator T on a Hilbert space H is **weak-sequentially open** if for each sequence $(x_n)_{n\geqslant 1}$ such that $Tx_n \to 0$, there exists a sequence $(y_n)_{n\geqslant 1}$ in $\ker T$ such that $x_n + y_n \overset{w}{\to} 0$, where $\overset{w}{\to}$ denotes weak convergence in H. According to Corollary 6.5.8 of [6], if T is a jointed operator—that is, one with an adjoint[5]—on H, then the following conditions are equivalent:

- T has located range.
- $\ker T^*$ is located and T^* is weak-sequentially open.
- $\ker T$ is located and T is weak-sequentially open.
- T^* has located range.

In particular, if $\ker T$ is located, then a necessary and sufficient condition for $\operatorname{ran}(T)$ to be located is that T is weak-sequentially open.[6]

Theorem 1. *Let H be a separable Hilbert space, and E, F projections on H such that $E \wedge F$ exists and $E - F$ is weak-sequentially open. Then $((EFE)^n)_{n\geqslant 1}$ converges strongly to $E \wedge F$.*

Proof. Let P denote $E \wedge F$, the projection on $E(H) \cap F(H)$ (Proposition 1). Fix x in H. Using Lemma 1, construct a strictly increasing sequence $(n_k)_{k\geqslant 1}$ of positive integers such that

$$\|(E-F)(EFE)^{n_k} x\| < 2^{-k} \qquad (k \geqslant 1).$$

[5] In **BISH** the existence of the adjoint cannot be proved in general (see page 101 of [6]). Richman [14] and Ishihara [10] have provided an elegant criterion for the existence of the adjoint; see also Section 6.3 of [6].

[6] Hence, classically, every bounded operator on H is weak-sequentially open.

Since $E - F$ is weak-sequentially open, there exists a sequence $(y_k)_{k\geqslant 1}$ in $\ker(E-F)$ such that $(EFE)^{n_k} x + y_k \xrightarrow{w} 0$ as $k \to \infty$. For each $z \in H$,

$$\begin{aligned}\langle (I-P)\, E\left[(EFE)^{n_k} x + y_k\right], z\rangle &= \langle (EFE)^{n_k} x + y_k, E\,(I-P)\, z\rangle \\ &\to 0 \text{ as } k \to \infty.\end{aligned}$$

However, since $(E-F)\, y_k = 0$,

$$Ey_k = Fy_k \in E(H) \cap F(H) = \operatorname{ran}(P)$$

and so $(I-P)\, Ey_k = 0$. Hence

$$(I-P)\,(EFE)^{n_k} x = (I-P)\, E\left[(EFE)^{n_k} x + y_k\right] \xrightarrow{w} 0 \text{ as } k \to \infty.$$

Now, for $n \geqslant n_k$ we have

$$0 \leqslant \langle (I-P)\,(EFE)^n x, x\rangle \leqslant \langle (I-P)\,(EFE)^{n_k} x, x)\rangle\,.$$

Hence $\langle (I-P)\,(EFE)^n x, x\rangle \to 0$ as $n \to \infty$. Since $PE = P = EP$ and $PF = F = FP$, we now have

$$\begin{aligned}\left\|(EFE)^n x - Px\right\|^2 &= \left\|(I-P)\,(EFE)^n x\right\|^2 \\ &= \langle (I-P)\,(EFE)^n x, (I-P)\,(EFE)^n x\rangle \\ &= \left\langle (EFE)^n (I-P)^2 (EFE)^n x, x\right\rangle \\ &= \left\langle (I-P)\,(EFE)^{2n} x, x\right\rangle \\ &\to 0 \text{ as } n \to \infty.\end{aligned}$$

Since $x \in H$ is arbitrary, we have proved the desired strong convergence of the sequence $\left((EFE)^n\right)_{n\geqslant 1}$ to $E \wedge F$.

Corollary 1. *Let H be a separable Hilbert space, and E, F projections on H such that $E \wedge F$ exists and $E-F$ has located range. Then $\left((EFE)^n\right)_{n\geqslant 1}$ converges strongly to $E \wedge F$.*

Proof. Apply the preceding theorem, noting the comment immediately before its statement.

Corollary 2. *Let H be a separable Hilbert space, and E, F projections on H such that $E \wedge F$ exists and $E-F$ has closed range. Then $\left((EFE)^n\right)_{n\geqslant 1}$ converges strongly to $E \wedge F$.*

Proof. By Theorem 6.5.9 of [6], the range of $E - F$ is located; so we can immediately invoke Corollary 1.

In order to analyse our earlier Markovian example further, we prove an elementary lemma in plane Euclidean geometry.

Lemma 2. *Let E, F be 1-dimensional projections in $\mathbf{R}^2$ such that $E - F \neq 0$. Then* $\operatorname{ran}(E - F) = \mathbf{R}^2$.

Proof. Without loss of generality take F as the projection on the x-axis, and E as the projection on the line through the origin at an angle $\theta \neq 0$ with the positive x-axis. The outline of the proof is this. Take any point $\mathbf{x} \neq \mathbf{0}$ in $\mathbf{R}^2$; for illustration, consider the case where $\mathbf{x}$ lies above the range of E and in the first quadrant. Form the parallelogram Π with one vertex at the origin, one side the vector $\mathbf{x}$, one side parallel to that vector and with one vertex on each of the ranges of E and of F, and one side along the x-axis. Draw the perpendicular to $\operatorname{ran}(F)$ through the nonzero vertex of Π on the x-axis, and the perpendicular to $\operatorname{ran}(E)$ through the vertex of Π diagonally opposite to $\mathbf{0}$. These two perpendiculars meet in a single point $\mathbf{z}$, and $\mathbf{x} = (E - F)\mathbf{z}$. Thus every nonzero vector in $\mathbf{R}^2$ is in the range of $E - F$, which is therefore dense in $\mathbf{R}^2$; from which it follows that $\operatorname{ran}(E - F) = \mathbf{R}^2$.

To accomplish the foregoing without reference to the geometrical figure, take F as the projection on $\mathbf{R}e$ (recall that $e = (1, 0)$) and

$$E = \mathbf{R}(\cos\theta, \sin\theta),$$

where, for convenience, we use E to denote both the projection and its range. Consider any $\mathbf{x} \equiv (x_1, x_2)$ in $\mathbf{R}^2$, and take the case where $0 < \theta < \pi$ (so $\cot\theta$ is defined). Define

$$\begin{aligned} \mathbf{y} &\equiv (x_2 \cot\theta, x_2), \\ \mathbf{z} &\equiv (x_2 \cot\theta - x_1, x_2 + x_1 \cot\theta). \end{aligned}$$

We claim that $E\mathbf{z} = \mathbf{y}$, that $F\mathbf{z} = (x_2 \cot\theta - x_1, 0)$, and hence (clearly) that $\mathbf{x} = (E - F)\mathbf{z}$. First we have

$$\mathbf{z} - \mathbf{y} = (-x_1, x_1 \cot\theta),$$

so

$$\langle \mathbf{z} - \mathbf{y}, (\cos\theta, \sin\theta) \rangle = -x_1 \cos\theta + x_1 \cot\theta \sin\theta = 0$$

and therefore $\mathbf{z} - \mathbf{y}$ is orthogonal to E; whence $E\mathbf{z} = \mathbf{y}$. On the other hand,

$$\langle \mathbf{z} - (x_2 \cot\theta - x_1, 0), e \rangle = 0,$$

so $\mathbf{z} - (x_2 \cot\theta - x_1, 0)$ is orthogonal to $\mathbf{R}e$; whence $F\mathbf{z} = (x_2 \cot\theta - x_1, 0)$. This completes the proof that $E - F$ has range $\mathbf{R}^2$ in the case $0 < \theta < \pi$. The case $0 > \theta > -\pi$ is handled similarly. These two cases are all we need, since we could have stipulated that $|\theta| < \pi$ from the outset.

We now re-examine the Markovian example (first discussed on page 48) in which $\neg(\theta = 0)$ and therefore $\ker(E - F) = \{0\}$. We show how this example accords with Theorem 1 and its corollaries. First, we observe from the remark immediately preceding Theorem 1 that the range of $E - F$ is located if and only if

$E - F$ is weak-sequentially open. Moreover, that range is located if and only if it is finite-dimensional; in that case,

$$\operatorname{ran}(E-F) = \ker(E-F) + \operatorname{ran}(E-F)$$

is dense in $H \equiv \mathbf{R}^2$, so $\operatorname{ran}(E-F)$ is 2-dimensional and equals H.

On the other hand, if $(EFE)^n$ converges strongly to 0, then $\theta \neq 0$, so $E-F \neq 0$ and therefore, by Lemma 2, $E-F$ has closed range equal to $\mathbf{R}^2$. Suppose, conversely, that $E-F$ has closed range. Then $\operatorname{ran}(E-F)$ is located, by the closed range theorem (Theorem 6.5.9 of [6]), and so is finite-dimensional. If $\theta \neq 0$, then, as above, the dimension of $\operatorname{ran}(E-F)$ is 2. Since $\neg\neg(\theta \neq 0)$, that dimension must indeed be 2. In order to obtain $\theta \neq 0$, it will suffice to prove that $\cos\theta < 1$. We can pick $\mathbf{x} \in \mathbf{R}^2$ such that $(E-F)\,\mathbf{x} = (0,1)$. Then

$$\begin{aligned}\langle E\mathbf{x}, F\mathbf{x}\rangle &= \langle (E-F)\,\mathbf{x}, F\mathbf{x}\rangle + \|F\mathbf{x}\|^2 \\ &= \langle (0,1), F\mathbf{x}\rangle + \|F\mathbf{x}\|^2 = \|F\mathbf{x}\|^2,\end{aligned}$$

so

$$\|E\mathbf{x}\|\,\|F\mathbf{x}\|\cos\theta = \|F\mathbf{x}\|^2. \tag{3}$$

Hence

$$\begin{aligned}1 &= \|E\mathbf{x} - F\mathbf{x}\|^2 \\ &= \|E\mathbf{x}\|^2 - 2\langle E\mathbf{x}, F\mathbf{x}\rangle + \|F\mathbf{x}\|^2 \\ &= \|E\mathbf{x}\|^2 - 2\|F\mathbf{x}\|^2 + \|F\mathbf{x}\|^2\end{aligned}$$

and therefore $\|E\mathbf{x}\|^2 = 1 + \|F\mathbf{x}\|^2 \geqslant 1$. Now, either $\|F\mathbf{x}\| < 1$ or $F\mathbf{x} \neq \mathbf{0}$. In the first case we have

$$1 > \|F\mathbf{x}\|^2 = \|E\mathbf{x} - (0,1)\|^2 = \|Ex\|^2 - 2\langle E\mathbf{x}, (0,1)\rangle + 1,$$

so

$$1 - 2\langle E\mathbf{x}, (0,1)\rangle < 0$$

and therefore $\langle E\mathbf{x}, (0,1)\rangle > 1/2$. But $\langle E\mathbf{x}, (0,1)\rangle = \|E\mathbf{x}\|\sin\theta$, so $\sin\theta > 0$ and therefore $\cos\theta < 1$. In the case $F\mathbf{x} \neq \mathbf{0}$, we see from (3) that

$$\cos\theta = \frac{\|F\mathbf{x}\|}{\|E\mathbf{x}\|} = \frac{\|F\mathbf{x}\|}{\sqrt{1 + \|F\mathbf{x}\|^2}} < 1.$$

This completes the proof that if $E-F$ has closed range, then $\theta \neq 0$, and therefore $(EFE)^n$ converges strongly to 0.

This analysis of our Markovian example confirms that it exemplifies Theorem 1 and its corollaries.

3 The Algorithm Revisited

Our aim in this section is to bring Markov's Principle to the fore, by proving the following result.

Theorem 2. ***MP*** $\vdash$ *Let H be a finite-dimensional Hilbert space, and E, F projections on H such that $E \wedge F$ and $E \vee F$ exist. Then $((EFE)^n)_{n \geqslant 1}$ converges strongly to $E \wedge F$.*

This will require a number of technical preliminaries.

Lemma 3. *Let E, F be projections on H such that $E \wedge F = 0$ and $E \vee F$ exist. Then $\ker(E - F) = \ker(E \vee F)$ and is located.*

Proof. For each x we have

$$\begin{aligned}(E \vee F)\, x = 0 &\Leftrightarrow x \perp \operatorname{ran}(E \vee F) \\ &\Leftrightarrow x \in E(H)^{\perp} \cap F(H)^{\perp} \\ &\Leftrightarrow Ex = 0 = Fx \\ &\Leftrightarrow Ex = Fx \in E(H) \cap F(H) = \{0\} \\ &\Leftrightarrow (E - F)x = 0.\end{aligned}$$

Hence $\ker(E - F) = \ker(E \vee F)$. Since $\ker(E \vee F)$ is the orthogonal complement of $\operatorname{ran}(E \vee F)$ and the latter is located, so is the former.

Lemma 4. *Let E, F be projections on a Hilbert space H such that $P \equiv E \wedge F$ and $E \vee F$ exist, let $Q = I - P$, and let $K = Q(H)$. Then $QE|_K$ and $QF|_K$ are projections on K, the projection $QE|_K \wedge QF|_K$ exists and equals 0, and the projection $QE|_K \vee QF|_K$ exists and equals $Q(E \vee F)|_K$.*

Proof. First observe that, by Proposition 2.5.2 of [11], $PE = P = EP$ and $PF = P = FP$; whence Q commutes with E and F. It follows that

$$(QE)^2 = QEQE = QQEE = QE,$$

so QE, and likewise QF, is a projection on H. Hence $QE|_K$ and $QF|_K$ are projections on K. Similarly, Q commutes with $E \wedge F$ and $E \vee F$, and both $Q(E \wedge F)|_K$ and $Q(E \vee F)|_K$ are projections on K. For each $x \in H$ we have

$$\langle QEQx, x\rangle = \langle EQx, Qx\rangle \leqslant \langle (E \vee F)\, Qx, Qx\rangle = \langle Q(E \vee F)\, Qx, x\rangle.$$

Hence $QEQ \leqslant Q(E \vee F)\, Q$ and therefore $QE|_K \leqslant Q(E \vee F)|_K$; similarly, $QF|_K \leqslant Q(E \vee F)|_K$. Now suppose that $QE|_K \leqslant S$ and $QF|_K \leqslant T$, where S, T are projections on K. Then $QS = S$, so

$$(SQ)^2 = SQSQ = SSQ = SQ$$

and therefore SQ is a projection on H. Clearly, SQ is orthogonal to P, so $SQ+P$ is a projection on H. For each $x \in H$ we have

$$\begin{aligned}\langle Ex, x\rangle &= \langle QEQx, x\rangle + \langle (I-Q)\,EQx, x\rangle + \langle E(I-Q)x, x\rangle \\ &= \langle QEQx, Qx\rangle + \langle (I-Q)\,QEx, x\rangle + \langle EPx, x\rangle \\ &\leqslant \langle SQx, Qx\rangle + 0 + \langle Px, x\rangle \\ &= \langle QSQx, x\rangle + \langle Px, x\rangle = \langle (SQ+P)\,x, x\rangle\,.\end{aligned}$$

Hence $E \leqslant SQ + P$; likewise, $F \leqslant SQ + P$. It follows that $E \vee F \leqslant SQ + P$; whence for each $x \in H$,

$$\begin{aligned}\langle Q\,(E \vee F)\,Qx, Qx\rangle &= \langle (E \vee F)Qx, Qx\rangle \\ &\leqslant \langle\langle (SQ+P)\,Qx, Qx\rangle\rangle \\ &= \left\langle SQ^2x, Qx\right\rangle + \langle PQx, Qx\rangle = \langle SQx, Qx\rangle\,.\end{aligned}$$

Thus $Q\,(E \vee F)|_K \leqslant S$ and similarly, $Q\,(E \vee F)|_K \leqslant T$. We now see that $QE|_K \vee QF|_K$ exists and equals $Q\,(E \vee F)|_K$, as required. A similar proof shows that $QE|_K \wedge QF|_K$ exists and equals $Q\,(E \wedge F)|_K$; in this case, since $Q = I - E \wedge F$, we have $QE|_K \wedge QF|_K = 0$.

A subset S of a metric space (X, ρ) is called **incomplete** if there exists a Cauchy sequence $(s_n)_{n\geqslant 1}$ in S that is eventually bounded away from each point x of X, in the sense that there exist N and $\delta > 0$ (depending on x) such that $\rho\,(s_n, x) \geqslant \delta$ for all $n \geqslant N$. On the other hand, a linear mapping $T : X \to Y$ between normed spaces is **unopen** if for each $r > 0$, there exists $y \in Y$ such that $\|y\| < r$ and $y \neq Tx$ for each $x \in X$ with $\|x\| \leqslant 1$. In the presence of countable choice (which we are assuming), this condition is equivalent to the existence of a sequence $(x_n)_{n\geqslant 1}$ of unit vectors in X such that $Tx_n \to 0$.

For our next result we note that every linear operator on a finite-dimensional Hilbert space is both bounded and jointed.[7]

Proposition 3. **MP** $\vdash$ *Let T be a bounded operator mapping H into a finite-dimensional subspace of itself such that* $\ker T^*$ *is located. Then the range of T is located.*

Proof. Let B denote the closed unit ball of H, and let P be the projection of H on $\ker T^*$. Since T is jointed, $T(nB)$ is located for each positive integer n (Theorem 6.3.4 of [6]). Given $x \in H$ and $\varepsilon > 0$, construct an increasing binary sequence λ such that

$$\lambda_n = 0 \Rightarrow \rho\,(x - Px, TT^*\,(nB)) > \frac{\varepsilon}{2},$$
$$\lambda_n = 1 - \lambda_{n-1} \Rightarrow \rho\,(x - Px, TT^*\,(nB)) < \varepsilon.$$

[7] Corollary 4.1.4 of [6] shows that every linear mapping T from a finite-dimensional Banach space into a normed space is bounded. When T is an operator on a finite-dimensional Hilbert space, the existence of T^* can be proved by applying the Riesz representation theorem in the standard way to the linear functionals of the form $x \rightsquigarrow \langle Tx, y\rangle$, since, by Corollary 4.1.8 of [6], the finite-dimensionality of H ensures that the norm of such a functional exists.

For each n with $\lambda_n = 0$, the separation and Riesz representation theorems (Corollary 5.2.10 and Theorem 4.3.6 of [6]) together provide us with a unit vector y_n such that

$$\langle x - Px, y_n \rangle > \langle TT^*z, y_n \rangle + \frac{\varepsilon}{2} \qquad (z \in nB);$$

taking $z = ny_n \in nB$, we obtain

$$n \left\| T^* y_n \right\|^2 < \left\langle TT^* \left(ny_n \right), y_n \right\rangle + \frac{\varepsilon}{2} < \langle x - Px, y_n \rangle \leqslant \left\| x - Px \right\|$$

and therefore $\left\| T^* y_n \right\| < n^{-1} \left\| x - Px \right\|$.

Assume that $\lambda = \mathbf{0}$. Then we obtain a sequence $(y_n)_{n \geqslant 1}$ of unit vectors such that $T^* y_n \to 0$, so T^* is an unopen mapping. It follows from Proposition 3.6 of [7] that $\operatorname{ran}(T^*)$ is infinite-dimensional,[8] which is absurd. Thus it is impossible that $\lambda = \mathbf{0}$. Now applying Markov's Principle, we obtain N such that $\lambda_N = 1$ and therefore $\rho\left(x - Px, TT^*(NB)\right) < \varepsilon$. Since $\varepsilon > 0$ is arbitrary and, by Lemma 6.5.3 of [6], $\operatorname{ran}(TT^*)$ is dense in $\operatorname{ran}(T)$, it follows that $\ker T^* + \operatorname{ran}(T)$ is dense in H; whence, by Lemma 6.5.2 of [6], $\operatorname{ran}(T)$ is located.

The preceding Proposition cannot be established without Markov's Principle. Indeed, given $a \in \mathbf{R}$ with $\neg(a = 0)$, we see that the linear mapping $T : x \rightsquigarrow ax$ on the Hilbert space $\mathbf{C}$ is bounded and selfadjoint, with kernel $\{0\}$. But if the range of T is located, then its distance from 1 cannot be positive and so is < 1; whence $a \neq 0$.

We now provide the proof of Theorem 2.

Proof. Under the hypotheses of Theorem 2, let $P = E \wedge F$, $Q = I - P$, and $K = Q(H)$. Note that K, being the range of a projection, is located in H and hence finite-dimensional ([6], Corollary 4.1.14). By Lemma 4, $QE|_K$ and $QF|_K$ are projections on K, the projections $QE|_K \vee QF|_K$ and $QE|_K \wedge QF|_K$ exist, and the latter equals 0. Hence, by Lemma 3,

$$\ker\left(Q\left(E - F\right)|_K\right) = \ker\left(QE|_K \vee QF|_K\right)$$

is located in K and so, again by Corollary 4.1.14 of [6], is finite-dimensional. Since we are working in **BISH** + **MP**, it follows from Proposition 3 that the range of $Q\left(E - F\right)|_K$ is located in K; whence it is finite-dimensional and therefore located in H. But for $x \in H$ we have

$$\begin{aligned}(E - F)x &= P\left(E - F\right)x + Q(E - F)x \\ &= \left(PE - PF\right)x + Q^2\left(E - F\right)x \\ &= \left(P - P\right)x + Q(E - F)Qx \\ &= Q(E - F)Qx = Q\left(E - F\right)|_K\left(Qx\right).\end{aligned}$$

[8] Up to this point, the proof requires only the existence of T^* and the locatedness of its kernel; it does not need H to be finite-dimensional.

Thus the range of $E - F$ equals that of $Q(E-F)|_K$ and so is located in H. Reference to Corollary 1 now completes the proof.

We have some final observations about our Markovian example. In that example, if $E \vee F$ exists, then, since $\neg(\theta = 0)$, the dimension of $\operatorname{ran}(E-F)$ must be 2; whence $\theta \neq 0$ and therefore $((EFE)^n)_{n\geqslant 1}$ converges strongly to $E \wedge F = 0$. Conversely, if that strong convergence obtains, then $\theta \neq 0$, so $E \vee F$ exists and equals $\mathbf{R}^2$. Thus we have

Proposition 4. *The following are equivalent:*

(i) *For all* 1*-dimensional projections* E, F *in* $\mathbf{R}^2$ *such that* $E \wedge F = 0$*, the sequence* $((EFE)^n)_{n\geqslant 1}$ *converges strongly to* 0*.*
(ii) *For all* 1*-dimensional projections* E, F *in* $\mathbf{R}^2$ *such that* $E \wedge F = 0$*, the supremum* $E \vee F$ *exists.*
(iii) **MP**.

In view of this proposition, it is reasonable to ask: can all reference to Markov's Principle be removed from Theorem 2?

Acknowledgement. *We thank Cristian Calude for many years of friendship and encouragement of our researches, and we wish him many fruitful, happy, and healthy years beyond 60!*

References

1. Aczel, P., Rathjen, M.: Notes on Constructive Set Theory, Report No. 40, Institut Mittag-Leffler, Royal Swedish Academy of Sciences (2001)
2. Aczel, P., Rathjen, M.: Constructive Set Theory (in preparation)
3. Bishop, E.A., Bridges, D.S.: Constructive Analysis. Grundlehren der Math. Wissenschaften, vol. 279. Springer, Heidelberg (1985)
4. Bridges, D.S., Ishihara, H.: Locating the range of an operator on a Hilbert space. Bull. London Math. Soc. 24, 599–605 (1992)
5. Bridges, D.S., Richman, F.: Varieties of Constructive Mathematics. London Math. Soc., Lecture Notes in Mathematics, vol. 97. Cambridge Univ. Press (1987)
6. Bridges, D.S., Vîţă, L.S.: Techniques of Constructive Analysis. Universitext. Springer, Heidelberg (2006)
7. Bridges, D.S., Julian, W.H., Mines, R.: A constructive treatment of open and unopen mapping theorems. Zeit. Math. Logik Grundlagen Math. 35, 29–43 (1989)
8. (Vîţă) Dediu, L.S.: The Constructive Theory of Operator Algebras, Ph.D. thesis, University of Canterbury, New Zealand (2000)
9. Halmos, P.R.: A Hilbert Space Problem Book. Grad. Texts in Math., vol. 19. Springer, Heidelberg (1974)
10. Ishihara, H.: Locating subsets of a Hilbert space. Proc. Amer. Math. Soc. 129(5), 1385–1390 (2001)
11. Kadison, R.V., Ringrose, J.R.: Fundamentals of the Theory of Operator Algebras, vol. I. Academic Press, New York (1983)

12. Martin-Löf, P.: An Intuitionistic Theory of Types: Predicative Part. In: Rose, H.E., Shepherdson, J.C. (eds.) Logic Colloquium 1973, pp. 73–118. North-Holland, Amsterdam (1975)
13. Myhill, J.: Constructive set theory. J. Symbolic Logic 40(3), 347–382 (1975)
14. Richman, F.: Adjoints and the image of the unit ball. Proc. Amer. Math. Soc. 129(4), 1189–1193 (2001)
15. Specker, E.: Nicht konstruktiv beweisbare Sätze der Analysis. J. Symbolic Logic 14, 145–158 (1949)

Bounded Randomness*

Paul Brodhead[1], Rod Downey[2], and Keng Meng Ng[3]

[1] Indian River State College, Fort Pierce, Florida
pbrodhea@irsc.edu
[2] School of Math, Statistics, & Operations Research, Victoria University
Cotton Building, Room 358, Gate 7, Kelburn Parade, Wellington, New Zealand
Rod.Downey@vuw.ac.nz
[3] School of Physical & Mathematical Sciences
Nanyang Technological University
21 Nanyang Link, Singapore
kmng@ntu.edu.sg

Abstract. We introduce some new variations of the notions of being Martin-Löf random where the tests are all clopen sets. We explore how these randomness notions relate to classical randomness notions and to degrees of unsolvability.

1 Introduction

The underlying idea behind algorithmic randomness is that to understand randomness you should tie the notion to computational considerations. Randomness means that the object in question avoids simpler algorithmic descriptions, either through effective betting, effective regularities or effective compression. Exactly what we mean here by "effective" delineates notions of algorithmic randomness. A major theme in the area of algorithmic randomness seeks to calibrate notions of randomness by varying the notion of effectivity. For example, classical Martin-Löf randomness[1] uses tests, shrinking connections of c.e. open sets whose measure is bounded by effective bounds, whereas Schnorr randomness has the tests of some precise effective measure. We then see that Schnorr and Martin-Löf randomness are related but can have very different properties; for example outside the high degrees they coincide, but the lowness concepts are completely disjoint.

Another major theme in the study of algorithmic randomness is the intimate relationship of randomness concepts with calibrations of computational power as given by measures of relative computability, like the Turing degrees. If something is random, can it have high computational power, for instance? A classic result in this area is Stephan's theorem [14] that if a Martin-Löf real is random and has

* Supported by the Marsden Fund of New Zealand. We wish to dedicate this to Cris Calude on the occasion of his 60th Birthday.

[1] We assume that the reader is familiar with the basic notions of algorithmic randomness as found in the early chapters of either Downey-Hirschfeldt [6] or Nies [13].

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 59–70, 2012.

enough computational power to be able to compute a $\{0,1\}$-valued fixed point free function then it must be Turing complete.

The goal of the present paper is to introduce some new variations in these studies, and to explore both themes. In particular, we will introduce what we call bounded variations of the notion of Martin-Löf randomness where the tests are all finite. These notions generalize the notion of Kurtz (or weak) randomness but are incomparable with both Schnorr and computable randomness.

More precisely, if W is a finite set then $\#W$ denotes the cardinality of W. $|\sigma|$ denotes the length of a finite string σ. We work in the Cantor space 2^ω with the usual clopen topology. The basic open sets are of the form $[\sigma]$ where σ is a finite string, and $[\sigma] = \{X \in 2^\omega \mid X \supset \sigma\}$. We fix some effective coding of the set of finite strings, and we freely identify finite strings with their code numbers. We denote $[W] = \cup\{[\sigma] : \sigma \in W\}$ as the Σ_1 open set associated with the c.e. set W. $\mu([W])$ denotes Lebesgue measure, and we write $\mu(W)$ instead of $\mu([W])$.

Definition 1. *(a) A Martin-Löf (ML) test is a uniform c.e. sequence $\{U_n\}_{n\in\omega}$ of sets U_n such that $\mu(U_n) < 2^{-n}$.*

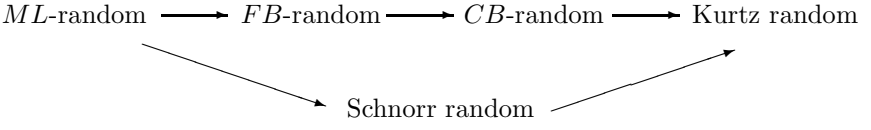
(b) A Martin-Löf test $\{U_n\}_{n\in\omega}$ is finitely bounded (FB) if $\#U_n < \infty$ for every n.

(c) A Martin-Löf test $\{U_n\}_{n\in\omega}$ is computably bounded (CB) if there is some total computable function f such that $\#U_n \leq f(n)$ for every n.

(d) A real $X \in 2^\omega$ passes a CB-test (FB-test) $\{U_n\}_{n\in\omega}$ if $X \notin \bigcap_n [U_n]$.

A real $X \in 2^\omega$ is computably bounded random *if X passes every CB-test. X is* finitely bounded random *if it passes every FB-test.*

These two notions of randomness are weaker than Martin-Löf randomness, although they imply Kurtz randomness. The obvious implications are:



No implications hold other than those stated in the diagram. This can be derived from the following facts: There is a Δ^0_3 1-generic real which is FB-random (see the remarks after Proposition 3), while no Schnorr random is weakly 1-generic. No incomplete c.e. degree can compute a FB-random (Proposition 1(i)) while some incomplete c.e. degree bounds a CB-random (Theorem 2). Lathrop and Lutz [12] showed that there is a computably random set X such that for every order function g, $K(X \upharpoonright n) \leq K(n) + g(n)$ for almost every n. Hence X cannot be CB-random, by Proposition 3.

There is a non-zero Δ^0_2 degree containing no CB-random (Theorem 2) while every hyperimmune degree contains a Kurtz random.

What is interesting is that these notions of randomness turn out to have strong relationships with degrees classes hitherto unrelated to algorithmic randomness.

We will show that FB-randomness and Martin-Löf -randomness coincide on the Δ^0_2 sets but are distinct on the Δ^0_3 sets (Theorem 1). There is some restriction on the degrees of these reals in that they cannot be c.e. traceable (Theorem 2). It is not clear exactly what the degrees of such reals can be.

In the case of CB-randomness there can be incomplete c.e. degrees containing such reals. We know that every c.e. degree contains a Kurtz random real, but the degrees containing a CB-random form a subclass of the c.e. degrees : those that are not totally ω-c.a.. This is a class of c.e. degress introduced by Downey, Greenberg and Weber [5] to explain certain "multiple permitting" phenomena in degree constructions such as "critical triples" in the c.e. degrees, and a number of other constructions as witnessed in the subsequent papers Barmpalias, Downey and Greenberg [1] and Downey and Greenberg [4]. This class extends the notion of array noncomputable reals, and correlates to the fact that all CB random reals have effective packing dimension 1 (Theorem 3). Downey and Greenberg [3] having previously showed that the c.e. degrees containing reals of packing dimension 1 are exactly the array noncomputable reals. We also show that if a c.e. degree $\mathbf{a}$ contains a CB random then every (not necessarily c.e.) degree above $\mathbf{a}$ contains a CB random as well. From all of this, we see that there remains a lot to understand for this class.

Some other results which space restrictions preclude us from including concern lowness for the classes we have introduced. We know that if A is K-trivial (i.e. low for Martin-Löf randomness) then A is low for FB-randomness. Also we know that if A is low for FB-randomness then A is Low(Ω). Finally in the case of CB-randomness, we know that if A is low for CB-randomness then A is of hyperimmune-free degree. However, we have a reasonably intricate construction which constructs a Δ^0_3 real which is low for CB-randomness.

2 Basic Results

We first show that the notions of FB-randomness and Martin-Löf -randomness coincide on the Δ^0_2 sets, and they differ on the Δ^0_3 sets.

Proposition 1. *(i) Suppose $Z \leq_T \emptyset'$. Then Z is ML-random iff Z is FB-random.*
(ii) There is some $Z \leq_T \emptyset''$ such that Z is FB-random but not ML-random.

Proof. (i): Given an approximation Z_s of Z, and suppose $\{U_x\}$ is the universal ML-test where $Z \in \cap_x [U_x]$. Enumerate an FB-test $\{V_x\}$ by the following: at stage s, enumerate into V_x, the string $Z_s \upharpoonright n$ for the least n such that $Z_s \upharpoonright n \in U_x[s]$. Then, $\{V_x\}$ is uniformly c.e., where $\mu(V_x) \leq \mu(U_x) < 2^{-x}$ for all x. Clearly $Z \in [V_x]$ for all x. We know $Z \upharpoonright n \in U_x$ for some least n, and let s be a stage such that $Z_s \upharpoonright n$ is correct and $Z \upharpoonright n$ has appeared in $U_x[s]$. Then, $Z \upharpoonright n$ will be in V_x by stage s, and we will never enumerate again into V_x after stage s.

(ii): We build $Z = \cup_s \sigma_s$ by finite extension. Let $\{U_x\}$ be the universal ML-test, and $\{V^e_x\}_x$ be the e^{th} ML-test. Assume we have defined σ_s, where for all $e < s$, we have

– all infinite extensions of σ_s are in U_e,
– if $\#V_x^e < \infty$ for all x, then there exists k such that no infinite extension of σ_s can be in U_k^e.

Now we define $\sigma_{s+1} \supset \sigma_s$. Firstly, find some $\tau \supseteq \sigma_s$ such that all infinite extensions of τ are in U_s; such τ exists because $\{U_e\}$ is universal. Let $k = |\tau|$. Next, ask if $\#V_k^s < \infty$. If not, let $\sigma_{s+1} = \tau^\frown 0$ and we are done. If yes, then figure out exactly the strings ρ_i such that $[V_k^s] = \cup\{[\rho_1], [\rho_2], \cdots, [\rho_n]\}$. We cannot have $[V_k^s] \supseteq [\tau]$ since $\mu(V_k^s) < 2^{-k}$, so there has to be some $\sigma_{s+1} \supset \tau$ such that $[\sigma_{s+1}] \cap [V_k^s] = \emptyset$, by the finiteness of V_k^s. We can figure σ_{s+1} out effectively from $\rho_1, \rho_2, \cdots, \rho_n$. Clearly the properties above continue to hold for σ_{s+1}. All questions asked can be answered by the oracle $\emptyset''$.

Note that there is no way of making $\{V_x\}$ computably bounded in (i), even if $Z \leq_{tt} \emptyset'$. It is easy to construct a low left c.e. real which is CB-random, while from Theorem 2 below, no superlow c.e. real can be CB-random. Hence CB-randomness and FB-randomness differ even on the c.e. reals.

CB-randomness is still sufficiently strong as a notion of randomness to exclude being traceable:

Proposition 2. *No CB-random is c.e. traceable.*

Proof. Suppose that A is c.e. traceable, and that A is coinfinite (otherwise we are done). We define the functional Φ by evaluating $\Phi^X(n)$ as σ where $\sigma \subset X$ is the shortest string such that $\#\{k : \sigma(k) = 1\} = 2n$, for any X and n. Since Φ^A is total, there is a c.e. trace $\{T_x\}_{x\in\mathbb{N}}$, such that $\#T_x \leq x$ and $\Phi^A(x) \in T_x$ for every x. We define the CB-test $\{U_x\}$ by the following: we enumerate σ into U_x if $|\sigma| \geq 2x$ and $\sigma \in T_x$. Then $\#U_x \leq x$ and $\mu(U_x) \leq x2^{-2x} < 2^{-x}$ for every x and $A \in \cap_x[U_x]$.

We next investigate the connection between CB-randomness and effective dimension.

Proposition 3. *Every CB-random is of effective packing dimension* 1.

Proof. Suppose $K(\alpha \upharpoonright n) \leq cn$ for all $n \geq N$ for some $N \in \mathbb{N}$ and $c < 1$ is rational. Fix a computable increasing sequence of natural numbers $\{n_i\}$ all larger than N, such that $n_i > \frac{i}{1-c}$ for all i. Now define a CB-test $\{V_i\}$ by the following: $V_i := \{\sigma \in 2^{n_i} \mid K(\sigma) \leq cn_i\}$. Here we have $\#V_i \leq 2^{cn_i}$.

In contrast, every incomplete c.e. real which is CB-random cannot be of d.n.c. degree (and hence has effective Hausdorff dimension 0). The proof of Theorem 1(ii) constructs a FB-random real by finite extensions. It is straightforward to modify the construction to build a Δ_3^0 FB-random which is 1-generic, and hence not of d.n.c. degree.

Next we investigate the upward closure of CB-random degrees.

Theorem 1. *If A is a c.e. real and is CB-random, and $A \leq_T B$, then $\deg_T(B)$ contains a CB-random.*

Proof. Fix a left-c.e. approximation A_s to A. Let $h : \mathbb{N} \mapsto \mathbb{N}$ be a strictly increasing function such that $h(n+1) \geq h(n)+2$ for every n. For any real X we let $(A \oplus_h X)(z)$ be defined by the following: if $z = h(n)$ for some n then $(A \oplus_h X)(z) = X(n)$, otherwise let $(A \oplus_h X)(z) = A(z-n-1)$ where $h(n) < z < h(n+1)$. This is the "sparse" join of A and X, and is obtained by copying the first $h(0)$ many digits of A followed by $B(0)$, the next $h(1)-h(0)-1$ many digits of A followed by $B(1)$, and so on. For numbers n, s we denote α^n_s as the finite string $A_s \upharpoonright h_s(n) - n$. This represents the A portion of the current approximation to $A \oplus_h X$ below $h_s(n)$.

The construction builds a function $h \leq_T A$ such that $A \oplus_h X$ is CB-random for any path $X \in 2^\omega$. This is achieved by specifying an effective approximation $h_s(n)$ which is non-decreasing in each variable n, s. We let $h(n) = \lim_s h_s(n)$. We also ensure that for every n, s if $h_{s+1}(n) > h_s(n)$ then $A_{s+1} \not\supset \alpha^n_s$. Intuitively $h_s(n)$ is the stage s coding location for $X(n)$, and we are insuring that before moving the coding location $h_s(n)$ we need to first obtain a change in α^n_s. The theorem is then satisfied by taking $A \oplus_h B$, for given $A \oplus_h B$ as oracle, to figure out $B(n)$, one can run the construction until a stage s is found such that α^n_s agrees with the true α^n of the oracle string. Then each of the coding location $h_s(0), \cdots, h_s(n)$ must already be stable at s.

Construction of h: Let $\{U^e_x\}$ be the e^{th} Martin-Löf test, and φ_e be the e^{th} partial computable function. We set $h_0(n) = 2n$ for every n. At stage $s > 0$ find the least $n < s$ such that $A_s \not\supset \alpha^n_{s-1}$, and there is some $e, x \leq n$ and some $\sigma \in U^e_x[s-1]$ such that $\varphi_e(x) \downarrow$ and $\#U^e_x[s-1] \leq \varphi_e(x)$. We also require that $\alpha^n_{s-1} \supseteq \sigma$ but $A_s \not\supset \sigma$. If such n is found we set $h_s(n+i) = s+n+2i$ for every i.

We now verify the construction works. Clearly h_s has the above-mentioned properties and $\lim_s h_s(n)$ exists. The only thing left is to check that $A \oplus_h X$ is CB-random. Suppose this fails for some $X \in 2^\omega$. Let $\{U_x\}$ be a CB-test such that $A \oplus_h X \in [U_x]$ for every x.

For a finite string σ and stage s, we let $\sigma^*(s)$ be the string obtained by removing the $(h_s(0)+1)^{th}, (h_s(1)+1)^{th}, \cdots$ digits from σ. We define a new CB-test $\{V_x\}$ by the following. At a stage s if we find some $\sigma \in U_{x^2}[s]$ and $\sigma^*(s) \subset A_s$ we enumerate $\sigma^*(s) \upharpoonright (h_s(x) - x)$ into V_x (unless some comparable string is already in V_x). That is, we enumerate the A-part of $A \oplus_h \emptyset$ below $h_s(x)$ into V_x, unless $\sigma^*(s)$ is shorter, in which case we enumerate $\sigma^*(s)$ instead.

We consider a large x. Clearly $\#V_x \leq \#U_{x^2}$, since each $\sigma \in U_{x^2}$ causes at most one $\sigma^*(s)$ (or part of) to be enumerated in V_x. We need to compute a bound on the measure of $[V_x]$. Each string enumerated into V_x is either $\sigma^*(s)$ or part of $\sigma^*(s)$ for some s and $\sigma \in U_{x^2}[s]$. Each string of the first type satisfies $|\sigma| - |\sigma^*(s)| \leq x$, while it is easy to see that strings of the second type must all be of different length greater than x. Hence the measure of $[V_x]$ is bounded by $2^x \mu(U_{x^2}) + 2^{-x+1} < 2^{-x}$. Since $\{V_x\}$ is a CB-test A must escape this test, a contradiction.

We conclude this section with several questions.

Question 1.
1. If $A \leq_T B$ and A is CB-random, must $deg_T(B)$ contain a CB-random?
2. Are there characterizations of CB-randomness and FB-randomness in terms of prefix-free complexity and martingales?
3. Are there minimal Turing degrees which contain CB-randoms?

3 A Characterization of the Left c.e. Reals Containing a CB-Random

The class of array computably c.e. sets was introduced by Downey, Jockusch and Stob [8,9] to explain a number of multiple permitting arguments in computability theory. Recall that a degree $\mathbf{a}$ is array non-computable[2] if for every function $f \leq_{wtt} \emptyset'$ there is a function $g \leq_T \mathbf{a}$ such that $f(x) < g(x)$ infinitely often. Downey, Greenberg and Weber [5] later introduced the totally ω-c.a.[3] sets to explain the construction needed for a weak critical triple, for which array non-computability seems too weak.

Definition 2 ([5]). *A c.e. degree* $\mathbf{a}$ *is totally* ω*-c.a. if every* $f \leq_T \mathbf{a}$ *is* ω*-c.e..*

Note that array computability can be viewed as a uniform version of this notion where the computable bound (for the mind changes) can be chosen independently of f; hence every c.e. array computable set is totally ω-c.e.. The class of totally ω-c.e. degrees capture a number of natural constructions. Downey, Greenberg and Weber [5] proved that a c.e. degree is not totally ω-c.e. iff it bounds a weak critical triple in the c.e. degrees.

In Theorem 2 we show that the non totally ω-c.e. degrees are exactly the class of c.e. degrees which permit the construction of a CB-random real:

Theorem 2. *Suppose A is a c.e. real. The following are equivalent.*

(i) $deg_T(A)$ is not totally ω-c.a.,
(ii) $deg_T(A)$ contains a CB-random,
(iii) There is some c.e. real $B \leq_T A$ which is CB-random,
(iv) There is some $B \leq_T A$ which is CB-random.

We fix a computable enumeration $\{\varphi_n\}_{n\in\omega}$ of all partial computable functions. We let $\{W_n^m\}_{n\in\omega}$ be the m^{th} Martin-Löf test. We use $<_L$ to denote the left-to-right lexicographical ordering on finite strings σ, τ, with 0 being to the left of 1 and $\sigma <_L \tau$ meaning that σ is to the left of τ. This ordering is extended naturally to $x <_L y$ for infinite strings x, y. We assume for any c.e. set U, that if $\sigma \in U_s$ then $|\sigma| < s$.

[2] This was not the original definition, but a later equivalent characterization, which is convenient for us to take as the definition.

[3] The original paper [5] called these *totally ω-c.e.*. However this terminology is somewhat at odds with Ershov's hierarchy of Δ_2^0 sets [10,11] and causes a problem when we work at various levels of the computable ordinals. Hence we will adopt the new name being used in Downey and Greenberg [4].

3.1 (i) ⇒ (iii)

Assume that $f = \Delta^A$ and that f is not ω-c.e. We will build $B \leq_T A$ and ensure that B is CB-random. We must ensure that $R_{m,i}$ holds for every m, i:

$$R_{m,i}: \quad B \notin \cap_n [W_n^m] \quad \text{if } \varphi_i \text{ is total and for all } n,\ \#W_n^m \leq \varphi_i(n).$$

To ensure that each requirement R is satisfied, suppose that R is the k^{th} requirement, where $k = \langle m, i \rangle$. Our construction will implement a sequence of modules $\{M_j^k\}_{j \in \omega}$ for R and each module is given infinitely many opportunities to act. At any particular stage, the construction attempts to satisfy at most one requirement through the implementation of at most one module. Associated with each module M_j^k is an integer $n = n_j^k$, and the module aims to ensure that if $\{W_e^m\}_{e \in \omega}$ is a CB-test then $B \notin [W_n^m]$ as follows. (Note that as long as some module succeeds, the requirement succeeds.)

Suppose at the current stage s of the construction that it is module M_j^k's turn to act and B is in $[W_n^m]$— that is, $B_{s-1} \in [W_{n,s}^m]$. The module's strategy is to redefine B to the right (outside of $[W_n^m]$), but on precondition that it receives an A-permission, due to certain conditions related to Δ^A.

To be more precise: throughout the construction, the modules $\{M_j^k\}_{j \in \omega}$ will collectively be defining an approximating function f_k for Δ^A towards ensuring that, for some j, module M_j^k's strategy succeeds (so that R_k is satisfied). We further discuss f_k and the A-permission below.

Module M_j^k is responsible for defining $f_k(j, s)$ for all s; it does so as follows. Whenever $B_{s-1} \in [W_n^m]$ as above, then— supposing this is the ${t_s}^{th}$ time it acts— M_j^k defines $f_k(j, t_s) := \Delta^A(j)[t_s]$. Module M_j^k waits to act at a later stage $q > s$ when either

- B remained in $[W_n^m]$ throughout all intermediate stages $\leq q$ and A changes below the use $\delta(j)$ for $\Delta^A(j)$, or
- B does not remain in $[W_n^m]$ until stage q due to an A-permission being granted to some other module, or perhaps some other requirement.

In either of these two cases, an A-permission is granted and M_j^k moves B to the right.

Now suppose $\{W_n^m\}$ is a CB-test so that $\#W_n^m \leq \varphi_i(n)$. Since B is only ever redefined to the right, it follows that there can be at most $\varphi_i(n) = \varphi_i(n_j^k)$ A-permissions associated with module M_j^k so that

$$\#\{s : f_k(j, s) \neq f_k(j, s+1)\} \leq \varphi_i(n) = \varphi_i(n_j^k).$$

It follows that if $B \in [W_{n_j^k}^m]$ for all j, then eventually no A-permission occurs for module M_j^k to act, for all j. Consequently, $f_k(j, t) = \Delta^A(j)[t] = f(j)$ for sufficiently large t and f_k must be an approximating function for $\Delta^A = f$. This means that f is ω-c.e., a contradiction, and thus requirement $R = R_k$ must be satisfied.

We are ready to describe the stage-by-stage construction.

Construction. The construction will proceed in stages of the form $\langle a+1, \langle j,k\rangle\rangle$. The intention is that stage $\langle a+1, \langle j,k\rangle\rangle$ is the a^{th} time in which module M_j^k is allowed to act. Consequently, in what follows, we will use ℓ to denote $\ell = \langle j,k\rangle$. We also define the integer $n_j^k = \langle k,j\rangle + 1$ associated with module M_j^k of the k^{th} requirement. Since Δ^A is total, we assume that $\Delta^A(j)[s]\downarrow$ at every stage $s > j$.

At stage $s = 0$, define $B_0 = 0^\omega$ and goto stage $s+1$.

At stage $s = \langle 0,\ell\rangle > 0$ define $f_k(j,0) = \Delta^A(j)[0]$ and goto stage $s+1$.

At stage $s = \langle a+1,\ell\rangle$, implement the j^{th} module M_j^k of requirement R_k defined as follows.

Module M_j^k.

1. If $\varphi_{i,s}(n_j^k)\uparrow$, or $\#W^m_{n_j^k,s} \not\leq \varphi_{i,s}(n_j^k)$, or $B_{s-1} \notin [W^m_{n_j^k,s}]$, then no non-trivial action is needed for M_j^k. We simply define $f_k(j,a+1) := f_k(j,a)$, define $B_s := B_{s-1}$ and go to stage $s+1$.
2. Otherwise, define $f_k(j,a+1) = \Delta^A(j)[a+1]$, let $r = \langle a,\ell\rangle$, and implement the following. If $A_{a+1}\upharpoonright\delta(j) \neq A_a\upharpoonright\delta(j)$, then do the following. Let $\sigma \subset B_{s-1}$ be maximal such that $N_\sigma := ([\sigma]\cap\{x : B_{s-1} <_L x\}) \setminus [W^m_{n_j^k,s}]$ is nonempty. Define B_s to be the left-most path of N_σ, and go to stage $s+1$. Otherwise define $B_s := B_{s-1}$ and go to stage $s+1$.

This completes the construction.

Verification. First observe that for any module M_j^k, whenever it changes B, it only adds an amount $q \in \mathbb{Q}$ to B_s where q can be accounted against a distinct part of $W^m_{n_j^k}$. Therefore M_j^k contributes at most $2^{-n_j^k}$ to B. Consequently the total effect of all the modules can contribute at most $\sum_{k,j\in\omega} 2^{-n_j^k} \leq \frac{1}{2}$ to B, which means that σ in the construction, at every stage, can always be found so that N_σ is non-empty.

Lemma 1. *Every requirement is satisfied.*

Proof. Suppose to the contrary that for some pair m,i, $B \in \cap_n[W_n^m]$, φ_i is total, and $\#W_n^m \leq \varphi_i(n)$ for all n. We first observe that $\lim_a f_k(j,a) = \Delta^A(j)$ for each j. Let $W = W^m_{n_j^k}$. Since $B \in [W]$, hence at almost every stage of the construction when M_j^k acts, we have case 2 holds; hence we will set $f_k(j,a) = \Delta^A(j)[a]$ at almost every a. Next, we want to show that the f_k-changes is bounded by $O(\varphi_i(n_j^k))$. We fix a j, and argue that if $\langle a_0+1,\ell\rangle < \langle a_1+1,\ell\rangle$ are two stages in the construction such that M_j^k acts under case 2, and $f_k(j,a_0+1) \neq f_k(j,a_1+1)$, then $B_s \notin [W_{\langle a_0+1,\ell\rangle}]$ for some $\langle a_0+1,\ell\rangle < s \leq \langle a_1+1,\ell\rangle$. This is because there must be some $a_0 < a \leq a_1$ such that $A_{a+1}\upharpoonright\delta(j) \neq A_a\upharpoonright\delta(j)$. At stage $\langle a+1,\ell\rangle$ of the construction we may assume case 2 holds (otherwise we are done). Hence we will define $B_{\langle a+1,\ell\rangle}$ to avoid $W_{\langle a+1,\ell\rangle} \supseteq W_{\langle a_0+1,\ell\rangle}$. This proves the claim. Now to see that the number of changes in $f_k(-,a)$ is bounded by $O(\varphi_i(n^k_-))$, observe that if $f_k(j,a) \neq f_k(j,a+1)$, we must have case 2 applies at stage $\langle a+1,\ell\rangle$ of the construction.

Lemma 2. $B \leq_T A$.

Proof. Next we describe how to compute $B \leq_T A$. To compute $B(x)$, we would like to say that only modules M_j^k for $n_j^k \leq x$ can change $B(x)$. This is unfortunately not true, because of the "carry-over" in the addition. Instead we have to compute B from A in a slightly more elaborate fashion. Define the total function $g \leq_A$ by the following. Let $g(0) = x$, and given $g(z)$ we define $g(z+1)$ by first searching recursively in A for some number a such that $A_a \upharpoonright \delta(g(z))$ is stable and correct. Let $g(z+1) = \max\{\langle a+1, \langle j,k\rangle\rangle \mid n_j^k \leq g(z)\}$. Hence the function g is defined so that after stage $g(z+1)$ of the construction, no module M_j^k for $n_j^k \leq g(z)$ can change B.

Assume we have computed $\sigma = B \upharpoonright x$. Now search for the least z such that either $B_{g(z+2)}(x) = 1$, or else $B_{g(z+2)}(y) = 0$ for some $x < y < g(z+1)$. This search will terminate because otherwise $B = \sigma 011111\cdots$ which means B is computable. Let z be the first found. If $B_{g(z+2)}(x) = 1$ then $B(x) = 1$. Otherwise we claim that $B(x) = 0$. After stage $g(z+2)$, only modules M_j^k for $n_j^k > g(z+1)$ can contribute to B, and the sum of their total contribution to B is $< 2^{-g(z+1)}$. On the other hand if $B_t(x) = 1$ at some $t > g(z+2)$, then the amount added to B after $g(z+2)$ is at least $2^{-x-1} - (2^{-x-2} + \cdots + 2^{-y-1}) = 2^{-y-1} \geq 2^{-g(z+1)}$.

3.2 (iv) ⇒ (i)

Suppose $B = \Delta^A$ and B is CB-random. Let φ_e be the e^{th} partial computable function. Fix a left c.e. approximation $\{A_s\}$ to A. Define $f(\langle e,k\rangle)$ by the following. Search for the first stage s such that $A_s \upharpoonright \delta(\langle e,k\rangle) = A \upharpoonright \delta(\langle e,k\rangle)$. If $\varphi_e(\langle e,k\rangle)[s] \uparrow$ then output $A \upharpoonright \delta(\langle e,k\rangle)$; otherwise output $A \upharpoonright \delta(\varphi_e(\langle e,k\rangle) + \langle e,k\rangle)$. Clearly f is total and $f \leq_T A$. Note that the use of the computation is not (and cannot be) computable. We claim f is not ω-c.e.; suppose the contrary we have $f(x) = \lim_s g(x,s)$ where $g(x,-)$ has at most $\varphi_e(x)$ mind changes for some total computable functions g and φ_e. We build a CB-test $\{V_k\}$ capturing B, contrary to assumption. For each k we find a stage s_0 such that $\varphi_e(\langle e,k\rangle)[s_0] \downarrow$, and $\Delta^A \upharpoonright \langle e,k\rangle[s_0] \downarrow$. We then enumerate $\Delta^A \upharpoonright \langle e,k\rangle[s_0]$ into V_k, and for every $s > s_0$ such that $\Delta^A \upharpoonright \langle e,k\rangle + \varphi_e(\langle e,k\rangle)[s] \downarrow$ with $g(\langle e,k\rangle, s) \supseteq A \upharpoonright \delta(\langle e,k\rangle + \varphi_e(\langle e,k\rangle))[s]$, we enumerate $\Delta^A \upharpoonright \langle e,k\rangle + \varphi_e(\langle e,k\rangle)[s]$ into V_k.

Clearly for each k we have $\#V_k \leq 1 + \varphi_e(\langle e,k\rangle)$, and that $\mu(V_k)$ is at most $2^{-\langle e,k\rangle} + \varphi_e(\langle e,k\rangle) 2^{-\langle e,k\rangle - \varphi_e(\langle e,k\rangle)} < 2^{-\langle e,k\rangle + 1} \leq 2^{-k}$. We claim that $B \in [V_k]$. At stage s_0 we threw in $\Delta^A \upharpoonright \langle e,k\rangle[s_0]$, and if $A \upharpoonright \delta(\langle e,k\rangle)$ is stable at s_0 then clearly $B \in [V_k]$. Since $\{A_s\}$ is a monotonic approximation to A, we therefore may assume that A was not stable at s_0, hence $f(\langle e,k\rangle) = A \upharpoonright \delta(\varphi_e(\langle e,k\rangle) + \langle e,k\rangle)$. Since g approximates f correctly, at some large enough stage we will enumerate $B \upharpoonright \langle e,k\rangle + \varphi_e(\langle e,k\rangle)$ into V_k.

Finally the proof of Theorem 2 is complete upon observing that (iii) implies (ii) follows from Theorem 1.

4 Lowness

Theorem 3. *There is a non-computable Δ^0_3 set A which is low for CB-randomness.*

Proof (Sketch of proof). The construction involves building a Δ^0_3 approximation to A. We will specify a computable approximation α_s and at the end we will take $A = \liminf_s \alpha_s$. We need to meet the requirements

$$P_e : A \neq \varphi_e$$
$$R_{e,i} : \text{If } \{U^A_{e,i}\}_{i\in\omega} \text{ is an } A\text{-relative } CB\text{-test with bound } \Psi^A_e, \text{ there is a } CB\text{-test } \{V_{e,i}\}_{i\in\omega} \text{ such that } \cap_{i\in\omega}[U^A_{e,i}] \subseteq \cap_{i\in\omega}[V_{e,i}]$$

Here we let $\{U^X_{e,i}\}$ be the e^{th} oracle CB-test, and Ψ_e be the e^{th} Turing functional. φ_e is the e^{th} partial computable function. The construction builds A of hyperimmune-free degree. For more details on how to construct a non-computable real of hyperimmune-free degree by a full Δ^0_3 approximation we refer the reader to Downey [2]. We sketch the main ideas here.

To make A of hyperimmune-free degree, for each Turing functional Ψ, we need to find a computable function δ that dominates Ψ^A. We begin by letting α_s be a string of zeroes. The aim is to build a perfect computable tree $T : 2^{<\omega} \mapsto 2^{<\omega}$ such that for every σ, $\Psi^{T(\sigma)}(|\sigma|) \downarrow$. We need to also ensure that A is in the range of T. If this fails then we will force Ψ^A to be non-total. In the former case we can read δ off T, and in the latter case we satisfy the requirement automatically. At every stage we let α_s extend $T(\sigma)$ for some σ of maximal length such that $T(\sigma)$ has been defined. If we never encounter a convergent Ψ^α we keep $\alpha \supset T(\sigma)$. If we find a convergent computation $\Psi^{\alpha_t\restriction u}(|\sigma|+1)$ at some stage t, we set $T(\sigma^\frown 0) \downarrow = \alpha_t \restriction u$ and move α to an incomparable string extending $T(\sigma)$ and search for a way to define $T(\sigma^\frown 1)$. In this way we define $T(\sigma)$ level by level, starting with $|\sigma| = 0$, and then $|\sigma| = 1$, and so on. It is clear that if we get stuck searching above some $T(\sigma)$ then $A = \liminf \alpha_s$ will extend $T(\sigma)$ and hence Ψ^A is not total. On the other hand if the procedure builds a total computable perfect tree T then $A = T(0^\omega)$. A lower priority requirement working for another Ψ' and believing in the totality of T will take the tree T as parameter and work to build a perfect subtree T' of T. A lower priority requirement working for P will be assigned a string σ_P in the domain of T, which is consistent with P's belief about the outcomes of higher priority requirements, and P will then later delete either $T(\sigma_P{}^\frown 0)$ or $T(\sigma_P{}^\frown 1)$ (or neither) depending on the value of φ_e.

When more requirements are considered it will become necessary to define A not as the direct lim inf of α_s, but as the lim inf with respect to the "true stages" of the construction, namely, the stages where the true path is visited.

How do we implement the R-requirements in this framework? Let us consider a top requirement working for R_0. It seeks to define a single CB-test $\{V_{0,i}\}_{i\in\omega}$ covering $\cap_{i\in\omega}[U^A_{0,i}]$. R_0 would pursue the abovementioned strategy to obtain a computable function dominating Ψ^A_0. Additionally it has to build the test $\{V_{0,i}\}_{i\in\omega}$. For $i = 0$ we wait for $T_0(\emptyset)$ to converge. If we ever discover some σ

entering $U^{\tau}_{0,0}$ with oracle string τ on T_0, we will delete every path on T_0 not extending τ, and enumerate σ into $V_{0,0}$. Since the cardinality of $U_{0,0}$ cannot exceed $\Psi_0^{T_0(\emptyset)}$, we will only act for $U_{0,0}$ finitely often, and succeed in making $V_{0,0} = U^A_{0,0}$.

Of course we cannot allow $U_{0,i}$ to delete paths in this way for every i, because we will end up with a computable path A. Suppose P is a requirement believing that R_0 has outcome ∞, i.e. R_0 succeeds in making T_0 total. The requirement P (and all other positive requirements) of lower priority will need to be assigned a diagonalization location σ_P. Suppose that P has been assigned σ_P for diagonalization. Each time $U_{0,0}$ acts as described above it will move σ_P. It is crucial to ensure that σ_P is moved only finitely often. We arrange for $U_{0,1}$ to respect σ_P, so $U_{0,1}$ will be prohibited from deleting prefixes of $T(\sigma_P{}^\frown 0)$ and $T(\sigma_P{}^\frown 1)$. If we consider infinitely many positive requirements $P_0 < P_1 < \cdots$ below R_0, we can arrange for a local priority ordering $U_{0,0} < P_0 < U_{0,1} < P_1 < U_{0,2} < \cdots$, where each $U_{0,i}$ has to respect $\sigma_{P_0}, \cdots \sigma_{P_{i-1}}$. This resolves the (potentially) infinitary conflicts between R_0 and lower priority P requirements. A computable bound for $V_{0,i}$ can then be easily computed from upperbounds for $\Psi_0^A(0), \cdots, \Psi_0^A(i)$.

Now consider requirements R_0, R_1 and P, where R_1 believes that R_0 has outcome ∞, and P believes that R_1 has outcome ∞. Say we arrange for the local priority ordering $U_{0,0} < U_{1,0} < P < U_{0,1} < \cdots$. Since R_0 cannot assume knowledge about the outcomes of the nodes of lower priority, $U_{0,1}$ cannot possibly wait for the tree T_1 to converge before fixing an upperbound for $V_{0,1}$. Furthermore $U_{0,1}$ has to respect σ_P, so we might enumerate a large number of elements into $V_{0,1}$ while α was extending $T(\sigma_P{}^\frown 0)$. Suppose α is next moved to extend $T(\sigma_P{}^\frown 1)$, and $U_{1,0}$ obtains an upperbound for $V_{1,0}$ after seeing T_1 grow. Now if we later discover some σ in $U^{\tau}_{1,0}$ with oracle $\tau \supseteq T(\sigma_P{}^\frown 1)$ we will have to move σ_P to make $T(\sigma_p) \supset \tau$, since $U_{1,0}$ is of higher local priority than P. This means that all the elements enumerated into $V_{0,1}$ so far are no longer possible elements of $U^A_{0,1}$, and the cardinality of $V_{0,1}$ has gone up unnecessarily. This wastage can be compounded each time $U_{1,0}$ moves σ_P, and since the bound for $V_{0,1}$ was computed with no knowledge of $\Psi_1^A(0)$, we might run out of space and exceed our declared cardinality bound for $V_{0,1}$.

Observe that we need not have fixed the local priority ordering beforehand. The solution is to assign the local priority of P *only when* P *is visited*. Let us consider the situation above again. Suppose P has not yet been visited by the construction (hence the local priority of P has not yet been decided). Suppose T_0 has been growing and we are currently waiting for T_1 to be defined at the root. At this point the local priority list reads $U_{0,0} < U_{1,0} < U_{0,1} < U_{0,2}, \cdots, < U_{0,i}$. If now $T_1(\emptyset)$ finds a definition, we will play outcome ∞ for R_0 and outcome ∞ for R_1, and visit P, who will now be queued after $U_{0,i}$.

The key point is that $U_{0,i+1}, U_{0,i+2}, \cdots$ will only be considered after this stage, so they can compute upperbounds for $V_{0,i+1}, V_{0,i+2}, \cdots$ using information about $\Psi_1^A(0)$. They are therefore safe from the actions of $U_{1,0}$ (and of course, from $U_{0,0}, \cdots, U_{0,i}$). On the other hand even though the upperbounds for $V_{0,0}, \cdots, V_{0,i}$ have been declared without any knowledge of $\Psi_1^A(0)$, they too,

are safe from the actions of $U_{1,0}$ because $U_{0,0}, \cdots, U_{0,i}$ are all allowed to move σ_P whenever we enumerate new elements into $V_{0,0}, \cdots, V_{0,i}$. The only downside is that σ_P gets injured a lot more times. Since the local priority of P once fixed, is never again changed, this means that σ_P will be eventually stable.

The interactions between other requirements present no new difficulty, and a formal construction proceeds in a more or less routine fashion. A complete proof will appear in the journal version of this paper.

References

1. Barmpalias, G., Downey, R., Greenberg, N.: Working with strong reducibilities above totally ω-c.e. degrees. Transactions of the American Mathematical Society 362, 777–813 (2010)
2. Downey, R.: On Π^0_1 classes and their ranked points. Notre Dame Journal of Formal Logic 32(4), 499–512 (1991)
3. Downey, R., Greenberg, N.: Turing degrees of reals of positive effective packing dimension. Information Processing Letters 108, 298–303 (2008)
4. Downey, R., Greenberg, N.: A Hierarchy of Computably Enumerable Degrees, Unifying Classes and Natural Definability (in preparation)
5. Downey, R., Greenberg, N., Weber, R.: Totally $< \omega$ computably enumerable degrees and bounding critical triples. Journal of Mathematical Logic 7, 145–171 (2007)
6. Downey, R., Hirschfeldt, D.: Algorithmic Randomness and Complexity. Springer, Berlin (2010)
7. Downey, R., Hirschfeldt, D., Nies, A., Terwijn, S.: Calibrating randomness. Bulletin of Symbolic Logic 3, 411–491 (2006)
8. Downey, R., Jockusch, C., Stob, M.: Array nonrecursive sets and multiple permitting arguments. In: Ambos-Spies, K., Muller, G.H., Sacks, G.E. (eds.) Recursion Theory Week. Lecture Notes in Mathematics, vol. 1432, pp. 141–174. Springer, Heidelberg (1990)
9. Downey, R., Jockusch, C., Stob, M.: Array nonrecursive degrees and genericity. In: Cooper, S.B., Slaman, T.A., Wainer, S.S. (eds.) Computability, Enumerability, Unsolvability. London Mathematical Society Lecture Notes Series, vol. 224, pp. 93–105. Cambridge University Press (1996)
10. Ershov, Y.: A hierarchy of sets, Part 1. Algebra i Logika 7, 47–73 (1968)
11. Ershov, Y.: A hierarchy of sets, Part 2. Algebra i Logika 7, 15–47 (1968)
12. Lathrop, J., Lutz, J.: Recursive computational depth. Information and Computation 153, 139–172 (1999)
13. Nies, A.: Computability and Randomness. Oxford University Press (in preparation)
14. Stephan, F.: Martin-Löf random sets and PA complete sets. In: Chatzidakis, Z., Koepke, P., Pohlers, W. (eds.) Logic Colloquium 2002, pp. 342–348. ASL and A. K. Peters, La Jolla (2006)

A Note on Blum Static Complexity Measures

Cezar Câmpeanu

Department of Computer Science and Information Technology
The University of Prince Edward Island, Canada
ccampeanu@upei.ca

Abstract. Dual complexity measures have been developed by Burgin, under the influence of the axiomatic system proposed by Blum in [3]. The concept of dual complexity measure is a generalization of Kolmogorov/Chaitin complexity, also known as algorithmic or static complexity. In this paper we continue this effort by extending some of the well known results for plain and prefix-free complexities to the general case of Blum universal static complexity. We also extend some results obtained by Calude in [9] to a larger class of computable measures, proving that transducer complexity is a dual (Blum static) complexity measure.

1 Introduction

The study of static, descriptional, or algorithmic complexity has been initiated by Ray J. Solomonoff [27,28], Andrey N. Kolmogorov [17], and Gregory J. Chaitin [10,11,12], and it was an algorithmic approach of information theory. For an introduction to Shannon's information theory [25] and its relation to algorithmic complexity, we refer the reader to [18,19,21].

Beside the plain complexity [17] and prefix complexity [12,16,20], there are several other forms of static or descriptional complexities considered in literature. We mention here only few of them: process complexity [26], monotone complexity [20], uniform complexity [22], Chaitin's complexity [14], or Solomonoff's universal complexity [27,28,29]. More examples of dual complexity are given in [4]. They often differ by a factor, but they have similar properties. Several variants were introduced and used to accommodate various needs, most notably being the effort of finding a complexity that can be used to define in an uniform way the randomness for both strings and sequences. Chaitin [11], and to some extent, Loveland [22], were successful in this attempt.

Manuel Blum proposed a set of simple axioms to measure the complexity of algorithms independently of the formal system [2], and few years later, a way of measuring the static complexity of algorithms in [3]. The first approach given in [2] is used in dynamic complexity and represents the basis for the Computational Complexity Theory. The latest approach was further developed by Burgin as a generalized Kolmogorov complexity, and is called dual complexity measure in [4]. It must be noted that the axioms proposed by Blum in 1967 in [3] "are all so fantastically weak that any reasonable model of a computer and any reasonable definition of size and step satisfies them" [3]. Thus, it is natural to ask what are

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 71–80, 2012.

the general properties of static complexities (as defined by Blum) and what kind of results can be proved for all "reasonable" static complexity measures.

If direct complexity measures estimate algorithms or programs, dual complexity measures are properties of objects that are constructed and processed by algorithms or programs. Further results for inductive algorithms were recently obtained in [5,6]. The axioms considered in these papers are the same as the ones used by Blum in [3] plus few other very simple ones, and they allow us to develop several hierarchies of complexities [24]. Using small modifications of the original idea, we can also define generalized complexity classes as in [1].

In this paper we prove a weak form of universality theorem for dual complexity measures in section 3, a strong version for universal encodings in section 3.2, and reprove in the new more general framework of Blum Static Complexity measures some well known results in section 4, thus validating the theory.

2 Notations and Definitions

We denote by $\mathbb{N} = \{0, 1, \ldots\}$ the set of natural numbers. For a set T, we denote by $\#T$ its cardinal.

For a finite alphabet with p letters, we use the set $A_p = \{0, 1, \ldots, p-1\}$. The free monoid generated by A_p is A_p^*.

The length of a word $w = w_1 \ldots w_n \in A_p^*, w_i \in A_p, 1 \leq i \leq n$, is $|w| = n$. The set A_p^* can be ordered using the quasi-lexicographical order: $\varepsilon, 0, 1, \ldots, p-1, 00, 01, 0(p-1), 10, \ldots$. We denote by $string(n)$ the one to one function between $\mathbb{N}$ and A_p^* representing the n-th string of A_p^* in the quasi-lexicographical order. Thus, $string(0) = \varepsilon$, $string(1) = 0, \ldots string(p) = p-1$, $string(p+1) = 00, \ldots$.

The set of strings $w \in A_p^*$ of length equal, less than, less then or equal to, greater than, and greater then or equal to n is denoted by: A_p^n, $A_p^{<n}$, $A_p^{\leq n}$, $A_p^{>n}$, and respectively, $A_p^{\geq n}$.

We consider an acceptable enumeration of the set of all partial computable functions over $\mathbb{N}$, $\mathcal{F} = (\phi_i^{(n)})_{i \in \mathbb{N}}$, i.e., an enumeration satisfying the Wagner-Strong axioms [23]. A function $< \cdot, \cdot >: \mathbb{N} \times \mathbb{N} \longrightarrow \mathbb{N}$ is a pairing function if it is bijective, and its inverses $(\cdot)_1, (\cdot)_2 : \mathbb{N} \longrightarrow \mathbb{N}$ satisfy the following properties:

1. $< (z)_1, (z)_2 >= z$;
2. $(< x, y >)_1 = x$, $(< x, y >)_2 = y$.

In what follows we will use only computable pairing functions. It is a common practice to use paring functions to extend unary functions to functions having more than one argument by defining : $\phi_i^{(2)}(x, y) = \phi_i^{(1)}(< x, y >)$, and to consider that indexes of algorithms are encodings of algorithms over a finite alphabet A_p. In case $p = 2$, the encoding is over the binary alphabet.

We refer the reader to [8,15] for more on computability, computable functions, and recursive function theory.

In [4,5], the definitions of direct and dual complexity measures are stated in a formalism similar with what follows.

Let $\mathcal{G} \subseteq \mathcal{F}$, $\mathcal{G} = (\phi_i^{(n)})_{i \in I}$ be a class of algorithms. A function $m : I \longrightarrow \mathbb{N}$ is called a (direct) complexity measure [3,5] if satisfies:

1. (Computational axiom) m is computable;
2. (Re-computational Axiom) the set $\{j \mid m(j) = n\}$ is computable;
3. (Cofinitness Axiom) $\#\{j \mid m(j) = n\} < \infty$.

In [5], additional axioms are considered for defining axiomatic complexity measures:

4. (Re-constructibility Axiom) For any number n, it is possible to build all algorithms A from $\mathcal{G}$ for which $m(A) = n$.
5. (Compositional Axiom) If $A \subseteq B$, then $m(A) \leq m(B)$.

Since the relation " $\subseteq$" between algorithms is usually defined depending on some encoding of the algorithm, for the moment we will only consider axioms 1–4.

Definition 1. *A space $(\mathcal{G}, m)$ satisfying axioms 1–4 is called Blum static complexity space.*

However, the axioms considered in [5], beside the axioms 1–5, are obviously satisfied by most usual measures.

Definition 2. *[5] Let $d : \mathbb{N} \longrightarrow \mathbb{N}$ be a function. An algorithm $U : \mathbb{N} \times \mathbb{N} \stackrel{\circ}{\longrightarrow} \mathbb{N}$ is called d-universal for the set $\mathcal{G} = (\psi_i)_{i \in I}$, if $\psi_i(n) = U(d(i), n)$, for all $i \in I$ and $n \in \mathbb{N}$.*

If U is a two argument universal algorithm for the algorithms with one argument, i.e., $U(i, x) = \psi_{univ_1}(i, x) = \psi_i(x)$, then U is $1_{\mathbb{N}}$-universal for $\mathcal{G}$.

Given a complexity measure $m : I \longrightarrow \mathbb{N}$ and $\psi \in \mathcal{G}$, the dual to m with respect to ϕ is[1] $m_\psi^0(x) = \min\{m(y) \mid y \in I, \psi(y) = x\}$[2].

In what follows we set the convention that a function $f : \mathbb{N}^k \longrightarrow \mathbb{N}$, when applied to a string $w \in A_p^*$ is in fact the function f applied to $string^{-1}(w)$, and when the result is in A_p^*, we apply the function $string$ to the result of f. Thus, we do not distinguish between a number $n \in \mathbb{N}$ and its representation $string(n) \in A_p^*$.

If I is the encoding of algorithms over A_p, then $m(I)$ is usually the length of the encoding; in this case, m_ϕ^0 is called dual to length complexity of x, with respect to the algorithm ϕ. [3]

Since the measure m satisfies the Cofinitness Axiom 3, we can define the maximum complexity of objects having a certain measure as:

$$\Sigma_\phi^{\mathcal{G}}(n) = \max\{m_\phi^0(x) \mid m(x) = n\}.$$

[1] $min\emptyset = \infty$.

[2] We can consider that I is embedded in $\mathbb{N}$.

[3] In this case, if we would not use the convention just established above, we would write $\phi(string^{-1}(y)) = x$ instead of $\phi(y) = x$.

Example 1. If $m(y) = |y|$, we analyze the dual to the length complexity measure for the following classes:

1. $\mathcal{G} = \{f \in \mathcal{F} \mid, dom(f) \text{ is prefix free}\}$,
2. $\mathcal{G} = \mathcal{F}$.

In both cases, we have $I = A_p^*$.

$m_\phi^0(x) = \min\{|y| \mid \psi(y) = x\}$, which is exactly the definition of prefix, or plain complexity of function $\psi \in \mathcal{G}$.

Both classes have universal d-algorithms, as follows:

- Let $e : \mathbb{N} \longrightarrow A_p^*$ be an enumeration of prefix free functions $\psi_i(y) = \phi_{e(i)}(y)$. Then $\psi_i(y) = \phi_{e(i)}(y) = \phi_{univ_1}(e(i), y)$. Here we have two options:
  1. $U(i, y) = \phi_{univ_1}(e(i), y)$ and $d(i) = i$ $(U \notin \mathcal{G})$,
  2. $U(i, y) = \phi_{univ_1}(i, y)$ and $d(i) = e(i)$. In this later case $U \in \mathcal{G}$, $\psi_i = \phi_{e(i)}$.
- $\phi_i(y) = \psi_{univ_1}(i, y) = U(d(i), y)$, $U = \psi^{(2)}_{univ_1}$, $d(i) = i$, $\phi_i = \psi_i$.

We set

$$C^{\mathcal{G}}(x) = \inf_{i \in I, y \in I} \{m(i) + m(y) \mid \psi_i(y) = x\} \tag{1}$$

and

$$C^{\mathcal{G}}_{\tau}(x) = \inf_{y \in I} \{m(y) \mid \psi(y) = x\}. \tag{2}$$

The superscript $\mathcal{G}$, may be ignored when it is understood from the context and we write $C_\psi(x) = C^{\mathcal{G}}_\psi(x)$, and $C(x) = C^{\mathcal{G}}(x)$.

If there exists i_0 such that for all $i \in I$, there is a $c \in \mathbb{N}$ such that:

$$C_{\psi_{i_0}}(x) \leq C_{\psi_i}(x) + c, \tag{3}$$

then the algorithm ψ_{i_0} is an universal algorithm for the family $\mathcal{G}$.

In case we do have an universal algorithm i_0 for $\mathcal{G}$, it follows that: $C^{\mathcal{G}}(x) \leq C^{\mathcal{G}}_{i_0}(x) + m(i_0) \leq C^{\mathcal{G}}_i(x) + m(i_0) + c(i_0, i)$, for all $i \in I$. Hence, $C^{\mathcal{G}}(x) = C^{\mathcal{G}}_{i_0}(x) + O(1)$.

Lemma 1. *Let ψ be an universal algorithm for $\mathcal{G}$. Then*

$$C(x) = C_\psi(x) + O(1).$$

In [5] it is shown that all universal algorithms for a class $\mathcal{G}$ differ by a constant, result that is now a consequence of Lemma 1.

For cases when the universal algorithm cannot be one of the algorithms of the family considered, we use the complexity as it is defined by (1). For all other cases it is more convenient to use the complexity to be equal to C_ψ.

In the next section we study some conditions for the existence and computability of universal algorithms.

3 Universal Algorithms

In this section we study the existence of universal algorithms for a given Blum static complexity space. In the first part we study the existence of universal algorithms for subsets of total computable functions, while in the second part we focus our attention on family of functions having the computational power comparable to Turing Machines.

3.1 Families of Totally Defined Functions

We prove that for families of functions for which the growth of the complexity is bounded by a linear function, we do not have universal algorithms. Also, these families are considered over A_p^*, where the measure m is the usual length $|\cdot|$.

Let $\mathcal{T} = (\tau_i)_{i\in I}$ be a set of computable, total functions over A_p^*, and $<\cdot,\cdot>$: $\mathbb{N}^2 \longrightarrow \mathbb{N}$, be a paring function.

Theorem 1. *If $\mathcal{T}$ is a family of total computable functions satisfying the following two properties:*

1. *for every $\tau \in \mathcal{T}$, there exists a function B_τ such that*

$$|\tau(xy)| \leq |\tau(x)| + B_\tau(|y|),$$

2. *for every $M > 0$, there is $i \in I$ and x such that $|\tau_i(x)| > |x| \cdot M$,*

then there is no universal function for $\mathcal{T}$ in $\mathcal{T}$.

Proof. Using the first property, we deduce that for every $\tau \in \mathcal{T}$, there is $B_\tau \in \mathbb{N}$ such that

$$|\tau(x)| \leq B_\tau(1) \cdot |x| + O(1).$$

Assume there is an universal function τ_{i_0} for $\mathcal{T}$, i.e., for every $x \in A_p^*$, there is $x' \in A_p^*$, such that: $|x'| \leq |x| + c$ and $\tau_{i_0}(x') = \tau_i(x)$. Thus, using the hypothesis of the second condition of our theorem, for any $M > 0$, there is an i such that: $M \cdot |x| \leq |\tau_i(x)| = |\tau_{i_0}(x')| \leq |x'| \cdot B_{i_0} + c_{i_0} \leq (|x| + c) \cdot B_{i_0} + c_{i_0}$, for some constant c_{i_0} and $B_{i_0} = B_{\tau_{i_0}}(1)$. Hence,

$$M \cdot |x| \leq |x| \cdot B_{i_0} + c \cdot B_{i_0} + c_{i_0},$$

holds for all $x \in A_p^*$. But this is true for all values of M, including $2B_{i_0}$. Hence, $B_{i_0}|x| \leq c \cdot B_{i_0} + c_{i_0}$, i.e., $|x| \leq \frac{c \cdot B_{i_0} + c_{i_0}}{B_{i_0}}$, which cannot be true for all x. □

It is relatively easy to check that functions that are realized by functional transducers satisfy the above theorem (see [9], for example). Although we may have an universal function U for $\mathcal{T}$, that function U cannot be one of the functions in $\mathcal{T}$.

Thus, Theorem 9 in [9] is a corollary of Theorem 1.

Definition 3. *We say that a family of computable functions encodes algorithm ψ_i if there exists $i_0 \in I$, and a computable function $E : A_p^* \longrightarrow A_p^*$, such that for every $x \in A_p^*$, there exists $u = E(x) \in A_p^*$, with $\tau_{i_0}(u) = \psi_i(x)$.*

The complexity of the family $\mathcal{T}$ is defined by (1), the following theorem is a straightforward generalization of Theorem 13 of [9], and the proof is obvious.

Theorem 2. *For any $\tau_i \in \mathcal{T}$, we have that:*

$$C(x) \leq C_{\tau_i}(x) + m(i).$$

Corollary 1. *If identity function can be encoded by $\mathcal{T}$, then the complexity C is computable.*

Proof. It is enough to observe that $C(x)$ is bounded by the sum $m(E(x))+m(i_0)$, where E is the encoding for the identity function $\phi_{j_0} = 1_{A_p^*}$, $\phi_{j_0}(x) = x$. We then use the Cofinitness Axiom and Re-computational Axiom to compute C. □

As we could see, all these results are generalization of the ones obtained by Cristian Calude in [9], where it is implicit proved that the class of functions computed by deterministic sequential transducers is a Blum static complexity space.

3.2 Encodings of Computable Functions

Very often, properties of Kolmogorov complexity are proved for a specific encoding (usually, either plain or prefix-free version) by using a very specialized construction that works only for that model. Furthermore, many authors claim that the property should be valid for any other encoding, without any proof. To close this loophole, I think is important to check what should be the properties of the encoding that preserves most of the results, and what are the properties of a certain encoding that allow us to differentiate it from other encodings.

In this section we start this study by proposing simple properties of encodings as follows.

Let e and E be two computable functions satisfying the following properties:

1. E is injective and is a length increasing function in the second argument, i.e., there exists c_e, such that if $|x| \leq |y|$, then $|E(i,x)| \leq |E(i,y)| + c_e$.
2. $|E(i,x)| \leq |E(i',x)| + \eta(i,i')$, for some function $\eta : \mathbb{N}^2 \longrightarrow \mathbb{N}$.

Definition 4. *We say that the family $\mathcal{G} = (\psi_j)_{j\in J}$ is an (e,E)-encoding of the family $\mathcal{H} = (\mu_i)_{i\in I}$, if for every $i \in I$ and all $x \in \mathbb{N}$, we have that:*

1. *$\mu_i(x) = \psi_{e(i)}(E(i,x))$, for all $i \in I$ and $x \in \mathbb{N}$,*
2. *if $\psi_j(z) = x$, then $e(i) = j$ and $E(i,y) = z$, for some $i \in I$ and $y \in \mathbb{N}$.*

Theorem 3. *If $\mathcal{H}$ has an universal algorithm in $\mathcal{H}$, and $\mathcal{G}$ is an encoding of $\mathcal{H}$, then $\mathcal{G}$ has an universal algorithm in $\mathcal{G}$.*

Proof. Let us assume that $\mu_i(y) = x$. Then, using 2 of Definition 4, it follows that $j = e(i)$, $y = E(i, z)$, and $\mu_i(z) = x$. Using the universality property for $\mathcal{H}$, we get: $\mu_{i_0}(z') = x$ for some z' such that $|z'| \leq |z|+c$. Thus, $\psi_{e(i_0)}(E(i_0, z')) = x$, and

$$|E(i_0, z')| \leq |E(i, z')| + \eta(i_0, i) \leq |E(i, z)| + \eta(i_0, i) + c_e.$$

But $\psi_i(z) = x$, and $\psi_{e(i_0)}(E(i_0, z')) = x$. □

A Blum static complexity space $\mathcal{G}$ closed under composition, which encodes $\mathcal{F}$, is called Blum universal complexity measure. It is easy to check that any encoding of Turing machine model is a Blum universal complexity measure. In case $\mathcal{G}$ encodes $\mathcal{F}$, then $(\mathcal{G}, m)$ is called Blum universal static complexity space (BUSC). One can check that (Prefix-free) Turing Machines, together with their sizes, form a Blum universal static complexity space. Thus, it is natural to check if common properties of plain and prefix-free versions of Kolmogorov-Chaitin complexity can be proved in the general context of Blum universal static complexity.

4 Properties of Blum Static Universal Complexity Measures

In this section we verify if general properties of Kolmogorov-Chaitin complexity hold for the general case of a Blum universal static complexity space. Even though most proofs make extensive use of particular properties of the object constructed (like in [7], for example), the expectation would be that most of the proofs for the general case will just resemble previous proofs for Kolmogorov-Chaitin complexity. However, there are few technicalities that must be taken care of, which are essential in proving the new results.

Let us fix a Blum universal static complexity space $(\mathcal{G}, m)$ with universal algorithm ψ_{i_0}.

Theorem 4. *The set* $\mathcal{C}_t = \{x \in A_p^* \mid C^{\mathcal{G}}(x) \geq m(x) - t\}$ *is immune.*

Proof. Assume by absurd that the set is not immune, and we can enumerate an infinite part $D \subseteq \mathcal{C}_t$. We define the function:

$F(0^i1) = \min\{x \in D \mid m(x) > m(E(i, 0^i1)) + 2i + 1 + t\}$,

thus $F = \phi_j$. Now $\phi_j(0^i1) = \psi_{e(i)}(E(i, 0^i1)$.

But $F(0^i1) \in D \subseteq \mathcal{C}_t$, thus

$C^{\mathcal{G}}(F(0^i1)) \geq m(F(0^i1))) - t > m(E(i, 0^i1)) + 2i + 1 + t - t = m(E(i, 0^i1)) + 2i + 1$.

Hence, for an infinity of i's, we have

$m(E(i, 0^i1)) + 2i + 1 < C^{\mathcal{G}}(F(0^i1)) \leq C^{\mathcal{G}}_{e(i)}(F(0^i1)) + c \leq m(E(i, 0^i1)) + c$,

i.e., $2i + 1 < c$, which is impossible. □

Corollary 2. *The function* $C^{\mathcal{G}}$ *is not computable.*

Proof. If $C^{\mathcal{G}}$ is computable, so is the predicate: $P(x) = 1$ iff $C^{\mathcal{G}}(x) > m(x) - t$. Then $\mathcal{C}_t$ is not immune. □

For $x \in \mathbb{N}$, if y is such that $m(y) = C^{\mathcal{G}}(x)$ and $y = \min\{z \in \mathbb{N} \mid \psi_{i_0}(z) = x\}$, then y is called the canonical program of x and it is denoted by x^*.

The canonical program of x is $x^* = \min\{y \in A_p^* \mid \psi_{i_0}(y) = x\}$.

Corollary 3. *The set of canonical programs is immune and the function $f(x) = x^*$ is not computable.*

Theorem 5. *The set of canonical programs CP is immune.*

Proof. We fix the universal computer $\psi \in \mathcal{G}$. Since $(\mathcal{G}, m)$ is Blum universal, we can construct the following computer: $D(u) = \psi(\psi(u)) = \psi_{j_0}(u)$.

If $z = x^*$ and $x = y^*$, then we have:

$D(z) = \psi(\psi(z)) = \psi(\psi(x^*)) = \psi(x) = \psi(y^*) = y$.

Hence, $C_D^{\mathcal{G}}(y) \leq m(z) = m(x^*) = C^{\mathcal{G}}(x)$, and

$m(x) = m(y^*) = C^{\mathcal{G}}(y) \leq C_D^{\mathcal{G}}(y) + c \leq C^{\mathcal{G}}(x) + t$, for some constant t. This means that if $x \in CP$, then $C^{\mathcal{G}}(x) \geq m(x) - t$, i.e., $CP \subseteq \mathcal{C}_t$. But $\mathcal{C}_t$ is immune, thus CP must be also immune. □

We say that a string is t-compressible if $C(x) < \Sigma(|x|) - t$, and that is t-incompressible if $C^{\mathcal{G}}(x) \geq \Sigma(|x|) - t$. A t-incompressible element is also called random in $(\mathcal{G}, m)$ and the set of all these elements is denoted by $RAND_t^{\mathcal{G}}$.

Given the fact that $\#\{x \in A_p^n \mid C^{\mathcal{G}}(x) < n - t\} \leq \frac{p^{n-t}-1}{p-1} < p^n$, it follows that $\Sigma^{\mathcal{G}}(n) \geq n$, regardless of the class of algorithms considered and the complexity measure chosen.

Corollary 4. *The set $RAND_t^{\mathcal{G}}$ is immune.*

Proof. $RAND_t^{\mathcal{G}} \subseteq \mathcal{C}_t$. □

5 Conclusion

In this paper we show that the original Blum axioms, with minor additions, can be used to prove results for static complexity for both computable and uncomputable measures. The existence of the Universal algorithm is guaranteed in any models encoding Turing Machines. We identify a class of families of algorithms without universal algorithms, but having a computable complexity; this class contains the family of algorithms computed by deterministic sequential transducers.

It is interesting to see that defining randomness in terms of minimal description size (MDS), most results on random strings are recovered for the case of BUSC. We expect that Solovay's theorems [30] relating plain to prefix-free complexity, can be proved in a more general framework, thus we can unveil the way of uniform usage of definitions for random strings and random sequences.

Acknowledgement. To Cristian Calude, for introducing me to this area of research and encouraging this approach.

References

1. Balcázar, J.L., Ronald, V.: Book On Generalized Kolmogorov Complexity. In: Monien, B., Vidal-Naquet, G. (eds.) STACS 1986. LNCS, vol. 210, pp. 334–340. Springer, Heidelberg (1985)
2. Blum, M.: A machine-independent theory of the complexity of recursive functions. Journal of the ACM 14(2), 322–336 (1967)
3. Blum, M.: On the size of machines. Information and Control 11, 257–265 (1967)
4. Burgin, M.: Generalized Kolmogorov complexity and other dual complexity measures. Translated from Kibernetica 4, 21–29 (1990); Original article submitted June 19 (1986)
5. Burgin, M.: Algorithmic complexity of recursive and inductive algorithms. Theoretical Computer Science 317, 31–60 (2004)
6. Burgin, M.: Algorithmic complexity as a criterion of unsolvability. Theoretical Computer Science 383, 244–259 (2007)
7. Calude, C.: Information and Randomness - An Algorithmic Perspective. Springer, Berlin (1994)
8. Calude, C.: Theories of Computational Complexity. North-Holland, Amsterdam (1988)
9. Calude, C., Salomaa, K., Roblot, T.K.: Finite State Complexity and Randomness, Technical Report CDMTCS 374 (December 2009/revised June 2010)
10. Chaitin, G.J.: On the Length of Programs for Computing Finite Binary Sequences. J. ACM 13(4), 547–569 (1966)
11. Chaitin, G.J.: On the Length of Programs for Computing Finite Binary Sequences: statistical considerations. J. ACM 16(1), 145–159 (1969)
12. Chaitin, G.J.: A Theory of Program Size Formally Identical to Information Theory. J. ACM 22(3), 329–340 (1975)
13. Chaitin, G.J.: A Theory of Program Size Formally Identical to Information Theory. J. ACM 22(3), 329–340 (1975)
14. Chaitin, G.J.: Algorithmic Information Theory, Cambridge Tracts in Theoretical Computer Science, vol. I. Cambridge University Press (1987)
15. Davis, M., Sigal, R., Weyuker, E.: Computability, Complexity, and Languages, 2nd edn. Academic Press, New York (1994)
16. Gacs, P.: On the symmetry of algorithmic information. Soviet Mathematics Doklady 15, 1477–1480 (1974)
17. Kolmogorov, A.N.: Problems Inform. Transmission 1, 1–7 (1965)
18. Kolmogorov, A.N.: Three approaches to the definition of the quantity of information. Problems of Information Transmission 1, 3–11
19. Kolmogorov, A.N.: Combinatorial foundations of information theory and the calculus of probabilities. Russian Mathematical Surveys 38(4), 27–36 (1983)
20. Levin, L.A.: Laws of information conservation (non-growth) and aspects of the foundation of probability theory. Problems of Information Transmission 10(3), 206–210 (1974)
21. Levin, L.A., Zvonkin, A.K.: The Complexity of finite objects and the Algorithmic Concepts of Information and Randomness. Russian Math. Surveys 25(6), 83–124 (1970)
22. Loveland, D.A.: On Minimal-Program Complexity Measures. In: STOC, pp. 61–65 (1969)
23. Papadimitriou, C.H., Lewis, H.: Elements of the theory of computation. Prentice-Hall (1982); 2nd edn. (September 1997)

24. Schmidhuber, J.: Hierarchies of generalized Kolmogorov complexities and nonenumerable universal measures computable in the limit. International Journal of Foundations of Computer Science 13(4), 587–612 (2002)
25. Oliver, B.M., Pierce, J.R., Shannon, C.E.: The Philosophy of PCM. Proceedings Institute of Radio Engineers 36, 1324–1331 (1948)
26. Schnorr, C.-P.: Process complexity and effective random tests. Journal of Computer and System Sciences 7(4), 376–388 (1973)
27. Solomonoff, R.J.: A Formal Theory of Inductive Inference, Part I. Information and Control 7(1), 1–22 (1964)
28. Solomonoff, R.J.: A Formal Theory of Inductive Inference, Part II. Information and Control 7(2), 224–254 (1964)
29. Solomonoff, R.J.: Complexity-Based Induction Systems: Comparisons and Convergence Theorems. IEEE Trans. on Information Theory IT-24(4), 422–432 (1978)
30. Solovay, R.M.: Draft of paper (or series of papers) on Chaitin's work. Unpublished notes, 1–215 (May 1975)

A Program-Size Complexity Measure for Mathematical Problems and Conjectures

Michael J. Dinneen

Department of Computer Science, University of Auckland, New Zealand
mjd@cs.auckland.ac.nz

Abstract. Cristian Calude et al. in [5] have recently introduced the idea of measuring the degree of difficulty of a mathematical problem (even those still given as conjectures) by the length of a program to verify or refute the statement. The method to evaluate and compare problems, in some objective way, will be discussed in this note. For the practitioner, wishing to apply this method using a standard universal register machine language, we provide (for the first time) some "small" core subroutines or library for dealing with array data structures. These can be used to ease the development of full programs to check mathematical problems that require more than a predetermined finite number of variables.

1 Introduction

In mathematics, when working on (or deciding to study) a new open problem or conjecture, one often wonders how difficult to solve the task will be. Most times, the human intuition is not so accurate—consider the diverse difficulty level of Hilbert's list of open problems that were proposed at the beginning of the twentieth century [17]. A proposal for using program-size complexity was developed in [5] to measure the relative difficulty between both solved and conjectured mathematical problems. In that original paper and subsequent papers (e.g. see [2,3,4,16]) many concrete measurements (rankings) in difficulty have been done on a wide-range of problems: Goldbach Conjecture, Collatz $3x + 1$ Conjecture, Riemann Hypothesis, Palindrome Conjecture, Fermat's Last Theorem, Dyson's Conjecture and the Four Color Theorem, among others. This method of comparison is based on using a very simple (but Turing-complete) model of computation based on register machine language specifications. This very low-level language allows for, in our opinion, a better comparison between mathematical problems that arise in totally different fields of mathematics such as statements in number theory versus graph theory.

Proof. Arrays of non-negative integers are fundamental to computer science. We can view the integer elements as an encoding of other data types, especially if there is a simple enumeration of the objects. Many combinatorial objects (of a fixed size) can be ranked as an integer and thus the integer elements of the array could easily be a natural way to store sets, permutations, and trees (see [15]). One also sees other data structures as being built upon arrays,

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 81–93, 2012.

such as adjacency matrices or adjacency lists for graphs [14]. Further, if the individual elements of an integer array are restricted to values of 0 and 1 then we essentially have a bit vector. Thus, there is a need to easily process the array data structure in any programming language. In this paper we develop a program-size efficient array library for register machines. This will hopefully then lessen the development time of writing small programs for comparing the difficulty of mathematical problems. As a further application, we note that when register machine subroutines become more complicated (e.g. by having more than two arguments) passing an encapsulated array of arguments as a single register argument should reduce the complexity of both the development process (ensuring correctness) and probably the overall size of the code. □

The rest of this paper is organized as follows. In the next section we briefly specify our method for measuring the level of difficulty of mathematical problems and conjectures by program-size complexity. In Section 3, we give details of a register-machine language that is used to specify programs that can check mathematical problem statements for correctness (never halts) or refute them (halts after finding a counter-example). In Section 4, we give some sample number-theoretic examples, specified as register machine subroutines. In Section 5, we present a new set of register machine subroutines for creating and manipulating arrays of integers stored in a single register. Finally, in the last section, we conclude with some final remarks and open problems.

2 The Method

Consider a prefix-free (self-delimiting) Turing machine M that reads, in a single pass, the program's binary input data x. If M successfully halts on input x, we write $M(x) < \infty$. If M goes in an infinite loop or crashes, we say $M(x) = \infty$. Our method also requires that a program successfully halts only after all of the data is read[1].

We know that there exists a *universal* self-delimiting Turing machine U, such that for every self-delimiting Turing machine M and any input x, there is a string p (depending upon U and M, but not of x) such that $U(px) = M(x)$. The halting probability of U, denoted by Ω_U (and called Chaitin's Omega number, [13,18]) is defined by $\Omega_U = \sum_{U(w)<\infty} 2^{-|w|}$, where $|w|$ denotes the length of $w = px$. With the first n digits of Ω_U, we can solve the *Halting Problem* for all programs w of length at most n.

The basic types of mathematical problems or conjectures that we are initially interested in have the following form: $(\forall N)P(N)$, where P is a computable predicate on N. Fermat's Last Theorem is a simple example: $(\forall (n, a, b, c) \geq (3, 1, 1, 1))(a^n + b^n \neq c^n)$. To make sure we cover all cases we would write our Fermat checking program to enumerate all $n+a+b+c = k$, as k starts at 6 and

[1] This is because we will represent a program p (for M) and its input x as a concatenated string px and require that no valid program can be obtained as a proper extension of another valid program.

increases to ∞. This approach can be extended to other finitely refutable problems with alternating quantifier symbols: $Q_1 n_1 Q_2 n_2 \ldots Q_k n_k P(n_1, n_2, \ldots, n_k)$, where $Q_i \in \{\forall, \exists\}$, $1 \leq i \leq k$, and P is a predicate over variables $n_1, n_2, \ldots, n_k$. For more details see Section 5 of [2].

Consider now a mathematical problem Π, for which we can construct a program M_Π, such that Π is false if and only if $U(M_\Pi) < \infty$ (if such a program exists). Note the input data x in these cases is empty since the program itself will enumerate through the problem domain space to verify or refute the problem Π. We define the "difficulty" of a problem Π by the minimal number of bits of Ω_U necessary to test whether M_Π terminates on U. Note that computing an arbitrary prefix of Ω_U is uncomputable (see [1]) so we do not expect to solve Π with this method. But we can get a feeling (upper bound) for the difficulty of solving Π with respect to other solved/unsolved problems Π' based on their relative program sizes.

3 A Universal Self-delimiting Turing Machine

In this section, we will briefly describe the syntax and the semantics of a register machine program which implements a (natural) universal self-delimiting Turing machine; it is a refinement of the languages described in [13,8,7]. Instead of using the specification of general Turing machines, we use a more practical, but Turing-equivalent, register machine model, which is similar to assembly language for modern digital computers.

Any *register machine* has a finite number of registers, each of which may contain an arbitrarily large non-negative binary integer. The list of instructions is given below in two forms: our compact form and its corresponding Chaitin style version. The main difference between Chaitin's implementation and the one in [5,7] is in the encoding: we use 4 bits instead of 8 bits per character. We note that a more recent variation of the machine uses variable length encodings of both register names and integer constants (see [3,4]).

By default, all registers, labeled with a string characters (restricted to 'a' to 'h' for 4-bit model), are initialized to 0. It is a syntax error if the first occurrence of register j appears before register i in a program, where j is lexicographic greater than i. Also, all registers that are lexicographically less than j must have occurred. Instructions are labeled by default with $0, 1, 2, \ldots$ (in binary).

The register machine instructions are listed below. Note that in all cases R2 denotes either a register or a binary constant (non-negative integer) of the form $1(0+1)^* + 0$, while R1 and R3 must be register variables. We also note that some models (e.g. a different universal register machine such as given in [3]) allow for the third operand R3 to also be a non-negative integer constant. This relaxation makes the programmer's task a little easier since registers do not need to be allocated with target line numbers for both subroutine calls and branching statements. For this paper, with subroutines being relatively small, we use the originally restricted universal machine model.

=R1,R2,R3 **(EQ R1 R2 R3)**

If the contents of R1 and R2 are equal, then the execution continues at the R3-th instruction, where R3 = 0 denotes the first instruction of the program. If they are not equal, then execution continues with the next instruction in sequence. If the content of R3 is outside the scope of the program, then we have an illegal branch error.

&R1,R2 **(SET R1 R2)**

The contents of register R1 is replaced by the contents of register R2.

+R1,R2 **(ADD R1 R2)**

The contents of register R1 is replaced by the sum of the contents of registers R1 and R2.

!R1 **(READ R1)**

One bit is read into the register R1, so the numerical value of R1 becomes either 0 or 1. Any attempt to read past the last data-bit results in a run-time error.

% **(HALT)**

This is the last instruction for each register machine program before the raw data. The program halts the execution in two possible states: either successfully halts or it halts with an under-read error.

A *register machine program* consists of a finite list of labeled instructions from the above list, with the restriction that the `HALT` instruction appears only once, as the last instruction of the list. The input data (a binary string) follows immediately after the `HALT` instruction. A program that does not read the whole input data or attempting to read past the last data-bit results in a under-read run-time error. Some programs (as the ones presented in this paper) have no input data.

To aid the presentation and development of the programs we use a consistent style for subroutines. We use the following conventions:

1. The letter 'L' followed by characters (usually $1, \ldots, 9$) and terminated by ':' is used to mark line numbers. These are local within the subroutine. References to them are replaced with the binary constant in the final program.
2. For unary subroutines, registers a = argument, b = return line, c = answer (a and b are unchanged on return).
3. For binary subroutines, registers a = argument1, b = argument2, c = return line, d = answer (a, b and c are unchanged on return).
4. For subroutines, registers r_0,r_1,... are used for temporary values and probably need to be initialized (for correctness). These are replaced by non-conflicting global registers $e, f, \ldots$ in the final programs (a 4-bit model using 'a' to 'h' characters and an 8-bit or variable compressed models allowing 'a' to 'z' in character strings).
5. For Boolean data types we use integers 0 = `false` and 1 = `true`.

4 Sample Register Machine Subroutines

For completeness and as a starter set of examples, we now include some fundamental number-theoretic algorithms from [5] that will be used in the next section. We have included a smaller-sized version of the subroutine MUL, which yields a slightly smaller complexity for those problems that rely on it. This highlights the fact that we are more interested in program size and not running time efficiency, since the new subroutine is slightly slower when register a has the value of zero.

```
  // Cmp(a,b) returns 0,1,2
      &d,0 // 0 if a=b
      =a,b,c
      +d,1 // 1 if a<b
      &r0,a
      &r1,b
      &r2,L1
      &r3,L2
  L1: +r0,1
      +r1,1
      =r0,b,c
      =r1,a,r3
      =a,a,r2
  L2: +d,1 // 2 if a>b
      =a,a,c
```

```
  // Sub(a,b) returns a-b
      &d,0
      =a,b,c // assumes a>=b
      &r0,b
      &r1,L1
  L1: +d,1
      +r0,1
      =a,r0,c
      =a,a,r1
```

```
// Mul(a,b) returns a*b
[original version]
    &d,0
    =a,0,c
    =b,0,c
    &r0,L1
    &r1,1
    +d,a
L1: =r1,b,c  // check if done
    +r1,1
    +d,a
    =a,a,r0
```

```
[reduced version]
    &d,0
    &r0,L1
    &r1,0
L1: =r1,b,c
    +r1,1
    +d,a // add a=0 is okay
    =a,a,r0
```

```
// Div(a,b) returns integer floor of a/b, assumes b>0
    &r0,L1
    &r1,L2
    &r2,a      // copy of a (variable not used in Cmp & Mul)
    &r3,b      // copy of b
    &r4,c      // copy of c
    &r5,1      // initial answer
```

```
    &c,L0
    &d,Cmp
    =a,a,d      // call Cmp(a,b)
L0: &c,r2
    =d,0,r0
    =d,2,r1
    &d,0        // else a < b so return 0
    =a,a,c
L1: &d,1        // a=b so return 1
    =a,a,c
L2: &c,L5
    &r6,Mul
    &a,r5
    =a,a,r6
L5: &a,d        // just computed ad*b
    &b,r2
    &c,L6
    &r6,Cmp
    =a,a,r6     // call Cmp(ad*b,a)
L6: &b,r3       // reset b
    &r6,L3
    =d,1,r6
    &r6,L4
    =a,a,r6
L3: +r5,1       // still <
    =a,a,r1
L4: &d,r5
    &a,r2       // unpop input parameters
    &b,r3
    &c,r4
    =a,a,c

// Mod(a,b) returns a mod b, assumes b>0
    &r0,a       // copy of parameters
    &r1,b
    &r2,c
    &c,L0
    &d,Div
    =a,a,d      // call Div(a,b)
L0: &a,d
    &c,L1
    &d,Mul
    =a,a,d      // call Mul(a/b,b)
L1: &a,r0
    &b,d
```

```
    &c,L2
    &d,Sub
    =a,a,d      // call Sub(a,[a/b]*b)
L2: &b,r1
    &c,r2
    =a,a,c
```

5 Array Data Structure Library

We now propose a standard way to represent arrays (also called lists, sequences and vectors) inside a single register variable. Basically we use bits of value 1 of the binary representation of an integer to denote (leading) separators or markers of the array elements. If there are no 1's (e.g. the register has value 0) then this represents an array of size 0. An integer element a_i within an array a is represented as a sequence of a_i bits of value 0. For example the array $a = [a_0, a_1, a_2] = [4, 0, 1]$ is represented in binary as 10000110 or in decimal as 134. The function depencancies of the new array library and those methods given in the previous section are illustrated below in Figure 1.

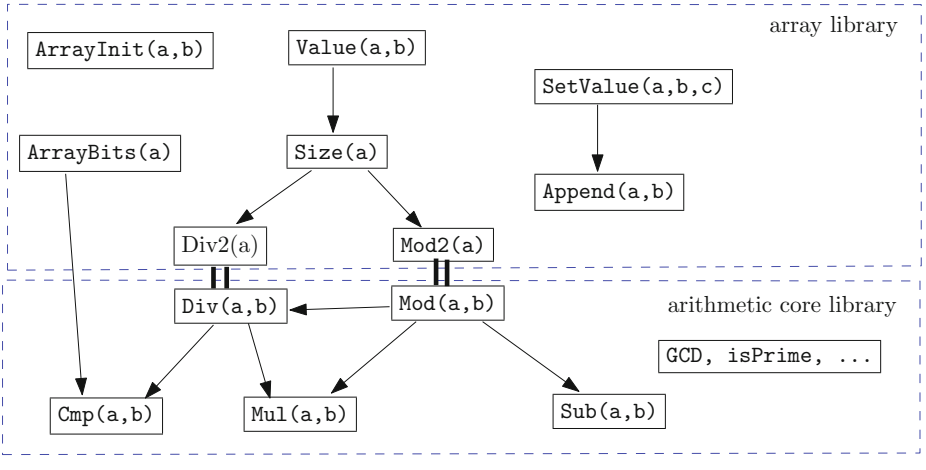


Fig. 1. Depanancies between earlier arithmetic library and the array library

We first start our array library with a utility subroutine that returns the internal bit size of an array (e.g. `ArrayBits([4,0,1])=8`). This can be omitted from a register machine program, if not required.

```
// ArrayBits(a) returns number of bits needed to represent array a
// For 2^k <= a < 2^{k+1}, returns k+1; special case ArrayBits(0)=0
    &r0,b     // save return line number
    &c,L0
    &r1,CMP
    &r2,0     // current k
    &b,1      // current 2^k
    =a,a,r1
L0: =d,1,L1   // check if found a < 2^{k+1}
    +r2,1
    =d,0,L1   // check if found 2^k = a
    +b,b      // double b gives 2^{k+1}
    =a,a,r1
L1: &c,r2
    &b,r0
    =a,a,b
```

Next we give a required core method of determining the capacity size (or length or dimension) of an array in terms of how many elements it contains. This is simply done by counting the number of bits of value 1 in the array register a.

```
// Size(a) returns number of elements of a, when viewed as an array
    &r0,a
    &r1,b
    &r2,0       // number of 1 bits found will equal the array size
    &r3,L3      // exit section
    &r4,L1
    &r5,Mod
    &r6,Div
    &b,2        // we have b=2 for both Mod and Div calls
    &c,L0       // where to return after Mod
    =a,a,r5
L0: =d,0,r4     // another zero so skip tally line
    +r2,1       // else increment bit tally
L1: &c,L2
    =a,a,r6     // shift bits left by dividing by 2
L2: &a,d
    =a,0,r3     // if no more bits exit to L3
    &c,L0
    =a,a,r5     // else repeat by calling Mod
L3: &a,r0
    &b,r1
    &c,r2
    =a,a,b
```

Note that if the subroutine `Mod` is not needed for any other part of the program, one can replace it with a smaller specialized `Mod2` subroutine that just returns the last bit of the binary representation. The idea for the following routine was motivated by Hertel's program-size reduction in [16] that showed that we do not need the full `Mod` subroutine if one just wants to decide if an integer divides another.

```
// Mod2(a) returns a mod 2, or last (low order) bit of a
    &r0,0           // count up to value of a
    &r1,L0
    &r2,L1
L0: &c,0            // even value
    =a,r0,r2
    +r0,1
    &c,1            // odd value
    =a,r0,r2
    +r0,1
    =a,a,r1
L1: =a,a,b
```

A customized and shorter version of `Div2` to "shift bits right" that replaces the need for `Div` is also possible. One way is to have a counter c and have the program stop when $2c = a$ or $2c + 1 = a$. We leave the register machine details of `Div2` to the reader. Note to multiply a register by two or "shift bits left" one just adds a register to itself.

It is easy to append new elements to an array making our representation a somewhat dynamic data structure (think of `ArrayList` in Java or `vector` in C++). Other methods for inserting and removing elements are also possible but the code is omitted here.

```
// Append(a,b) appends element b to end of array a (d not changed)
    &r0,0       // loop counter
    &r1,L1
    &r2,L0
    +a,a        // a=2*a+1 for another element (1 as list separator)
    +a,1
L0: =r0,b,r1
    +a,a        // shift 0 to end until we represent b in unary
    +r0,1
    =a,a,r2
L1: =a,a,c      // return new array
```

For convenience we need a way to create an array of a fixed size. The constructor for creating an array of b elements, with all elements initialized to zero, is as follows.

```
// ArrayInit(a,b) initializes register a to an array of size b
    &a,0       // clear array
    &r0,0      // loop counter
    &r1,L1
    &r2,L0
L0: =r0,b,r1
    +a,a       // shift 0 to end until we represent b in unary
    +a,1       // change last 0 to 1
    +r0,1
    =a,a,r2
L1: =a,a,c     // return new array
```

The standard method for obtaining element a_b from an array $a = [a_0, a_1, \ldots, a_{n-1}]$ is presented next. Note, to keep the program size as small as possible, we do not check that the index b is within the array's index range, which is at most $n - 1 = \texttt{Size}(a) - 1$.

```
// Value(a,b) returns d=a[b]; assumes b<Size(a)
    &r0,a      // copy parameters and label registers
    &r1,b
    &r2,c
    &r3,Size
    &r4,Sub
    &r5,Mod
    &r6,Div
    &b,L0
    =a,a,r3    // compute Size(a)
L0: &a,c
    &b,r1
    +b,1
    &c,L1
    =a,a,r4    // index from left is offset = Size(a)-(b+1)
L1: &r7,d
    &r8,0      // counter of number of times a bit of value 1 seen
    &r3,L2     // recycle r3 since finished with Size
    &r9,L5     // goto L5 when we find the element index
    &a,r0
    &b,2       // we now use b=2 for both Mod and Div calls
L2: =r7,r8,r9
    &c,L3
    =a,a,r5    // get last bit using Mod
L3: +r8,d      // add remainder to our 1 bit count
    &c=L4
    =a,a,r6
L4: &a,d       // divide a by two and repeat to L2
    =a,a,r3
```

```
L5: &r8,0       // now count the number of zeros until preceding 1
    &r3,L6
    &r9,L9
L6: &c,L7
    =a,a,r5     // get last bit using Mod
L7: =d,1,r9     // check end of item
    +r8,1       // else increment element value
    &c=L8
    =a,a,r6
L8: &a,d        // divide a by two and repeat to L6
    =a,a,r3
L9: &a,r0       // restore arguments and return answer
    &b,r1
    &c,r2
    &d,r8
    =a,a,c
```

Finally to change the value of elements of an array we need a method to update the array. This is easily done by creating a new array with the prefix elements, new element and suffix elements concatenated together (using our `Append` subroutine). The template structure for this register machine subroutine `SetValue` is given next, where we use register `d` as the return line (no explicit value is returned).

```
// SetValue(a,b,c) alters element a[b]=c; assumes b<Size(a)
    &r0,a   // copy parameters
    &r1,b
    &r2,c
    &r3,0   // create new array r3
    &r4,Append
```

```
Append a[0..b-1] to r3 by using Value(a,i) calls
```

```
L0: &a,r3   // now Append parameter c to r3
    &b,r2
    &c,L1
    =a,a,r4
L1: &r3,a
```

```
Append a[b+1..Size(a)-1] to r3 by using Value(a,i) calls
```

```
L2: &a,r3   // assign new array r3 to a and return
    &b,r1
    &c,r2
    =a,a,d
```

6 Final Comments

We have promoted a complexity model, based on minimum program sizes for a simple universal register machine, that is easily used and objectively compares the difficulty of mathematical statements and conjectures. In this paper we have extended our earlier register machine's core library of concise subroutines to make it easier to deal with problems that are best implemented (both in terms of bit length and human understanding of correctness) using the array data structure.

We have mentioned in this paper that there are several variations for a universal machine to be used as the base for a complexity measure for problems. Even for our proposed self-delimiting register machines there are seemingly small variations such as allowing the argument R3 to be a constant or having the machine representation of variable length for each programming language token. We believe the relative complexities between mathematical problems are preserved by small changes to the languages. In fact, if we just count the number of instructions used per program, the complexity rankings are most likely to be the same. So the actual machine-level encoding lengths probably do not need to be exactly computed for a complexity comparison. However, if we used a different model with more powerful primitive instructions this may not be the case (e.g. think of the situation where a built-in `IsPrime` function is part of the language).

We end with some open problems and ideas for future work.

- ⋆ Programming using the register machine language is not the most enjoyable way of expressing mathematical problems. It would be nice to have an optimizing compiler that could produce machine-level register machines from the higher-level human-level specifications. It should support different encodings and allow the users to compare their programs' complexity (bit-length sizes) under different models.
- ⋆ For the refined or compressed universal register machine U proposed in [3], can we predict how many initial bits of Ω_U (if any) that can be computed? Compare this value with the 40 bits that were computed for the first-generation register machine of [8,7]. These first bit counts may indicate how far away we are from actually solving mathematical conjectures such as the Riemann's Hypothesis. There are some other interesting questions such as suppose we can only compute 20 bits for Ω_U then one should expect that the same Riemann's Hypothesis program of [5] should encode to about half the length when using U. If not, what are the reasons? Conversely, the encoding length of this program might indicate how many bits we can anticipate for Ω_U.
- ⋆ In addition to having an explicitly specified library for dealing with arrays in register machine language, there are other flavors and simple data structures like dictionaries and graphs that would be nice to have at our disposal. Focusing on arrays, it would be nice to see if a more efficient representation can be used instead of the unary representation of integers that was used here. Note we are still interested in small program-sizes of the subroutines so it is not obviously clear, if reducing the data representation helps. However,

suppose we know each element needs at most k bits (e.g. think of current computers with 32-bit integer primitive data types), can we exploit this property for a smaller array API (application programmers interface)?

⋆ As a possible "home work" exercise, it would be nice to know what is the smallest register machine program (or subroutine) that can sort an array of integers using the proposed array representation and API.

References

1. Calude, C.S.: Information and Randomness: An Algorithmic Perspective, 2nd edn. Revised and Extended. Springer, Berlin (2002)
2. Calude, C.S., Calude, E.: Evaluating the Complexity of Mathematical Problems. Part 1. Complex Systems 18, 267–285 (2009)
3. Calude, C.S., Calude, E.: Evaluating the Complexity of Mathematical Problems. Part 2. Complex Systems 18, 387–401 (2010)
4. Calude, C.S., Calude, E.: The Complexity of the Four Colour Theorem. LMS J. Comput. Math. 13, 414–425 (2010)
5. Calude, C.S., Calude, E., Dinneen, M.J.: A new measure of the difficulty of problems. Journal of Multiple-Valued Logic and Soft Computing 12, 285–307 (2006)
6. Calude, C.S., Calude, E., Svozil, K.: The complexity of proving chaoticity and the Church-Turing Thesis. Chaos 20, 1–5 (2010)
7. Calude, C.S., Dinneen, M.J.: Exact approximations of Omega numbers. Intl. J. of Bifurcation and Chaos 17(6), 1937–1954 (2007)
8. Calude, C.S., Dinneen, M.J., Shu, C.-K.: Computing a glimpse of randomness. Experimental Mathematics 11(2), 369–378 (2002)
9. Calude, C.S., Jürgensen, H., Legg, S.: Solving Finitely Refutable Mathematical Problems. In: Calude, C.S., Păun, G. (eds.) Finite Versus Infinite. Contributions to an Eternal Dilemma, pp. 39–52. Springer, London (2000)
10. Calude, E.: The complexity of Riemann's Hypothesis. Journal for Multiple-Valued Logic and Soft Computing (December 2010) (accepted)
11. Calude, E.: Fermat's Last Theorem and chaoticity. Natural Computing (accepted, June 2011)
12. Calude, E.: The complexity of Goldbach's Conjecture and Riemann's Hypothesis, Report CDMTCS-370, Centre for Discrete Mathematics and Computer Science, p. 9 (2009)
13. Chaitin, G.J.: Algorithmic Information Theory. Cambridge University Press, Cambridge (1987) (third printing, 1990)
14. Dinneen, M.J., Gimel'farb, G., Wilson, M.C.: Introduction to Algorithms, Data Structures and Formal Languages, 2nd edn., p. 264. Pearson Education, New Zealand (2009)
15. Knuth, D.E.: The Art of Computer Programming, 1st edn. Combinatorial Algorithms, Part 1, vol. 4A, p. xv+883. Addison-Wesley, Reading (2011) ISBN 0-201-03804-8
16. Hertel, J.: On the Difficulty of Goldbach and Dyson Conjectures, Report CDMTCS-367, Centre for Discrete Mathematics and Theoretical Computer Science, p. 15 (2009)
17. Hilbert, D.: Mathematical Problems. Bull. Amer. Math. Soc. 8, 437–479 (1901-1902)
18. Weisstein, E.W.: Chaitin's Constant. From MathWorld–A Wolfram Web Resource (2011), http://mathworld.wolfram.com/ChaitinsConstant.html

On Degrees of Randomness and Genetic Randomness

Monica Dumitrescu

Faculty of Mathematics and Computer Science
Department of Mathematics, University of Bucharest
Bucharest, Romania
mdumi@fmi.unibuc.ro

Abstract. This essay is inspired by Cristian Calude's view on degrees of randomness, in relation with "algorithmic randomness". As a "probability person", I am interested in "probabilistic randomness", which can be considered, within the omnipresent uncertainty, only in relation with a real phenomenon/source. Both approaches would produce a characterization of "randomness", as well as a hierarchy of randomness sources. The degree of adequacy for probabilistic randomness can only be evaluated by statistical procedures and it will serve for reliable predictions—which represent the goal of the science "stochastics", as stated by Jakob Bernoulli in the beginning of the18th century.

Quantum randomness, produced by a natural source, can only be evaluated in a relative way, when compared with randomness produced by non-quantum sources. Genetic randomness represents the probabilistic randomness of the actual, observable source of genetic information, DNA. A degree of adequacy should be considered in this case, as expressing the degree the probabilistic model observes the variability and allows reproducibility of the real phenomenon. Such a degree of adequacy can be evaluated by statistical procedures.

1 Introduction

Randomness exists everywhere in our real environment. It is a feature which makes the real phenomena truly interesting, allows diversity and evolution. According to Calude (2000), randomness is one of the most powerful driving forces of life.

Modelling real phenomena goes back in ancient history. For millennia, deterministic models were constructed to explain and interpret the surrounding world and its facts.

It was not until the beginning of the 18th century when people started to notice surrounding randomness. About the year 1700, Jakob Bernoulli drew up the plan to inaugurate a new branch of science, which he called in Latin "Ars conjectandi", or, in Greek, "stochastike". According to von Collani (2000), Bernoulli had identified omnipresent "uncertainty" as a real-world aspect, succeeded in showing that "uncertainty" could be quantified, and was convinced

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 94–104, 2012.

that the new "science of prediction" would change the world for the better. Unfortunately, Jakob Bernoulli passed away in 1705 and his incomplete masterpiece was eventually published only in 1713, by his nephew Nikolaus Bernoulli (see also Bernoulli (1899)).

The adjective "random" is generally used to underline the fact that the corresponding phenomenon has no specific pattern, purpose, or objective. Modelling a random circumstance or event is relating it to a probability distribution. It was Kolmogorov's axiomatic (1933) which allowed the flourishing of "probabilistic randomness".

In parallel with development of computers, a new approach was considered, the "algorithmic randomness". The term covers Martin-Löf, Chaitin, Schnorr, Solovay and Hertling-Weihrauch randomness, among others. According to Chaitin (1975), a series of numbers is "random" if the smallest algorithm capable of specifying it to a computer has about the same number of bits of information as the series itself. This definition was independently proposed around 1965 by A. N. Kolmogorov in USSR and by G. Chaitin in USA.

At this stage, the issue of "degrees of randomness" appears natural. According to Calude (2004), "degrees of algorithmic randomness" can be identified in the sense of algorithmic information theory, Calude (2002). Quantum experimental processes produce measurements which lead to randomness. The irreducible indeterminacy of individual quantum processes postulated by Born (1926) could be interpreted to allow for the production of "random" finite strings, hence "quantum randomness".

A similar issue can be explored for probabilistic randomness in relation to a real–life phenomenon, hence the "degree of adequacy" for probabilistic randomness should be considered and evaluated. One could start from a genuine, observable source of randomness, the source of genetic information–DNA. What would be the degree of adequacy for "genetic randomness"?

2 Algorithmic Randomness and Quantum Randomness

Questions about degrees of algorithmic randomness are studied in algorithmic information theory. Four types could be identified: (i) standard pseudo-randomness produced by software such as MATHEMATICA or MAPLE, which are not only Turing computable but cyclic; (ii) pseudo-randomness produced by software which is Turing computable but not cyclic (e.g. the digits of π, the ratio between the circumference and the diameter of an ideal circle, or Champernowne's constant); (iii) Turing incomputable, but not algorithmically random; and (iv) algorithmically random. It seems this is a positive answer to the question "Do the degrees of randomness mean anything?"

Here we summarise some ideas and conclusions in Calude et al. (2010) on the position of quantum randomness with respect to the above mentioned degrees of algorithmic randomness. Operationally, in the extreme form, Born's postulate could be interpreted to allow for the production of random finite strings; hence quantum randomness could be of type (iv).

But quantum randomness is postulated and is not at all a mathematical consequence of the standard model of quantum mechanics. The legitimacy of the experimental approach comes from characterisations of random sequences in terms of the degrees of incompressibility of their finite prefixes. There is no a priori reason to interpret Born's indeterminism by its strongest formal expression (i.e. in terms of algorithmic randomness). Several tests based on algorithmic information theory analysing algorithmic randomness have produced evidence—with different degrees of statistical significance—of differences between quantum and non-quantum sources.

But one would not be able to "prove absolute randomness"! Any claim of randomness can only be secured relative to, and with respect to, a more or less large class of laws or behaviours, as it is impossible to inspect the hypothesis against an infinity of—and even less so for all—conceivable laws.

3 Probabilistic Randomness and Genetic Randomness

One can discuss about probabilistic randomness only when a probability field $(\Omega, \mathcal{K}, P)$ is accepted as an "environment" for every mathematical item (in this notation, $\mathcal{K}$ is the $\sigma-$field of all events). Thus, one can consider sets, functions, families of functions as follows:

- every subset of Ω is a measurable set, $M \in \mathcal{K}$, called "random event" and it has an associated number, $P(M)$, called the probability of M;
- every function X defined on Ω is a measurable function called "random variable" or "random vector" and it has an associated probability distribution $P \circ X^{-1}$;
- every family of functions depending on a "time parameter" t, $\{X_t, t \in T\}$ is a family of random variables called a "stochastic process" to which one associates a family of finite dimensional probability distributions $\{P \circ (X_{t_1}, ..., X_{t_n})^{-1},\ t_1 < ... < t_n,\ n \in N_+\}$.

So, a random item would be a mathematical item defined on a probability field. Typically, random items should be models for real facts and phenomena which can have different outcomes each time they occur.

In this approach, one cannot expect for any degrees of probabilistic randomness! But one should be able to say "the model $(\Omega, \mathcal{K}, P)$ is more adequate than $\left(\Omega, \mathcal{K}, \widetilde{P}\right)$" and a degree of random–adequacy could be taken into consideration!

Since the probability field $(\Omega, \mathcal{K}, P)$ is the central core of probabilistic randomness, it should be directly connected to facts of the real life. This hope is rarely fulfilled! One good example is Survey Sampling, where $n-$dimensional samples are extracted from a finite, $N-$dimensional population by means of some random procedure. In the case of sampling without replacement from the finite population $\Pi = \{\omega_1, \omega_2, ..., \omega_N\}$, the sample space is

$$\Omega = \left\{ s = (\omega_{i_1}, ..., \omega_{i_n}) \mid \omega_{i_j} \in \Pi \ \forall j,\ \omega_{i_j} \neq \omega_{i_t} \ \forall j \neq t,\ j, t = 1, ..., n \right\},$$
$$\mathcal{K} = \mathcal{P}(\Omega), \quad P : \mathcal{K} \to [0,1], \text{ with } P(\{s\}) = 1 \Big/ \binom{N}{n} \text{ for all } s \in \Omega.$$

Probability theory aims at developing an abstract structure of definitions and theorems based on a system of axioms (such as Kolmogorov's axiom system). The elements and problems dealt with should be related with something real, but this relation is not always obvious.

By contrast, according to von Collani (2000), Bernoulli's "stochastics" aims at investigating and quantifying uncertainty, i.e. variability, in order to make reliable and accurate predictions in real world situations. A model is "adequately random" with respect to a real phenomenon if it assures the reproducibility of that phenomenon and it respects its variability.

Genetic randomness can be a good candidate for adequate probabilistic randomness when we discuss about genetic information and its correct transmission by means of the genetic code.

Let us discuss some models which express real facts from communication, the *information sources with discrete time* $T = Z$ *and finite alphabet* A, denoted $\left(A^T, \mathcal{F}_A, \mu\right)$, where A^T is the set of all double-infinite sequences with letters from A, $\mathcal{F}_A$ is the $\sigma-$field generated by finite dimensional cylinders and μ is a probability measure on $\mathcal{F}_A$. By A^n we denote the set of all strings (words) of length n, and the set of all strings over the alphabet A is denoted A^*. The terms "string" or "word, and "sequence" are quite natural for communication. The main issue is the construction of μ, which would allow us to characterize the information source and see to what degree it fits a real communication situation. A brief notation for $\left(A^T, \mathcal{F}_A, \mu\right)$ is (A, μ), as the broadcast of the source is modelled by the probability measure μ.

An information source (A, μ) is called *stationary* if the distribution μ is shift invariant; that is, the distribution of $(X_{t_1+s}, ..., X_{t_n+s})$ is independent of s for any positive integer n and $t_1, ..., t_n \in T$.

- A stationary information source (A, μ) is called a *source with independent signals* if for every $t \in T$ the following condition is satisfied

$$\Pr(X_t = x_t \mid X_u, u \leq t-1) = \Pr(X_t = x_t),$$

  for every $x_t \in A$, where $\Pr(X_t = x_t \mid Y)$ denotes the conditional probability of $\{X_t = x_t\}$ given Y.
- A source with independent signals is called a *Bernoulli source* if there exists a probability distribution on A, $\{p(x) \geq 0,\ x \in A,\ \sum_{x \in A} p(x) = 1\}$ such that for every n and every $n-$dimensional string we have

$$\mu(x_1, ..., x_n) = \prod_{i=1}^{n} p(x_i).$$

  The brief notation for a Bernoulli source is $(A, p(x))$.

In algorithmic information theory, *Bernoulli sources with uniform distribution* $p(x) = 1/ \mid A \mid$, for all $x \in A$ are of the highest interest, as they produce independent, equally probable signals from A and, for every n and every $n-$dimensional string, we have $\mu(x_1, ..., x_n) = (1/ \mid A \mid)^n$.

– A stationary information source (A, μ) is called a *Markov source* if it satisfies the condition

$$\Pr(X_t = x_t \mid X_u, u \leq t-1) = \Pr(X_t = x_t \mid X_{t-1}),$$

for every $x_t \in A$.

For a Markov source there exists a probability distribution on A,

$$\left\{ p(x) \geq 0,\ x \in A,\ \sum_{x \in A} p(x) = 1 \right\},$$

and a stochastic matrix

$$\left\{ \parallel p(x' \mid x) \parallel_{x, x' \in A},\ p(x' \mid x) \geq 0,\ \sum_{x' \in A} p(x' \mid x) = 1\ \forall x \in A \right\}$$

such that, for every $n > 1$ and every $n-$dimensional string we have

$$\mu(x_1, ..., x_n) = p(x_1) \prod_{i=2}^{n} p(x_i \mid x_{i-1}).$$

The brief notation for a Markov source is $(A, p(x), p(x' \mid x))$.

– A stationary information source (A, μ) is called a *Markov source of order* r if it satisfies the condition

$$\Pr(X_t = x_t \mid X_u, u \leq t-1) = \Pr(X_t = x_t \mid X_u,\ t-r \leq u \leq t-1),$$

for every $x_t \in A$.

For a Markov source of order r there exist a probability distribution on A^r,

$$\left\{ p(x_1, ..., x_r) \geq 0,\ x_1, ..., x_r \in A,\ \sum_{x_1, ..., x_r \in A} p(x_1, ..., x_r) = 1 \right\}$$

and a transition probability matrix

$$\left\{ \begin{array}{c} \| p(x_{r+1} \mid x_1, ..., x_r) \|_{x_1, ... x_r, x_{r+1} \in A},\ p(x_{r+1} \mid x_1, ..., x_r) \geq 0, \\ \sum_{x_{r+1} \in A} p(x_{r+1} \mid x_1, ..., x_r) = 1\ \forall x_1, ..., x_r \in A \end{array} \right\}$$

such that for every $n > r$ and every $n-$dimensional string we have

$$\mu(x_1, ..., x_n) = p(x_1, ..., x_r) \prod_{i=r+1}^{n} p(x_i \mid x_{i-1}, ..., x_{i-r}).$$

The brief notation for a Markov source of order r is

$$(A, p(x_1, ..., x_r), p(x' \mid (x_1, ..., x_r))).$$

One can consider non-Markovian processes, that is processes whose transition probabilities depend on the whole past history. These processes can be found in the literature under different appellations, Maillard (2007). They were first introduced by Onicescu and Mihoc (1935) under the name *chains with complete connections* to study urn models. Chains with complete connections are induced by conditional probabilities of the form $\Pr(X_t = x_t \mid X_u, u \leq t-1)$. These transition probabilities appear to be an extension of the notion of Markov chain of order r with an infinite r. These objects must be taken with some precautions because, in the non-Markovian case, the conditioning is always on an event of probability zero.

– A stationary information source (A, μ) is called a *source with complete connections* consistent with the system of transition probabilities $\{P_n\}_{n \in T}$ if for all n and all $B(X_u, u \leq n)-$measurable functions h,

$$\int_{A^T} h(..., x_{n-1}, x_n)\, d\mu(x) = \int_{A^T} \sum_{y_n \in A} h(..., x_{n-1}, y_n)\, P_n(y_n \mid ..., x_{n-1})\, d\mu(x).$$

The next natural issue is be the quantification of "uncertainty" involved by an information source $\left(A^T, \mathcal{F}_A, \mu\right)$. There are numerous "entropy" measures in circulation but, perhaps, the best known and most used measure of uncertainty is Shannon's entropy (1948).

For an $n-$ dimensional outcome $(X_1, ..., X_n)$, Shannon's entropy is defined by the relation

$$H(X_1, ..., X_n) = -\sum_{x_1, ..., x_n \in A} \mu(x_1, ..., x_n) \log_2 \mu(x_1, ..., x_n).$$

In most cases, the entropy $H(X_1, ..., X_n)$ diverges to infinity as $n \to \infty$. In this sense, the source has infinitely large entropy. This fact suggests that a quantity which plays an important role is, not the limit of $H(X_1, ..., X_n)$, but the rate of growth of entropy. Thus, the entropy of the source is defined by

$$\overline{H} = \lim_{n \to \infty} \frac{1}{n} H(X_1, ..., X_n),$$

when the limit exists.

The following result is well known: *If the discrete time information source* $\left(A^T, \mathcal{F}_A, \mu\right)$ *is stationary, then the entropy of the source exists and it is equal to* $\inf_n (H(X_1, ..., X_n)/n)$.

Notice that, from a physicists standpoint, entropy equals information (but not complexity) and Shannon entropy is not a measure of information, but of information rate.

The entropy of a Bernoulli source $(A, p(x))$ is

$$\overline{H}_{(B)} = -\sum_{x \in A} p(x) \log_2 p(x),$$

the entropy of a Markov source $(A, p(x), p(x' \mid x))$ is

$$\overline{H}_{(M;1)} = -\sum_{x \in A} p(x) \sum_{x' \in A} p(x' \mid x) \log_2 p(x' \mid x),$$

and the entropy of a Markov source of order r is

$$\overline{H}_{(M;r)} = -\sum_{x_1,...,x_r \in A} p(x_1, ..., x_r) \sum_{x' \in A} p(x' \mid x_1, ..., x_r) \log_2 p(x' \mid x_1, ..., x_r).$$

Another candidate for quantification of uncertainty is *the program-size complexity* induced by a (Turing) machine M,

$$H_M(x) = \min\{|z| \mid M(z) = x\},$$

with the convention that the minimum of the empty set is undefined (see, for example, Calude & Dumitrescu (2002)).

For a Bernoulli source $(A, p(x))$ with binary alphabet $A = \{0, 1\}$, let us consider all strings of length n arranged in order of decreasing probability. For $r \in (1/2, 1)$, let $k(n)$ denote the least integer such that

$$\sum_{i=1}^{k(n)} \Pr(x_i) > r.$$

It is well known that the most likely strings have a complexity equal to the entropy. That is, for a universal machine U, we have

$$\overline{H}_{(B)} = \lim_{n \to \infty} \frac{1}{n \cdot k(n)} \sum_{i=1}^{k(n)} H_U(x_i).$$

A similar result can be obtained for Markov binary information sources. For a universal machine U, we have

$$\overline{H}_{(M;1)} = \lim_{n \to \infty} \frac{1}{n} \sum_{|x|=n} H_U(x) \cdot \Pr(x).$$

Bernoulli/Markov/Markov of order r information sources are expressions of genuine randomness. At the same time, they exhibit some "ordering" within sequences $x \in A^T$, such that the characteristics of x are well preserved during evolution.

A beautiful information source where genuine randomness is accompanied by genuine ordering is *DNA, the source of genetic information.*

Let us denote by $A = \{a, u, c, g\}$ the alphabet of DNA, consisting of the four nucleotides {adenine $\longleftrightarrow$ thymine, cytosine $\longleftrightarrow$ guanine}. According to Gatlin (1972), the statistical analysis made for the DNA of more than 60 organisms proved that the nucleotides are not independent in any genetic message. Therefore, the information source which models DNA in the DNA-to-protein communication is not a Bernoulli source. The next natural supposition was that the genetic information source is a stationary Markov source. Based on this assumption, the DNA-to-protein communication channel was analysed (see, for example, Guiasu (1977)).

But the genetic code experimentally determined since 1961 "translates" the genetic message to aminoacids by means of codons, which are triplets of consecutive nucleotides. Hence, it seems that a Markov source of order $r \geq 3$ could be of interest for modelling DNA.

The *genetic randomness* can be the randomness corresponding to a Markov information source of order r, where r is chosen such that the correct DNA-to-protein communication is secured. Thus, the degree of adequacy would tell us to what degree the probabilistic model observes the organism's variability and allows reproducibility of the genome.

It has been experimentally established that living organisms tend to counteract the effect of mutations by increasing the redundancy R of the genetic message,

$$R = 1 - \frac{\overline{H}_{(M;r)}}{\max \overline{H}} = 1 - \frac{\overline{H}_{(M;r)}}{\log_2 4}.$$

According to Gatlin (1972), one can consider the divergence from equiprobability

$$D_1 = \log_2 4 - \overline{H}_{(B)},$$

and the divergence from independence

$$D_2 = \overline{H}_{(B)} - \overline{H}_{(M;r)}.$$

Then,

$$R \cdot \log_2 4 = D_1 + D_2.$$

Experimentally it was noticed that D_2 appears as an evolutionary index which separates the vertebrates from all other lower organisms.

If the letters of A are equiprobable then the potential message variety is maximal but, in compensation, the capacity to detect and correct errors is minimal. On the other hand, if the entropy is reduced to the point where the detection of errors is possible, then the transmission of information is reliable but the message variety can be very low.

The degree of adequacy is expressed by just a balance between the two tendencies, or by just a value of the "memory" of the genetic message.

How could we identify the degree of adequacy? It should be expressed in terms of a pair of values: the order r of the associated Markov information source and the $p-value$ given by a statistical test which decides the acceptance of the order r.

Let us consider the hypothesis

$$H_s : \{\text{the order of dependence of the source is } s, \text{ with } s < r\}$$

against the alternative

$$H_r : \{\text{the order of dependence of the source is } r\}$$

If H_s is true, then

$$p\left(x_{r+1} \mid x_1, ..., x_r\right) = p\left(x_{r+1} \mid x_{r-s+1}, ..., x_r\right)$$

and the maximum likelihood estimates of these transition probabilities are

$$\widehat{p}\left(x_{r+1} \mid x_1, ..., x_r\right) = \widehat{p}\left(x_{r+1} \mid x_{r-s+1}, ..., x_r\right) = \frac{n_{x_{r-s+1},...,x_r,x_{r+1}}}{n_{x_{r-s+1},...,x_r\circ}},$$

where $n_{x_{r-s+1},...,x_r,x_{r+1}}$ represents the number of occurrences of the string $(x_{r-s+1}, ..., x_r, x_{r+1})$ in the observed genetic message and

$$n_{x_{r-s+1},...,x_r\circ} = \sum_{x_{r+1}\in A} n_{x_{r-s+1},...,x_r,x_{r+1}}.$$

In a similar way, if H_r is true, then the maximum likelihood estimates are

$$\widehat{p}\left(x_{r+1} \mid x_1, ..., x_r\right) = \frac{n_{x_1,...,x_r,x_{r+1}}}{n_{x_1,...,x_r\circ}}.$$

The likelihood ratio test for the hypothesis H_s against H_r is based on the following statistic (see, for example, Basawa & Prakasa Rao (1980)):

$$S^2_{(s)} = \sum_{x_1,...,x_{r+1}} \frac{\left(n_{x_1,...,x_r,x_{r+1}} - n_{x_1,...,x_r\circ} \cdot n_{x_{r-s+1},...,x_r,x_{r+1}}/n_{x_{r-s+1},...,x_r\circ}\right)^2}{n_{x_1,...,x_r\circ} \cdot n_{x_{r-s+1},...,x_r,x_{r+1}}/n_{x_{r-s+1},...,x_r\circ}}.$$

If H_s is true, then $S^2_{(s)}$ is approximately CHI-square distributed with $\left(m^s\left(m-1\right)\left(m^{r-s}-1\right)\right)$ degrees of freedom. Here m denotes the number of elements in the alphabet A, and $m = 4$ for DNA.

The decision rule for a significance level α is the following:

- if $S^2_{(s)} < h_{1-\alpha\ ;\ m^s(m-1)(m^{r-s}-1)}$, accept H_s;
- if $S^2_{(s)} \geq h_{1-\alpha\ ;\ m^s(m-1)(m^{r-s}-1)}$, reject H_s.

where $h_{1-\alpha\ ;\ m^s(m-1)(m^{r-s}-1)}$ is the $(1-\alpha)-$quantile of the corresponding CHI-square distribution.

Using a statistical software (for example R, or SPLUS, or STATISTICA) for implementing a CHI-square test, the program returns a $p-value$,

$$p-value = P\left(\chi^2_{m^s(m-1)(m^{r-s}-1)} \geq S^2_{(s)}\right).$$

Very small $p-values$ (close to zero) lead to the rejection of the hypothesis H_s, while larger $p-values$ lead to the acceptance of H_s.

Here we present *an algorithm for establishing the dependence order* r (when the maximum order is v), as suggested in Basawa & Prakasa Rao (1980):

- Test the hypothesis $H_{v-1}:\{$the order of dependence of the source is $v-1\}$ against the alternative $H_v:\{$the order of dependence of the source is $v\}$, using the CHI-square test based on $S^2_{(v-1)}$.
- If H_{v-1} is rejected, decide to accept $r=v$ as the true order of dependence **(stop).**
- If H_{v-1} is accepted, test the hypothesis $H_{v-2}:\{$the order of dependence of the source is $v-2\}$ against the alternative $H_{v-1}:\{$the order of dependence of the source is $v-1\}$, using the CHI-square test based on $S^2_{(v-2)}$.
- etc.;
- Test $H_1:\{$the order of dependence of the source is $1\}$ against the alternative $H_2:\{$the order of dependence of the source is $2\}$, using the CHI-square test based on $S^2_{(1)}$.
- If H_1 is rejected, decide to accept 2 as the true order of dependence **(stop);**
- If H_1 is accepted, test the hypothesis $H_0:\{$Bernoulli information source$\}$ against the alternative $H_1:\{$Markov information source$\}$ by means of an appropriate CHI-square test.

Notice that, for the genetic information source, the hypothesis H_0 should be rejected. Genetic randomness should never resemble quantum randomness!

4 Conclusions

Degrees of algorithmic randomness (see Calude, Svozil (2008)) could be expressed in terms of algorithmic information theory. Quantum randomness should have the highest degree, but this claim can only be secured relative to, and with respect to, a more or less large class of laws or behaviours, as it is impossible to inspect the hypothesis against an infinity of laws, cf. Calude, Dinneen, Dumitrescu, Svozil (2010).

Probabilistic randomness can be considered, within the omnipresent uncertainty, only in relation with a real phenomenon. Genetic randomness should represent the probabilistic randomness of the actual source of genetic information, DNA. One cannot say that an information source is "more random" than another, but a degree of adequacy could be considered, as expressing to what degree the probabilistic model observes the variability and allows reproducibility of the real phenomenon. Such a degree of adequacy could be evaluated by statistical procedures.

References

Basawa, I.V., Prakasa Rao, B.L.S.: Statistical inference for stochastic processes. Academic Press, New York (1980)

Bernoulli, J.: Ars Conjectandi, Basileæ (1713)

Bernoulli, J.: Wahrscheinlichkeitsrechnung (Ars Conjectandi), übersetzt von R. Haussner, Ostwalds Klassiker der exakten Wissenschaften 107 u. 108, Leipzig, Akadem. Verlagsanstalt (1899)

Born, M.: Zur Quantenmechanik der Stoßvorgänge. Z. Phys. (1926); 37, 863; 38, 803

Calude, C.S.: Who is afraid of randomness. In: Millennial Symposium Defining the Science of Stochastics, Würzburg University (2000); published by von Collani, E. (ed.) Defining the Science of Stochastics. Sigma Series in Stochastics, vol. 1, pp. 95–116. Heldermann Verlag (2003)

Calude, C.S.: Information and Randomness—An Algorithmic Perspective, 2nd edn. Springer, Berlin (2002)

Calude, C.S., Dinneen, M.J., Dumitrescu, M., Svozil, K.: Experimental evidence of quantum randomness incomputability. Physical Review A 82, 1–8 (2010)

Calude, C.S., Dumitrescu, M.: Entropic measures, Markov information sources and complexity. Appl. Math. Comput. 132, 369–384 (2002)

Calude, C.S., Svozil, K.: Quantum randomness and value indefiniteness. Advanced Science Letters 1, 165–168 (2008)

Chaitin, G.J.: Randomness and mathematical proof. Scientific American 232(5), 47–52 (1975)

Collani, E.: Theoretical stochastics. In: Millennial Symposium Defining the Science of Stochastics. Würzburg University (2000); published by von Collani, E. (ed.) Defining the Science of Stochastics. Sigma Series in Stochastics, vol. 1, pp. 147–174. Heldermann Verlag (2003)

Guiasu, S.: Information Theory and Applications. McGraw-Hill (1977)

Gatlin, L.: Information Theory and the Living System. Columbia University Press, New York (1972)

Kolmogorov, A.: Grundbegriffe der Wahrscheinlichkeitsrechnung, Julius. Springer, Berlin (1933)

Maillard, G.: Introduction to Chains with Complete Connections. Notes de Cours Doctoral (2007), http://www.latp.univ-mrs.fr/~maillard/Greg/Publications_files/main.pdf

Onicescu, O., Mihoc, G.: Sur les chaînes statistiques. C. R. Acad. Sci. Paris 200, 511–512 (1935)

Shannon, C.E.: A mathematical theory of communication. Bell System Tech., J. 27, 379–423, 623–656 (1948)

Hartmanis-Stearns Conjecture on Real Time and Transcendence*

Rūsiņš Freivalds

Institute of Mathematics and Computer Science,
University of Latvia, Raiņa bulvāris 29, Riga, LV-1459, Latvia

Abstract. Hartmanis-Stearns conjecture asserts that any number whose decimal expansion can be computed by a multitape Turing machine is either rational or transcendental. After half a century of active research by computer scientists and mathematicians the problem is still open but much more interesting than in 1965.

1 Transcendental Numbers

The most interesting results in mathematics, computer science and elsewhere are those which expose unexpected relations between seemingly unrelated objects. One of the most famous examples is the Cauchy-Hadamard theorem relating the radius of convergence of a power series to the properties of the complex variable function defined by the power series. The radius of convergence of a power series f centered on a point a is equal to the distance from a to the nearest point where f cannot be defined in a way that makes it holomorphic.

The nearest point means the nearest point in the complex plane, not necessarily on the real line, even if the center and all coefficients are real. For example, the function

$$f(z) = \frac{1}{1+z^2}$$

has no singularities on the real line, since $1 + z^2$ has no real roots. Its Taylor series about 0 is given by

$$\sum_{n=0}^{\infty} (-1)^n z^{2n}.$$

The Cauchy-Hadamard theorem shows that its radius of convergence is 1. In accordance with this, the function $f(z)$ has singularities at i, which are at a distance 1 from 0. This theorem is unexpected because "I have no interest in complex numbers, my power series is in real numbers only". However, there exists a deep relation between convergence of real power series and complex numbers, and Cauchy-Hadamard theorem merely invites us to investigate this relation more closely.

* The research was supported by Agreement with the European Regional Development Fund (ERDF) 2010/0206/2DP/2.1.1.2.0/10/APIA/VIAA/011.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 105–119, 2012.

This survey is devoted to another unexpected relation in mathematics (or/and theoretical computer science), namely to the Hartmanis-Stearns conjecture which was posed in 1965 and whose status still is "open". In the paper [29] (The ACM A.M. Turing Award, 1993) Juris Hartmanis (1928–) and Richard Edwin Stearns (1936–) asked do there exist irrational algebraic numbers which are computable in real time.

More precisely, a real number is said to be computable in time $T(n)$ if there exists a multitape Turing machine which gives the first n-th terms of its binary expansion in (at most) $T(n)$ operations. *Real time* means that $T(n) = n$. All rational numbers clearly share this property. On the other hand, there are some transcendental numbers that can be computed in real time. Of course, Hartmanis-Stearns conjecture can be posed but why it is interesting? First of all, because mathematicians have had and they still have enormous difficulties to prove transcendence of numbers. Had Hartmanis-Stearns conjecture been proved, this would have been a very powerful tool to obtain new transcendence proofs.

A *rational number* is a number of the form $\frac{p}{q}$, where p and q are integers and q is not zero. An *irrational number* is any complex number which is not rational. A *transcendental number* is a number (possibly a complex number) that is not algebraic–that is, it is not a root of a non-constant polynomial equation with rational coefficients.

The name *transcendental* comes from Gottfried Wilhelm von Leibniz (1646–1716) in his 1682 paper where he proved $sinx$ is not an algebraic function of x. Leonhard Euler (1707–1783) was probably the first person to define transcendental numbers in the modern sense.

Joseph Liouville (1809–1882) first proved the existence of transcendental numbers in 1844, and in 1851 gave the first decimal examples such as the Liouville constant

$$\sum_{k=1}^{\infty} 10^{-k!} = 0.11000100000000000000000001000\ldots$$

We call an irrational number α *well-approximable* if for all positive integers N, n, there is a rational number $\frac{p}{q}$ such that

$$\left|\alpha - \frac{p}{q}\right| < \frac{1}{Nq^n}.$$

It is easy to see that the Liouville constant is well-approximable.

Theorem 1. *(Joseph Liouville [34]) No well-approximable number can be algebraic.*

A completely different proof was given three decades later by Georg Ferdinand Ludwig Philipp Cantor (1845–1918). He proved that there are more real numbers than algebraic numbers. According to the intuitionist school in the philosophy of mathematics (originated by Luitzen Egbertus Jan Brouwer, 1881–1966), such a pure existence proof is not valid unless it explicitly provides an algorithm for the construction of the object whose existence is asserted. However, even much

less radical mathematicians felt that Cantor's theorem does not eliminate the need for explicit proofs of transcendence for specific numbers. Charles Hermite (1822 –1901) proved the transcendence of the number e in 1873. In 1882, Ferdinand von Lindemann published a proof that the number π is transcendental. He first showed that e to any nonzero algebraic power is transcendental, and since $e^{i\pi} = -1$ is algebraic $i\pi$ and therefore π must be transcendental. This approach was generalized by Karl Theodor Wilhelm Weierstrass (1815–1897) to the Lindemann–Weierstrass theorem. The transcendence of π allowed the proof of the impossibility of several ancient geometric constructions involving compass and straightedge, including the most famous one, squaring the circle.

In 1900, David Hilbert (1862–1943) posed an influential question about transcendental numbers, Hilbert's seventh problem: If α is an algebraic number, that is not zero or one, and β is an irrational algebraic number, is α^β necessarily transcendental? The affirmative answer was provided in 1934 by the Gelfond–Schneider theorem (Alexander Osipovich Gelfond, 1906–1968, Theodor Schneider, 1911–1988).

This work was extended by Alan Baker (1939–) in 1966 by proving a result on linear forms in any number of logarithms (of algebraic numbers).

Theorem 2. *(Alan Baker [12]) Let $\alpha_1, \alpha_2, \cdots, \alpha_M$ be nonzero algebraic numbers such that the numbers $\log \alpha_1, \log \alpha_2, \cdots, \log \alpha_M$ are linearly independent over rational numbers. Then for any algebraic numbers $\beta_1, \beta_2, \cdots, \beta_M$, not all zero, the number*

$$\beta_0 + \sum_{m=1}^{M} \beta_m \log \alpha_m$$

is transcendental.

He was awarded the Fields Medal in 1970 for this result. Baker's theorem can give the impression that there is no more difficulty to prove transcendence of numbers widely used in mathematics. Unfortunately, we are still very far from such a situation. Even for many numbers constructed from e, π and similar ones, we do not know much. It is known that e^π is transcendental (implied by Gelfond–Schneider theorem), but for the number π^e it is not known whether it is rational. At least one of $\pi \times e$ and $\pi + e$ (and probably both) are transcendental, but transcendence has not been proven for either number. It is not known if e^e, π^π, π^e are transcendental.

However, not only Liouville's result but also his method was important. It was later generalized to a great extent. For Liouville the most important lemma was as follows.

Lemma 1. *Let α be an irrational algebraic number of degree d. Then there exists a positive constant depending only on α, $c = c(\alpha)$, such that for every rational number $\frac{p}{q}$, the inequality*

$$\frac{c}{q^d} \leq \left| \alpha - \frac{p}{q} \right|$$

is satisfied.

This lemma produced the notion of Liouville number. We say that L is a *Liouville number* if there exists an infinite sequence of rational numbers $\frac{p_n}{q_n}$ satisfying

$$\left|\alpha - \frac{p_n}{q_n}\right| < \frac{1}{q_n^n}.$$

Liouville's theorem asserts that all Liouville numbers are transcendental. Many mathematicians including Axel Thue (1863–1922), Carl Ludwig Siegel (1896–1981), Freeman Dyson (1923–) made important improvements to Liouville's theorem. In 1955 Klaus Friedrich Roth (1925–) provided the best possible improvement.

Theorem 3. *(K.F. Roth [36]) Let α be an irrational algebraic number of degree $d \geq 2$ and let $\epsilon > 0$. Then there exists a positive constant $c = c(\alpha, \epsilon)$, such that for all $\frac{p}{q}$,*

$$\left|\alpha - \frac{p}{q}\right| > \frac{c(\alpha, \epsilon)}{q^{2+\epsilon}}.$$

Roth's result is the best possible, because this statement would fail on setting $\epsilon = 0$ (by Dirichlet's theorem on diophantine approximation there are infinitely many solutions in this case). K.F. Roth was awarded Fields Medal for this result in 1958.

Roth's theorem easily implies transcendence of Champernowne's number considered in Section 2.

Roth's theorem continued the research started by Adolf Hurwitz (1859–1919). Hurwitz's theorem asserted that for arbitrary irrational number α there are infinitely many rationals $\frac{m}{n}$ such that

$$\left|\alpha - \frac{m}{n}\right| < \frac{1}{\sqrt{5}\, n^2},$$

and $\sqrt{5}$ cannot be substituted by a smaller number. Hurwitz's theorem is often used to classify irrational numbers according to the rate of the well-approximability. For example, for the number $\xi = (1+\sqrt{5})/2$ (the golden ratio) there exist only finitely many rational numbers $\frac{m}{n}$ such that the formula above holds true. Unfortunately, the rate of the well-approximability has no direct relation to the number's transcendentability.

2 Normal Numbers

Transcendental numbers initially were supposed to be more complicated rather than algebraic numbers. At least, the choice of the term "transcendental" suggests so. On the other hand, the Liouville constant has a rather simple description

$$0.11000100000000000000000001000\ldots$$

while the algebraic number

$$\sqrt{2} = 1.41421356237309504880168872420969807$$
$$856967187537694807317667973799...$$

seems to be quite "random". Of course, all rational numbers are algebraic and decimal expansions of them (all the other b-adic expansions as well, by the way) are periodic and hence simple. It turns out that all irrational algebraic numbers are rather complicated.

In 1909 Emile Borel (Félix Édouard Justin Émile Borel, 1871–1956) asked whether it is possible to tell transcendental numbers from algebraic ones by statistics of their digits in some b-adic expansion. He introduced the notion of a *normal number*.

Let x and $b \geq 2$. Consider the sequence of digits of the expansion of x in base b. We are interested in finding out how often a given digit s shows up in the above representation of x. If we denote by $N(s, n)$ the number of occurrences of s in the first n digits of x, we can calculate the ratio $\frac{N(s,n)}{n}$. As n approaches ∞, this ratio may converge to a limit, called the frequency of s in x. The frequency of s in x is necessarily between 0 and 1. If all base b digits are equally frequent, i.e. if the frequency of each digit s, $0 \leq s < b$, is $\frac{1}{b}$, then we say that x is *simply normal* in base b. For example, in base 5, the number $0123401234012340123401234\cdots$ is simply normal.

If we allow s to be any finite string of digits (in base b), then we have the notion of a normal number. However, we have to be careful as to how to count the number of occurrences of s and what is the meaning of the frequency of s in x. Let x be a real number as stated in the previous section. Let s be a string of digits of length k, in base $b : s = s_1 s_2 \cdots s_k$ where $0 \leq s_j < b$. Define $N(s, n)$ to be the number of times the string s occurs among the first n digits of x in base b. For example, if $x = 21131112$ in base 4, then $N(1, 8) = 5, N(11, 8) = 3$, and $N(111, 8) = 1$. We say that x is normal in base b if

$$\lim_{n\to\infty} \frac{n}{N(s,n)} = \frac{1}{b^k}$$

for every finite string s of length k. We see that if $k = 1$, we are back to the definition of a simply normal number, so every number normal in base b is in particular simply normal in base b.

Intuitively, x is normal in base b if all digits and digit-blocks in the base b digit sequence of x occur just as often as would be expected if the sequence had been produced completely randomly. Unlike simply normal numbers, normal numbers are necessarily irrational. Normal numbers are not as easy to find as simply normal numbers. One example is Champernowne's number

$$0.1234567891011121314\cdots$$

(obtained by concatenating the decimal expansions of all natural numbers), which is normal in base 10 [20]. It is not known whether Champernowne's number is normal in other bases. Champernowne's number can be written as

$$C_{10} = \sum_{n=1}^{\infty} \sum_{k=10^{n-1}}^{10^n-1} \frac{k}{10^{kn-9\sum_{k=0}^{n-1} 10^k(n-k)}}.$$

There exist numbers which are normal in all bases $b = 2, 3, 4, \cdots$. They are called *absolutely normal*. The first absolutely normal number was constructed by Wacław Franciszek Sierpiński (1882–1969) in 1917 [39]. Verónica Becher and Santiago Figueira [14] proved.

Theorem 4. *(Becher, Figueira [14]) There exists a computable absolutely normal number.*

The construction of computable absolutely normal numbers is an innovative and complicated recursive function theoretical adaptation of Sierpiński's construction. (By the way, the authors of [14] acknowledge valuable comments from Cristian Calude.)

Later in [16] Borel asked whether all irrational algebraic numbers are absolutely normal. It is still not known. The mere existence of this open problem shows that absolute normality of numbers is a property that can be possessed only by numbers whose decimal (and other b-adic) expansions are very complicated. Unfortunately, no one has been able to use this observation to tell transcendental numbers from algebraic ones.

In contrast to Borel's conjecture, it is needed to say that all algebraic numbers about whom we know that they are absolutely normal, are highly artificial. They are specially constructed to prove their absolute normality.

However, is the notion of absolutely normal numbers the notion we need to prove the Hartmanis-Stearns conjecture? Existence of computable absolutely normal numbers may be interpreted as a sign that we are to include ideas of inductive inference (see, e.g. [28,27,11]) in the search for a notion suitable to distinguish transcendental numbers from algebraic ones.

3 Continued Fractions

Now we are looking for another way to describe irrational numbers with a hope that this new description could be used to distinguish transcendental numbers. One such potentially useful description is continued fractions.

The continued fraction is a natural notion. Most people believe that there cannot exist ways to memorize good approximations of the number

$\pi = 3.14159265358979323846264338327950288419716939937510582097494459230781640628620899862803482534211706798214808651328230664709384460955058223172535940812848111745 02...$

However, they exist:

$$\pi = 3 + \cfrac{1^2}{6+\cfrac{3^2}{6+\cfrac{5^2}{6+\cfrac{7^2}{6+\cfrac{9^2}{6+\cdots}}}}} = \cfrac{4}{1+\cfrac{1^2}{3+\cfrac{2^2}{5+\cfrac{3^2}{7+\cfrac{4^2}{9+\cdots}}}}} = \cfrac{4}{1+\cfrac{1^2}{2+\cfrac{3^2}{2+\cfrac{5^2}{2+\cfrac{7^2}{2+\cfrac{9^2}{2+\cdots}}}}}}.$$

A *finite continued fraction* is an expression of the form

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cfrac{1}{\cdots + \cfrac{1}{a_n}}}}}$$

where a_0 is an integer, any other a_i members are positive integers, and n is a non-negative integer. An *infinite continued fraction* can be written as

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cfrac{1}{a_4 + \cfrac{1}{\cdots}}}}}$$

One can abbreviate a continued fraction as $x = [a_0; a_1, a_2, a_3, a_4, \cdots]$.

The decimal representation of real numbers has some problems. One problem is that many rational numbers lack finite representations in this system. For example, the number $\frac{1}{3}$ is represented by the infinite sequence $(0, 3, 3, 3, 3,)$. Another problem is that the constant 10 is an essentially arbitrary choice, and one which biases the resulting representation toward numbers that have some relation to the integer 10. Continued fraction notation is a representation of the real numbers that avoids both these problems.

Continued fractions provide regular patterns for many important numbers. For example, the golden ratio

$$\frac{1+\sqrt{5}}{2} = \varphi = 1 + \frac{1}{\varphi}$$

has a continued fraction representation $\varphi = [1; 1, 1, 1, 1, \cdots]$.

Notably,

$e = [2; 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 8, 1, 1, 10, 1, 1, 12, 1, 1, \cdots]$,
$e^2 = [7; 2, 1, 1, 3, 18, 5, 1, 1, 6, 30, 8, 1, 1, 9, 42, 11, \cdots, 3k, 12k + 6, 1, 1, \cdots]$,
$e^{\frac{1}{n}} = [1; n - 1, 1, 1, 3n - 1, 1, 1, 5n - 1, 1, 1, 7n - 1, 1, 1, \cdots]$
$\tan(1) = [1; 1, 1, 3, 1, 5, 1, 7, 1, 9, 11, 1, 13, 1, 15, 1, 17, 1, \cdots]$.

If arbitrary values and/or functions are used in place of one or more of the numerators the resulting expression is a *generalized continued fraction*. The three distinct fractions above for π were generalized continued fractions. Every real number has exactly one standard continued fraction. The continued fraction for π is not as regular as the generalized continued fractions shown above: $\pi = [3; 7, 15, 1, 292, 1, 1, 1, 2, 1, 3, 14, 2, 1, 1, 2, 2, 2, 2, 1, 84, \cdots]$. From results of Leonhard Euler (1707–1783) and Joseph-Louis Lagrange (1736–1813) we know that the regular continued fraction expansion of x is periodic if and only if x is a quadratic irrational.

Continued fractions may give us many still not discovered criteria for properties of numbers. For example, if $a_1, a_2, \cdots$ and $b_1, b_2, \cdots$ are positive integers with $a_k \leq b_k$ for all sufficiently large k, then the generalized continued fraction

$$b_0 + \cfrac{a_1}{b_1 + \cfrac{a_2}{b_2 + \cfrac{a_3}{b_3 + \cdots}}}$$

converges to an irrational limit.

Aleksandr Yakovlevich Khinchin (1894–1959) [32] expressed a conjecture in 1949 which is now widely believed that the continued fraction expansion of any irrational algebraic number α is either eventually periodic (and we know that this is the case if and only if α is a quadratic irrational), or it contains arbitrarily large partial quotients.

J.P. Allouche [9] conjectures that the continued fraction expansion of any algebraic irrational number that is not a quadratic number is normal.

4 Automata in Number Theory

J. Hartmanis and R. Stearns asked do there exist irrational algebraic numbers which are computable in real time. We wish to prove a negative result. Hence it seems natural that we start by proving that it is not possible to compute an irrational algebraic number by a finite automaton. Indeed, a finite automaton with no input information can produce only a periodic sequence but every number whose b-adic expansion is periodic, is inevitably rational. Too simple for a good result.

However, there is a possibility for nontrivial results.

Definition 1. *Let $b \geq 2$ be an integer. A sequence (a_n) is called b-automatic if there exists a finite automaton taking the base-b expansion of n as input and producing the term a_n as output.*

It is not hard to prove that all periodic sequences are b-automatic for every integer $b \geq 2$. But is every 2-automatic sequence also 3-automatic? Alan Cobham (1927-) published two influential papers [21,22] on this topic.

Theorem 5. *(Cobham [21]) A sequence is b-automatic if and only if it is b^r-automatic for all positive integers r.*

Definition 2. *Two positive integers b and d are multiplicatively independent if the equation $b^a = d^b$ has no nontrivial integer solution (a, b), that is, $\frac{\log b}{\log d}$ is irrational.*

Theorem 6. *(Cobham [21]) A nonperiodic sequence cannot be both b-automatic and d-automatic for two multiplicatively independent positive integers b and d.*

The class of automatic sequences remains unchanged if we choose in the definition of automatic sequences that automata should read the input from the right to the left, that is, when starting with the least significant digit [6].

There are several characterizations of automatic sequences. One of the is related to morphisms. Let A and B be two finite sets. A map σ from A to B^* extends uniquely to a homomorphism between the free monoids A^* and B^* (that is, $\sigma(w_1 w_2 \cdots w_r) = \sigma(w_1)\sigma(w_2)\cdots\sigma(w_r)$). Such a homomorphism from A^* to B^* is called a *morphism*.

If there is a positive integer k such that each element of A is mapped to a word of length k, then the morphism is called *k-uniform*. A *coding* is a 1-uniform morphism.

A k-uniform morphism σ from A^* to itself is said to be prolongable if there exists a letter a such that $\sigma(a) = aw$. In that case, the sequence of finite words $(\sigma^n(a))_{n\geq 0}$ converges in $A^\infty = A^* \cup A^{\natural N}$ (endowed with its usual topology) to an infinite word denoted $\sigma^\infty(a)$. This infinite word is a *fixed point* for σ (extended by continuity to infinite words) and we say that $\sigma(a)$ is generated by the morphism σ.

For example, the morphism τ defined over the alphabet$\{0, 1\}$ by $\tau(0) = 01$ and $\tau(1) = 10$ is a 2-uniform morphism which generates the *Thue–Morse sequence*

$$t = \tau^\infty(0) = 0; 1; 10; 1001; 10010110; 1001011$$

$$001101001; 1001011001101001011010011001010010\cdots$$

(where the signs ";" are not members of this sequence, they are inserted only to show the structure).

Theorem 7. *(Cobham [22]) An infinite word is k-automatic if and only if it is the image by a coding of a word that is generated by a k-uniform morphism.*

Definition 3. *The k-kernel of a sequence $a = (a_n)_{n\geq 0}$ is defined as the set*

$$N_k(a) = \{(a_{k^r n+i})_{n\geq 0} \mid i \geq 0, 0 \leq i < k^r\}.$$

For example, the 2-kernel of the Thue–Morse sequence t has only two elements (t and the sequence $\overline{t}$ obtained by exchanging the symbols 0 and 1 in t).

Theorem 8. *(Eilenberg [24]) A sequence is k-automatic if and only if its k-kernel is finite.*

Hint of proof. The k-kernel of a given k-automatic sequence corresponds to the different sequences you can obtain by changing the initial state in such an automaton. Since there are only a finite number of state, the k-kernel has to be finite. □

In [35] John Loxton and Alf van der Poorten proved transcendence results on values of Mahler functions giving strong support for the belief that the decimal expansion (more generally the base b expansion) of an irrational algebraic number cannot be generated by a finite automaton. In consequence the matter of the transcendence of irrational automatic numbers became known as the conjecture of Loxton and van der Poorten.

In 2004 Boris Adamczewski, Yann Bugeaud and Florian Luca [7] applied Schlickeweis p-adic generalization [37] of Wolfgang Schmidt's subspace theorem [38] (which is itself a multidimensional generalization of Roth's theorem [36]) to prove that for any b, the base b expansion of irrational algebraic number

$$\liminf_{n\to\infty} \frac{p(n)}{n} = +\infty$$

where $p(n)$ is the number of distinct subwords of the length n in the b-ary expansion of the given irrational algebraic number. Almost all numbers are such that their $p(n)$ is near to b^n but for the sequences generated by a finite automaton $p(n) = O(n)$. This proves.

Theorem 9. *(Adamczewski, Bugeaud, Luca [7,3]) Let $b \geq 2$ be an integer. The b-ary expansion of any irrational algebraic number cannot be generated by a finite automaton.*

In other words, irrational automatic numbers are transcendental.

For continued fractions the results are somewhat similar. In 1997, Ferenczi and Mauduit [25] proved.

Theorem 10. *(Ferenczi, Mauduit [25]) Assume that the base b representation of α is for each n of the form $0.U_nV_nV_nV'_n\cdots$, where V'_n is a prefix of V_n, and the following length conditions are satisfied:*

$$\mid V_n \mid \to \infty; \liminf_{n\to\infty} \frac{\mid U_n \mid}{\mid V_n \mid} < \infty.$$

Then the number α is transcendental.

Allouche [9] noticed that the methods of [25] give a bit more. First define the *complexity* of a sequence $\{u_n\}$ of digits as the function $k \mapsto p(k)$ that counts the number of distinct blocks of length k appearing in the sequence. A normal number (in base b) certainly has $p(k) = b^k$. Thus, we might expect that any number with $p(k) < b^k$ is transcendental. A step in this direction is provided by

Theorem 11. *(Allouche [9]) Assume that $p(k)$ is for large k large enough dominated by a function of the form $k+a$. Then x is either rational or transcendental.*

Let S_0 be 0 and S_1 be 01. Now $S_n = S_{n-1}S_{n-2}$ (the concatenation of the previous sequence and the one before that).

We have: $S_0 = 0$, $S_1 = 01$, $S_2 = 010$, $S_3 = 01001$, $S_4 = 01001010$, $S_5 = 0100101001001$. The Fibonacci sequence is

$$0100101001001010010100100101001001010010100100101001010010100$$

$$10010100100101001010010010100100101001010010100100101001\cdots$$

Theorem 12. *(Allouche and Zamboni [10]) If the binary expansion of a real number is the fixed point of a morphism that is either primitive (e.g., the Fibonacci sequence) or of fixed length (e.g., the Thue-Morse sequence), then this number is either rational or transcendental.*

Since in (generalized) continued fraction the partial numerators and partial denominators can be large numbers, it is not easy to have a *natural* notion of computation of a continued fraction by a finite automaton. It seems that for a suitable formalization of this notion the counterpart of the Hartmanis-Stearns

conjecture may be true, and moreover, be a part of more more general problem connected with the Hartmanis-Stearns conjecture.

Impressing results are obtained by B. Adamczewski and Y. Bugeaud [4]. Let A be a given set, not necessarily finite. The length of a word W on the alphabet A, that is, the number of letters composing W, is denoted by $\mid W \mid$. For any positive integer k, we write W_k for the word $W \cdots W$ (k times repeated concatenation of the word W). More generally, for any positive rational number x, we denote by W^x the word $W^{\lfloor x \rfloor} W'$, whereW' is the prefix of W of length $\lceil (\lfloor x \rfloor) \mid W \mid \rceil$. Here $\lfloor y \rfloor$ and $\lceil y \rceil$ denote, respectively, the integer part and the upper integer part of the real number y. For example, if W denotes the word 232243, then $W^{3/2}$ is the word 232243232. Let $a = (a_s)_{s \geq 1}$ be a sequence of elements from A, that we identify with the infinite word $a_1 a_2 \cdots a_s \cdots$. Let w be a rational number with $w > 1$. We say that a satisfies Condition $(*)_w$ if a is not eventually periodic and if there exists a sequence of finite words $(V_n)_{n \geq 1}$ such that:

(i) For any $n \geq 1$, the word V_n^w is a prefix of the word a;
(ii) The sequence $(\mid V_n \mid)_{n \geq 1}$ is increasing.

Theorem 13. *(Adamczewski, Bugeaud [4]) Let $a = (a_s)_{s \geq 1}$ be a sequence of positive integers. Let $(p_s/q_s)_{s \geq 1}$ denote the sequence of convergents to the real number $\alpha = [0; a_1, a_2, \cdots, a_s, \cdots]$. If there exists a rational number $w \geq 2$ such that a satisfies Condition $(*)_w$, then α is transcendental. If there exists a rational number $w > 1$ such that a satisfies Condition $(*)_w$, and if the sequence $(q_s^{1/s})_{s \geq 1}$ is bounded (which is in particular the case when the sequence a is bounded), then α is transcendental.*

Let w and w' be nonnegative rational numbers with $w > 1$. We say that a satisfies Condition $(**)_{w,w'}$ if a is not eventually periodic and if there exist two sequences of finite words $(U_n)_{n \geq 1}, (V_n)_{n \geq 1}$ such that:

(i) For any $n \geq 1$, the word $U_n V_n^w$ is a prefix of the word a;
(ii) The sequence $(\mid U_n \mid / \mid V_n \mid)_{n \geq 1}$ is bounded from above by w';
(iii) The sequence $(\mid V_n \mid)_{n \geq 1}$ is increasing.

Theorem 14. *(Adamczewski, Bugeaud [4]) Let $a = (a_s)_{s \geq 1}$ be a sequence of positive integers. Let $(p_s/q_s)_{s \geq 1}$ denote the sequence of convergents to the real number $\alpha = [0; a_1, a_2, \cdots, a_s, \cdots]$. Assume that the sequence $(q_s^{1/s})_{s \geq 1}$ converges. Let w and w' be non-negative real numbers with $w > w' + 1$. If a satisfies Condition $(**)_{w,w'}$, then α is transcendental.*

5 How Close We Are to the Hartmanis-Stearns Conjecture?

Two real numbers α and α' are said to be equivalent if their b-adic expansions have the same tail.

Adamczewski and Bugeaud [5] say that a is a *stammering sequence* if there exist a real number $w > 1$ and two sequences of finite words $(W_n)_{n \geq 1}, (X_n)_{n \geq 1}$ such that:

(i) For any $n \geq 1$, the word $W_n X_n^w$ is a prefix of the word a;
(ii) The sequence $(\mid W_n \mid / \mid X_n \mid)_{n\geq 1}$ is bounded from above;
(iii) The sequence $(\mid X_n \mid)_{n\geq 1}$ is increasing.

Theorem 15. *(Adamczewski, Bugeaud, Luca [7]) Let $a = (a_k)_{k\geq 1}$ be a stammering sequence of integers from $\{0, 1, \cdots, b-1\}$. Then, the real number*

$$\alpha = \sum_{k=1}^{+\infty} \frac{a_k}{b^k}$$

is either rational or transcendental.

Let $a = (a_k)_{k\geq 1}$ and $a' = (a'_k)_{k\geq 1}$ be sequences of elements from A, that we identify with the infinite words $a_1 a_2 \cdots$ and $a'_1 a'_2 \cdots$, respectively. We say that the pair (a, a') satisfies Condition $(*)$ if there exist three sequences of finite words $(U_n)_{n\geq 1}, (U'_n)_{n\geq 1}$, and $(V_n)_{n\geq 1}$ such that:

(i) For any $n \geq 1$, the word $U_n V_n$ is a prefix of the word a;
(ii) For any $n \geq 1$, the word $U'_n V_n$ is a prefix of the word a';
(iii) The sequences $(\mid U_n \mid / \mid V_n \mid)_{n\geq 1}$ and $(\mid U'_n \mid / \mid V_n \mid)_{n\geq 1}$ are bounded from above;
(iv) The sequence $(\mid V_n \mid)_{n\geq 1}$ is increasing.

If, moreover, we add the condition

(v) The sequence $(\mid U_n \mid - \mid U'_n \mid)_{n\geq 1}$ is unbounded, then, we say that the pair (a, a') satisfies Condition $(**)$.

Theorem 16. *(Adamczewski, Bugeaud [5]) Let $a = (a_k)_{k\geq 1}$ and $a' = (a'_k)_{k\geq 1}$ be sequences of integers from $\{0, 1, \cdots, b-1\}$. If the pair (a, a') satisfies Condition $(*)$, then at least one of the real numbers*

$$\alpha = \sum_{k=1}^{+\infty} \frac{a_k}{b^k} \ , \ \alpha' = \sum_{k=1}^{+\infty} \frac{a'_k}{b^k}$$

is transcendental, or α and α' are equivalent. Furthermore, if the pair (a, a') satisfies Condition $(**)$, then at least one of the real numbers α, α' is transcendental, or they are equivalent and both rational.

Theorems 15 and 16 show that the solution to Hartmanis-Stearns conjecture may be coming soon. Indeed, consider the case when a and a' in Theorem 16 is the same. Conditions $(*)$ and $(**)$ are similar (but not quite identical) to the situation when a is generated by a Turing machine working in real time. It is important that the machine prints a symbol of output one symbol per step of computation. The output symbols at moments $[t, t+u]$ depend only on the state of the machine and of what is written on fragments (not longer than u squares) of a constant number of work tapes. Hence if at two distinct moments t and t' this information is the same then the output symbols at moments $[t, t+u]$ and $[t', t'+u]$ also are the same.

Suppose, we have denoted outputs of the machine till the moments t and t' by U_t and $U_{t'}$, respectively, and the output at $[t, t+u]$ by V_t. The "naive" observation of Turing machines does not allow to prove the properties (i)–(v). On the other hand, who knows whether this approach to proof of the Hartmanis-Stearns conjecture is perspective.

References

1. Ablayev, F.M., Freivalds, R.: Why Sometimes Probabilistic Algorithms Can Be More Effective. In: Wiedermann, J., Gruska, J., Rovan, B. (eds.) MFCS 1986. LNCS, vol. 233, pp. 1–14. Springer, Heidelberg (1986)
2. Adamczewski, B., Allouche, J.-P.: Reversals and palindromes in continued fractions. Theoretical Computer Science 380(3), 220–237 (2007)
3. Adamczewski, B., Bugeaud, Y.: On the complexity of algebraic numbers I. Expansions in integer bases. Annals of Mathematics 165, 547–565 (2007)
4. Adamczewski, B., Bugeaud, Y.: On the complexity of algebraic numbers II. Continued fractions. Acta Mathematica 195(1), 1–20 (2005)
5. Adamczewski, B., Bugeaud, Y.: On the independence of expansions of algebraic numbers in an integer base. Bulletin London Mathematical Society 39(2), 283–289 (2007)
6. Adamczewski, B., Bugeaud, Y.: Mesures de transcendance et aspects quantitatifs de la mthode de Thue-Siegel-Roth-Schmidt. Proceedings of the London Mathematical Society 101(1), 1–26 (2010)
7. Adamczewski, B., Bugeaud, Y., Luca, F.: Sur la complexité des nombres algébriques. Comptes Rendus de l'Académie des Sciences, Paris Ser. I 336, 11–14 (2004)
8. Agafonov, V.N.: Normal sequences and finite automata. Soviet Mathematics Doklady 9, 324–325 (1968)
9. Allouche, J.-P.: Nouveaux résultats de transcendance de réels à développement non aléatoire. Gazette des Mathématiciens (84), 19–34 (2000)
10. Allouche, J.-P., Zamboni, L.Q.: Algebraic irrational binary numbers cannot be fixed points of non-trivial constant length or primitive morphisms. Journal of Number Theory 69, 119–124 (1998)
11. Ambainis, A., Apsītis, K., Calude, C., Freivalds, R., Karpinski, M., Larfeldt, T., Sala, I., Smotrovs, J.: Effects of Kolmogorov Complexity Present in Inductive Inference as Well. In: Li, M. (ed.) ALT 1997. LNCS, vol. 1316, pp. 244–259. Springer, Heidelberg (1997)
12. Baker, A.: Linear forms in the logarithms of algebraic numbers I–III. Mathematika. A Journal of Pure and Applied Mathematics 13, 204–216 (1966); 14, 102–107 (1967); 14, 220–228 (1967)
13. van Aardenne-Ehrenfest, T., de Bruijn, N.G.: Circuits and trees in oriented linear graphs. Simon Stevin 28, 203–217 (1951)
14. Becher, V., Figueira, S.: An example of a computable absolutely normal number. Theoretical Computer Science 270(1-2), 947–958 (2002)
15. Borel, É.: Les probabilités dénombrables et leurs applications arithmétiques. Rendiconti del Circolo Matematico di Palermo 27, 247–271 (1909)

16. Borel, É.: Sur les chiffres décimaux de $\sqrt{2}$ et divers problèmes en châine. Comptes Rendus de l'Académie des Sciences, Paris 230, 591–593 (1950)
17. Calude, C.: What is a random string? Journal of Universal Computer Science 1(1), 48–66 (1995)
18. Calude, C.: Borel normality and algorithmic randomness. In: Rozenberg, G., Salomaa, A. (eds.) Developments in Language Theory: At the Crossroads of Mathematics, Computer Science and Biology, pp. 113–119. World Scientific, Singapore (1994)
19. Calude, C.S., Zamfirescu, T.: Most numbers obey no probability laws. Publicationes Mathematicae Debrecen 54(Supplement), 619–623 (1999)
20. Champernowne, D.G.: The construction of decimals normal in the scale of ten. The Journal of the London Mathematical Society 8, 254–260 (1933)
21. Cobham, A.: On the base-dependence of sets of numbers recognizable by finite automata. Mathematical Systems Theory 3, 186–192 (1969)
22. Cobham, A.: Uniform tag sequences. Mathematical Systems Theory 6, 164–192 (1972)
23. Copeland, A.H., Erdös, P.: Note on normal numbers. Bulletin of the American Mathematical Society 52, 857–860 (1946)
24. Eilenberg, S.: Automata, Languages and Machines, vol. A, B. Academic Press, New York (1974)
25. Ferenczi, S., Mauduit, C.: Transcendence of numbers with a low complexity expansion. Journal of Number Theory 67(2), 146–161 (1997)
26. Freivalds, R.: Amount of nonconstructivity in deterministic finite automata. Theoretical Computer Science 411(38-39), 3436–3443 (2010)
27. Freivalds, R., Kinber, E.B., Wiehagen, R.: How inductive inference strategies discover their errors. Information and Computation 118(2), 208–226 (1995)
28. Gold, E.M.: Language identification in the limit. Information and Control 10(5), 447–474 (1967)
29. Hartmanis, J., Stearns, R.E.: On the computational complexity of algorithms. Transactions of the American Mathematical Society 117, 285–306 (1965)
30. Hurwitz, A.: Über die angenäherte Darstellung der Irrationalzahlen durch rationale Brüche. Mathematische Annalen 39(2), 279–284 (1891)
31. Kaneps, J., Freivalds, R.: Minimal Nontrivial Space Complexity of Probabilistic One-Way Turing Machines. In: Rovan, B. (ed.) MFCS 1990. LNCS, vol. 452, pp. 355–361. Springer, Heidelberg (1990)
32. Khinchin, A.Y.: Continued Fractions. Dover Publications (2007) (translation from Russian original, GITTL, 1949)
33. Kolmogorov, A.N., Uspensky, V.A.: Algorithms and randomness. Theory of Probability and Its Applications 32, 389–412 (1987)
34. Liouville, J.: Sur des classes très-étendues de quantités dont la valeur n'est ni algébrique, ni même réductible à des irrationnelles algébriques. Journal de Mathématiques Pures et Appliquées 16(1), 133–142 (1851)
35. Loxton, J.H., van der Poorten, A.J.: Transcendence and algebraic independence by a method of Mahler. In: Baker, A., Masser, D.W. (eds.) Transcendence Theory: Advances and Applications. ch. 15, pp. 211–226. Academic Press, London (1977)
36. Roth, K.F.: Rational approximations to algebraic numbers. Mathematika. A Journal of Pure and Applied Mathematics 2, 1–20, 168 (1955)
37. Schlickewei, H.P.: The 2-adic Thue-Siegel-Roth-Schmidt theorem. Archiv der Mathematik 29(1), 267–270 (1977)

38. Schmidt, W.: The subspace theorem in diophantine approximation. Compositio Mathematica, Tome 69(2), 121–173 (1989)
39. Sierpiński, W.: Démonstration élémentaire d'un théorème de M. Borel sur les nombres absolument normaux et détermination effective d'un tel nombre. Bulletin de la Société Mathématique de France 45, 125–144 (1917)
40. Shallit, J., Breitbart, Y.: Automaticity I: Properties of a measure of descriptional complexity. Journal of Computer and Systems Science 53(1), 10–25 (1996)
41. Weisstein, E.W.: Transcendental number. From MathWorld–A Wolfram Web Resource, http://mathworld.wolfram.com/TranscendentalNumber.html

Learning Families of Closed Sets in Matroids*

Ziyuan Gao[1], Frank Stephan[2], Guohua Wu[3], and Akihiro Yamamoto[4]

[1] Department of Mathematics,
National University of Singapore, Singapore 117543, Republic of Singapore
ziyuan84@yahoo.com
[2] Department of Mathematics and Department of Computer Science,
National University of Singapore, Singapore 117543, Republic of Singapore
fstephan@comp.nus.edu.sg
[3] School of Physical and Mathematical Sciences
Nanyang Technological University, Singapore 637371
guohua@ntu.edu.sg
[4] Graduate School of Informatics, Kyoto University
Yoshida Honmachi, Sakyo-ku, Kyoto 606-850, Japan
akihiro@i.kyoto-u.ac.jp

Abstract. In this paper it is studied for which oracles A and which types of A-r.e. matroids the class of all A-r.e. closed sets in the matroid is learnable by an unrelativised learner. The learning criteria considered comprise in particular criteria more general than behaviourally correct learning, namely behaviourally correct learning from recursive texts, partial learning and reliably partial learning. For various natural classes of matroids and learning criteria, characterisations of learnability are obtained.

1 Introduction

This paper extends previous studies on the learnability of mathematically defined objects. While Stephan and Ventsov [28] mainly investigated the learnability of the full class of substructures (all ideals of a ring, all subspaces of a vector space, ...), Harizanov and Stephan [12] looked more at structures where one wants to learn only r.e. subobjects of a certain type and ignores all other subobjects. The subobjects considered by Harizanov and Stephan [12] were the r.e. linear subspaces of a recursive vector space or, more generally, the r.e. closed sets in an r.e. matroid. Indeed, matroids are a generalisation of two known concepts: classes of sets closed under equivalence relations and classes of sets closed under linear combinations inside vector spaces. They are described by a closure operation Φ and in general one is interested in the class of the r.e. closed sets, that is, in the class

$$C_\Phi = \{L : \exists H\, [L = \Phi(H)] \wedge \exists \text{ recursive } f\, [H = \text{range}(f)]\}.$$

* Part of this paper was written during F. Stephan's sabbatical leave to the University of Kyoto. F. Stephan is partially supported by grant R252-000-420-112.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 120–139, 2012.

Harizanov and Stephan [12] investigated the learnability properties of this class in various contexts, in particular for k-thin vector spaces [16]. It turns out that the class C_{Φ} is behaviourally correctly learnable if and only if the matroid is Noetherian, that is, every closed set L is generated by a finite set D, that is, satisfies $L = \Phi(D)$ for a finite set D. Starting from this result, further investigations in the present work go into two direcions.

(a) It is shown how much the qualities of the learner in the learnable case can be improved. Indeed, it is shown that whenever every closed set is r.e. in an r.e. matroid, then the class of all closed sets is behaviourally correctly learnable and this learner can be chosen to be consistent, conservative and confident; actually, the learner is confident in the very strict way as it makes only a constant amount of mind changes where this constant bound is the dimension of the matroid. This carries over the results for recursive Artinian rings [28] to matroids; de Brecht, Kobayashi, Tokunaga and Yamamoto [2] considered more general closure systems of recursive sets which then do not have the mind change bound. Kasprzik and Kötzing [17] considered a related model of string extension learning.

(b) The second direction (which is the main novel part of the present work) is to ask what can be said about the learnability of more general matroids. As those which are not Noetherian are also not behaviourally correctly learnable, one has to consider learning criteria which are more general than behaviourally correct learning. Natural candidates are obtained by either making the notion of text more restrictive or by relaxing the convergence requirement.

- Behaviourally correct learning from recursive texts: Here the restrictions on the text simplify the learning task as the learner deals only with friendly modes of data presentation.
- Partial learning: Here the learner does not converge to a hypothesis but singles out the valid hypothesis by outputting it infinitely often while all other hypotheses are output only finitely often.
- Reliable partial learning: This is a restriction of partial learning where the learner either partially learns a set or outputs on every text for the set each hypothesis only finitely often.

Osherson, Stob and Weinstein [23, Exercise 7.5A] showed that already the class of all r.e. sets is partially learnable and therefore the same holds for the class of the r.e. closed sets of a given r.e. matroid. Hence one would like to study the limits of this learning notion by looking at the more general question for which A-r.e. matroids $(\mathbb{N}, \Phi)$ the class C_{Φ}^{A} of all closed A-r.e. sets is learnable under a suitable criterion. Here (reliably) partially learning is considered where the learner itself can conjecture indices from a given acceptable numbering of all A-r.e. sets, but the learner should not have access to the oracle A itself. As the learner can put some of the computations of finding the right hypothesis into the index of an A-recursive enumeration procedure, it follows that there is still some indirect access to A in this way. While partial learners and reliable partial learners cannot exploit that well, the behavioural correct learners can exploit this indirect usage of the oracle and so the corresponding learnability results often generalise to all A-r.e. matroids.

A further learning criterion considered is that of confident partial learning. On one hand, this criterion is much more restrictive than partial learning and does not permit to learn all cofinite sets. On the other hand, certain classes which are not behaviourally correctly learnble are learnable under this criterion and it permits to go beyond Angluin's tell-tale condition [1]. For matroids, it is shown that one one hand the class of closed sets of Noetherian K-r.e. matroids is confidently partially learnable while on the other hand there are some quite simple r.e. matroids where the class of closed r.e. sets is not confidently partially learnable.

Metakides and Nerode [18,19] gave the definition of an r.e. matroid using a hull operation as follows; the reader is referred to the textbooks and papers of Calude [4,5], Odifreddi [21,22], Rogers [26] and Soare [27] for background on recursion theory.

Definition 1 (Metakides and Nerode [18,19]). An r.e. matroid $(\mathbb{N}, \Phi)$ is given by an enumeration operator Φ mapping subsets of $\mathbb{N}$ to subsets of $\mathbb{N}$ such that the following axioms hold for all sets $R, S \subseteq \mathbb{N}$ and all $a, b \in \mathbb{N}$:

- $S \subseteq \Phi(S)$;
- $\Phi(\Phi(S)) = \Phi(S)$;
- $R \subseteq S \Rightarrow \Phi(R) \subseteq \Phi(S)$;
- If $a \in \Phi(\Phi(S) \cup \{b\}) - \Phi(S)$ then $b \in \Phi(\Phi(S) \cup \{a\})$;
- $\Phi(S) = \bigcup\{\Phi(D) : D \text{ is finite and } D \subseteq S\}$.

Furthermore, sets of the form $\Phi(S)$ are called closed sets and $C_\Phi = \{\Phi(S) : S$ is an r.e. subset of $\mathbb{N}\}$. A natural generalisation is that of an A-r.e. matroid where the closure operation is r.e. relative to A and also the class C_Φ^A of interest is the class of A-r.e. closed sets in the matroid.

Remark 2. If $\approx$ is an r.e. equivalence relation then $\Phi(S) = \{a : \exists b \in S\, [a \approx b]\}$ defines an r.e. matroid.

A matroid is called Noetherian iff for every S there is a finite subset $D \subseteq S$ with $\Phi(S) = \Phi(D)$. Noetherian matroids have a constant c, called the dimension, such that one can chose the D above as having at most c elements.

The linear hull in finite-dimensional vector spaces (coded into $\mathbb{N}$) is an example of a Noetherian r.e. matroid.

Let A be a coinfinite subset of $\mathbb{N}$. Then by letting $\Phi(S) = A$ for $S \subseteq A$ and $\Phi(S) = \mathbb{N}$ for $S \not\subseteq A$ one gets an A-r.e. matroid with exactly two closed sets: A and $\mathbb{N}$.

This example can be generalised to matroids for which there is a finite set D such that every closed set S satisfies $S = \Phi(D \cap S)$. For each A-r.e. matroid of this kind it holds that the class C_Φ^A can be explantorily learnt with at most $|D|$ mind changes, as the learner has only to track which elements of D have been seen so far and output an A-r.e. index of the set generated by the corresponding subset of D.

Let A be a proper subset of $\mathbb{N}$. Then there is an A-r.e. matroid having the closed sets A, $A \cup \{a\}$ for every $a \in \mathbb{N} - A$ and $\mathbb{N}$. This is a Noetherian matroid

of dimension 2: $A = \Phi(\emptyset)$, $A \cup \{a\} = \Phi(\{a\})$ and $\mathbb{N} = \Phi(\{a, b\})$ where a, b are distinct nonelements of A.

If $(\mathbb{N}, \Phi)$ is a Noetherian r.e. matroid then the relation $a, D \mapsto \Phi(D)(a)$ is recursive. The reason is that one can, for a fixed set I of size c, search until one of the following two conditions hold:

- $a \in \Phi(D)$, that is, $\Phi(D)(a) = 1$;
- There is E with $|E| + 1 = c$, $D \subseteq \Phi(E)$ and $I \subseteq \Phi(E \cup \{a\})$, that is, $\Phi(D)(a) = 0$.

The function $a, D \mapsto \Phi(D)(a)$ is called the dependence relation associated to Φ.

A set S is called *independent* iff $a \notin \Phi(S - \{a\})$ for any $a \in S$. The emptyset is independent and subsets of independent sets are independent. A matroid is Noetherian iff every independent set is finite iff there is a maximum c of the possible cardinalities of independent sets. This constant c is called the dimension of the matroid.

Example 3. An r.e. matroid is called *almost Noetherian* iff there are finitely many r.e. sets $E_0, E_1, \ldots, E_n$ such that every r.e. closed set L is of the form $\Phi(D \cup E_m)$ for some finite D and $m \in \{0, 1, \ldots, n\}$. Every k-thin vector space [16] is an example of an almost Noetherian r.e. matroid which is not Noetherian.

2 Learnability

Gold [11] laid the foundations of inductive inference and defined that a class C of r.e. sets is explanatory learnable iff there is a learner M which outputs on every text T for a language $L \in C$ an infinite sequence $e_0, e_1, \ldots$ of indices such that almost all e_n are actually the same index e for the set L with respect to a preassigned hypothesis space, usually just a fixed acceptable numbering of all r.e. sets. Here a text is an infinite sequence $a_0\, a_1\, \ldots$ of elements from $\mathbb{N}$ plus perhaps a pause symbol and T is *a text for* L if the range of T, that is, the set of all members of $\mathbb{N}$ occurring in T, is equal to L. Furthermore, for a finite prefix σ of T, that is, for a σ of the form $a_0\, a_1\, \ldots\, a_n$, one let $M(\sigma)$ denote the last hypothesis output by M while processing this initial part of T; without loss of generality M starts with a hypothesis of $\emptyset$ so that $M(\sigma)$ is always defined. It is required that M is recursive and thus induces a recursive function from $\mathbb{N}^*$ to $\mathbb{N}$ which maps σ to $M(\sigma)$. In this context it is noted that, whenever talking of sequences of data, it is understood that pause symbols can arise as well and so L^* actually means $(L \cup \{\#\})^*$, but for convenience reasons the first is written in place of the second. It is clear from the context which of the two is meant by "L^*".

Subsequently, the model of Gold had been expanded and thoroughly studied. So in addition to explantory convergence, additional criteria where introduced which were satisfied by some but not all learners.

- M is consistent iff $W_{M(\sigma)}$ contains range(σ) for all strings σ;
- M is conservative iff range($\sigma\tau$) $\not\subseteq W_{M(\sigma)}$ whenever $M(\sigma\tau) \neq M(\sigma)$;

– M makes at most c mind changes iff there are no $c+2$ strings $\tau_0, \tau_1, \ldots, \tau_c$, τ_{c+1} such that $M(\tau_0\tau_1\ldots\tau_b) \neq M(\tau_0\tau_1\ldots\tau_{b+1})$ for $b = 0, 1, \ldots, c$.

Note that when learning the closed sets in matroids, one can assume that the first hypothesis is $\Phi(\emptyset)$ and therefore the definition of learning with bounded mind changes does not need to conjecture a special waiting symbol until the first hypothesis is ready.

The interested reader is referred to the two editions of the text book "Systems that learn" for more information on inductive inference [13,23]. The next result shows that Noetherian matroids have learners which satisfy all these properties simultaneously.

Theorem 4. *The class C_Φ of the closed sets in a Notherian r.e. matroid of dimension c has a consistent and conservative learner which makes at most c mind changes.*

Proof. The learner produces from the text $a_0\, a_1\, \ldots$ all sets of the form $\{a_{d_0}, a_{d_1}, \ldots, a_{d_e}\}$ where $d_k = \min\{b : \forall \ell < k\, [d_\ell < b] \wedge a_b \notin \Phi(\{a_\ell : \ell < b\})\}$ for $k = 0, 1, \ldots, e$. Note that one can find a_{d_k} only for $k < c$ and the initial hypothesis is $\Phi(\emptyset)$ which will be replaced by sets of the form $\Phi(\{a_{d_0}, a_{d_1}, \ldots, a_{d_e}\})$ for $e = 0, 1, \ldots, c-1$; each mind change is caused by a non-element of the previous hypothesis and thus the learner is conservative. Furthermore, it is consistent as an update is done whenever a datum observed is not generated by the previous hypothesis. ∎

A learner M is behaviourally correctly learning a set L iff M outputs on every text T of L an infinite sequence $e_0\, e_1\, \ldots$ of hypotheses such that $W_{e_k} = L$ for almost all k. Note that a behaviourally correct learner only converges semantically and not syntactically, so it is okay if every hypothesis is different from all previous ones. A class C is behaviourally correctly learnable iff there is a learner M which behaviourally correctly learns every language in C. In the case of learning A-r.e. languages, the learner uses A-r.e. indices from a given acceptable numbering of all A-r.e. sets. Here acceptable is meant in the strict way such that for every further A-r.e. numbering $V_0, V_1, \ldots$ of A-r.e. sets there is a recursive function f such that $W^A_{f(e)} = V_e$ for all e; several learning algorithms would fail in the case that one takes the weaker notion of acceptable numbering where the above f is permitted to be A-recursive in place of recursive. Note that acceptable numberings of all A-r.e. sets in the strict sense exist for every oracle A. The following result is mainly a mirror image of the corresponding results in the world of recursive sets [2,3], but some adjustments have to be done so that a recursive learner can deal with A-r.e. sets.

Theorem 5. *If $(\mathbb{N}, \Phi)$ is an A-r.e. matroid then the class C^A_Φ of the A-r.e. closed sets is behaviourally correctly learnable (using A-r.e. indices and a recursive learner) iff $(\mathbb{N}, \Phi)$ is Noetherian.*

Proof. If $(\mathbb{N}, \Phi)$ is an A-r.e. Noetherian matroid then one can at each stage conjecture $\Phi(D)$ where D is the data observed so far; note that there is a recursive

function f which finds for D an A-r.e. index $f(D)$ of $\Phi(D)$. Namely there is an A-r.e. enumeration of sets $V_D = \Phi(D)$ whose indices are finite sets (given in some canonical form) and f is then the recursive translations from the indices of the sets V_D into indices of the form $W^A_{f(D)}$ with respect to suitable acceptable numbering of all A-r.e. sets. As every closed set L is generated by a finite subset E, that is, $L = \Phi(D)$ for all D with $E \subseteq D \subseteq L$, the indices $f(\text{range}(\sigma))$ are correct for all sufficiently long prefixes σ of any fixed text for L. Hence the class is behaviourally correctly learnable.

For the converse direction, assume that M is a behaviourally correct learner for the matroid $(\mathbb{N}, \Phi)$ and L an A-r.e. closed set. Then there is a locking sequence σ for L, that is, $\sigma \in L^*$ and $M(\sigma\tau)$ is an index for L for all $\tau \in L^*$. As the range D of σ is finite, $\Phi(D)$ is an A-r.e. subset of L and M learns that set as well. As $M(\sigma\tau)$ is an index for L for all $\tau \in \Phi(D)^*$ it follows that these two sets must coincide and $L = \Phi(D)^*$. Hence every closed A-r.e. set is finitely generated.

Now assume that there is a set $\Phi(S)$ such that $\Phi(S) \neq \Phi(D)$ for all finite D. As shown before, the superset $\mathbb{N}$ is finitely generated and $\mathbb{N} = \Phi(D)$ for some finite set D, let D be the minimal such set so that D is independent. As $\Phi(S)$ is not finitely generated, there is an independent subset E of $\Phi(S)$ with $|E| = |D| + 1$. It follows that there is some $a \in E$ such that $D \cup \{a\}$ is independent by the rules of independent sets in matroids. This gives then that $a \notin \Phi(D)$ and $\Phi(D) \neq \mathbb{N}$, in contradiction to the assumption. Hence every closed set is finitely generated and A-r.e.; so $(\mathbb{N}, \Phi)$ is a Noetherian matroid. ∎

This characterisation shows that the study of behaviourally correct learning in the context of matroids is only interesting when the setting is a bit modified in order to make the learner more powerful. So it might be interesting to ask for which r.e. matroids the class of all closed r.e. sets is learnable from recursive texts. This is not true for the full matroid where every set is closed: the class of all r.e. languages is not behaviourally correctly learnable from recursive texts. Example 6 and Theorem 7 show that there are non-Noetherian matriods for which the class of closed r.e. sets is behaviourally correctly learnable from recursive texts.

Example 6. Let B be a maximal set, that is, a coinfinite r.e. set satisfying that every r.e. superset is either a finite variant of B or is cofinite. Let $\approx$ be defined such that

$$x \approx y \Leftrightarrow \forall z\, [\min\{x, y\} \leq z < \max\{x, y\} \Rightarrow z \in B]$$

and let Φ be the corresponding closure operator. Then the class C_Φ of all r.e. closed sets consists only of finite and cofinite sets. Furthermore, $\mathbb{N}$ is the ascending union of finite members of C_Φ and so C_Φ is not behaviourally correctly learnable. But, as the class of finite and cofinite sets is an indexed family which is a superclass of C_Φ, the class C_Φ is behaviourally correctly learnable from recursive texts [6, Theorem 28 and Corollary 33].

Theorem 7. *Let C_Φ be the class of all closed r.e. sets in an almost Noetherian r.e. matroid $(\mathbb{N}, \Phi)$. Then C_Φ is behaviourally correctly learnable from recursive texts.*

Proof. Let $E_0, E_1, \ldots, E_n$ be the n r.e. sets witnessing that $(\mathbb{N}, \Phi)$ is almost Noetherian and for each m let $E_{m,s}$ be the set of numbers enumerated into E_m within s steps. Let $\tilde{\varphi}_e$ be the e-th recursive text from a list of all partial-recursive texts; note that some of the $\tilde{\varphi}_e$ are partial. Now define the following behaviourally correct learner M for C_Φ.

$M(\sigma)$ outputs an r.e. index for $\Phi(X_\sigma)$ where X_σ is the union of range(σ) and all E_m for which there exists an e with
- $\sigma \preceq \tilde{\varphi}_e$;
- $E_{m,e} \subseteq$ range(σ);
- $E_{m,|\sigma|} \subseteq$ range$(\tilde{\varphi}_e)$.

The first condition on σ makes sure that $\tilde{\varphi}_e$ is a text extending σ; the second condition wants that the index is permitted by L so that only finitely many e get permitted when $E_m \not\subseteq L$; the third condition wants to make sure that every index e which gets permitted infinitely often is a superset of E_m.

Now it is shown that when learning $L \in C_\Phi$ from a recursive text $\tilde{\varphi}_e$, then for almost all prefixes σ of this text, X_σ is the union of range(σ) and all those E_m which satisfy $E_m \subseteq L$. Hence X_σ contains in particular the "right E_m", so that $L = \Phi(D \cup E_m)$ for some D, as well as some E_m which are "not harmful"; furthermore, the elements of the just mentioned D are also in X_σ for sufficiently long prefixes σ of $\tilde{\varphi}_e$. Hence almost all conjectures of M are correct.

Now the verification of the learner is given in detail. Let σ be a prefix of $\tilde{\varphi}_e$ which is so long that the following conditions are satisfied:

- $e \leq |\sigma|$;
- $D \subseteq$ range(σ);
- if $E_m \not\subseteq L$ then the least $c \in E_m - L$ satisfies that there is no index $d \leq c$ such that $\sigma \preceq \tilde{\varphi}_d \wedge E_{m,|\sigma|} \subseteq$ range$(\tilde{\varphi}_d)$.

It is clear that the first two conditions are true for sufficiently long prefixes σ of $\tilde{\varphi}_e$. The third condition is true for sufficiently long σ as either φ_d is a finite sequence and therefore cannot extend sufficiently long σ or φ_d is a text which is not for L and therefore does not extend sufficiently long prefixes of the text $\tilde{\varphi}_e$ for L or φ_d is a text for L which then does not enumerate c and therefore $E_{m,|\sigma|} \not\subseteq$ range$(\tilde{\varphi}_d)$ for sufficiently long $\sigma \preceq \tilde{\varphi}_e$.

As there are only finitely many E_m and as the c in the third condition bounds the number of d over which is quantified in the third condition, it follows that one has the conjunction of all these conditions is satisfied for sufficiently long prefixes σ of $\tilde{\varphi}_e$. As a consequence, X_σ is, for sufficiently long σ, just the union of D and range(σ) and those E_m which are subsets of L; hence $\Phi(X_\sigma) = L$ for those σ and the learner M behaviourally correctly learns L from each recursive text of L. ∎

Remark 8. Every A-recursive behaviourally correct learner (from A-recursive texts) can be turned into a recursive behaviourally correct learner (from A-recursive texts). The reason is that the algorithm to compute the hypothesis can be incorporated into the hypothesis where the computation then, as it is

an A-recursive enumeration, has access to the oracle A. Thus the class of the closed sets of any A-r.e. Noetherian matroid has a recursive behaviourally correct learner using A-r.e. indices.

Furthermore, one can generalise Example 6 and Theorem 7 to show that there are A-r.e. matroids where the class of A-r.e. closed sets has a recursive learner which learns these sets behaviourally correctly from A-recursive text outputting A-r.e. indices.

One can generalise behaviourally correct learning in the way that one only requires that almost every hypothesis is a finite variant of the target language; here the number of errors might be different from hypothesis to hypothesis [7,8,24]. Harrington [7] showed that the class of all recursive functions can be learnt by an algorithm which outputs on data for a function f a sequence $e_0, e_1, \ldots$ of hypotheses such that almost all φ_{e_k} are finite variants of f and furthermore total. Hence one can also, when reading an A-recursive text, produce a sequence $e_0, e_1, \ldots$ of indices such that $\tilde{\varphi}^A_{e_k}$ is for almost all k a text which is a finite variant of the input text; thus, when looking at the ranges $W^A_{e_k}$ in place of the texts itself, the corresponding indices are almost all finite variants of the set to be learnt. It follows that the class of all A-r.e. sets is behaviourally correctly learnable with at most finitely many errors at almost every hypothesis from A-recursive texts.

3 Partial Learning

Osherson, Stob and Weinstein [23, Exercise 7.5A] introduced the notion of a partial learner and showed that a partial learner can learn the class of all r.e. sets. Minicozzi [20] introduced the notion of reliable learning; although she introduced it for the learning criterion of explanatory learning, it can be brought over to other learning criteria as well. Osherson, Stob and Weinstein [23, Section 4.6.2] also introduced the notion of confident learning where a confident learner has to converge on every text, even a text for a non-r.e. language, to some hypothesis according with respect to the convergence criterion which applies.

Definition 9 (Minicozzi [20], Osherson, Stob and Weinstein [23]). A learner M partially learns a language L iff M, given any text T for L, outputs on this text exactly one index infinitely often and that is an index for L; M partially learns a class C iff M partially learns every $L \in C$.

A partial learner M is reliable iff M for every set L either M outputs on each text for L exactly one index e infinitely often which satisfies $L = W_e$ or M outputs on each text for L each index only finitely often.

A partial learner M is confident iff M for every set L and every text for L outputs on this text exactly one index infinitely often.

Remark 10. One might also ask whether there is a counterpart of partial learning when combined with semantic convergence which gives someting interesting when learning in relativised worlds. Here one could say that a learner M behaviourally correctly partially learns a class C of A-r.e. languages iff M outputs on every text for a language $L \in C$ an infinite sequence $e_0, e_1, \ldots$ of indices such

that $W^A_{e_k} = L$ for infinitely many k while for every $H \neq L$ there are only finitely many k with $W^A_{e_k} = H$. Unfortunately this notion does not give anything interesting: Relativising the result of Osherson, Stob and Weinstein [23], there is an A-recursive partial learner which learns all A-r.e. languages; one can in addition have that the learner uses an underlying one-one numbering so that no incorrect language appears on a text by giving infinitely many different indices for that language. Hence one can make a new learner N which on input σ outputs an A-r.e. index e_σ enumerating $W^A_{M(\sigma)}$. That learner has then the required learning properties. Hence this recursive learner learns even the class of all A-r.e. sets using A-r.e. indices.

Remark 11. If M partially learns L then there is a sequence $\sigma \in L^*$ and an index e for L such that for every $\tau \in L^*$ there is an $\eta \in L^*$ such that $M(\sigma\tau\eta) = e$; such a sequence σ is called a locking sequence for L. If such σ, e would not exist then one could find sequences $\tau_0, \tau_1, \ldots \in L^*$ such that for all indices $e_0, e_1, \ldots$ of L the following holds:

- The sequence $\tau_0\tau_1 \ldots \tau_d$ contains all elements of L below d plus perhaps some other elements of L;
- $M(\tau_0\tau_1 \ldots \tau_d\eta) \neq e_d$ for all $\eta \in L^*$.

These two conditions together would then give that $\tau_0\tau_1 \ldots$ is a text for L and that M does not output any index e_d of L on this text infinitely often.

If M is a reliable partial learner learning L, $H \neq L$ and e an index of L then there is a sequence $\sigma \in H^*$ such that $M(\sigma\tau) \neq e$ for all $\tau \in H^*$. If such a σ would not exist, one could construct a text for H on which M outputs e infinitely often in contrast to M being a reliable partial learner.

In the case of a partial learner which is not reliable, the sequence $\sigma \in H^*$ is only guaranteed to exist when H is in the class of sets to be learnt.

Theorem 12. *The class of all cofinite sets is not confidently partially learnable.*

Proof. Assume that M is a partial confident learner. Furthermore, choose $a_0, a_1, a_2, \ldots$ such that for every n, $a_{n+1} > a_n$ and $a_{n+1} > \varphi^K_e(a_0, a_1, \ldots, a_n)$ for all $e \leq n$ where $\varphi^K_e(a_0, a_1, \ldots, a_n)$ is defined. Let $L = \mathbb{N} - \{a_0, a_1, \ldots\}$. By confidence and Remark 11, there is an index d and a $\sigma \in L^*$ such that for all $\tau \in L^*$ there is an $\eta \in L^*$ with $M(\sigma\eta\tau) = d$. With except perhaps one exception, d is not the index of the cofinite set $\mathbb{N} - \{a_0, a_1, \ldots, a_n\}$ and not output infinitely often on any text for this set. Hence, for all but at most one n, there is a sequence $\tau_n \in (\mathbb{N} - \{a_0, a_1, \ldots, a_n\})^*$ such that for all $\eta \in (\mathbb{N} - \{a_0, a_1, \ldots, a_n\})^*$ it holds that $M(\sigma\tau_n\eta) \neq d$. The τ_n and the maximum value occuring in τ_n can be found using a partial K-recursive function, let $\varphi^K_e(a_0, a_1, \ldots, a_n)$ be this value and let n be so large that $n > e$ and $\varphi^K_e(a_0, a_1, \ldots, a_n)$ is defined. By the choice of a_{n+1} it holds that $a_{n+1} > \varphi^K_e(a_0, a_1, \ldots, a_n)$ and that therefore $\tau_n \in L^*$ which contradicts the property according to which σ was chosen. Hence the partial confident learner M cannot exist. ∎

It follows immidiately that also the class of all recursive sets is not confidently partially learnable.

Furthermore, if A is a maximal set and $\Phi(S) = A \cup S$ then every r.e. closed set in $(\mathbb{N}, \Phi)$ is either a cofinite superset of A or the union of A with a finite set. This class is also not confidently partially learnable, as one can take the sequence $a_0, a_1, \ldots$ in above proof outside A. Hence there is a quite easy matroid for which the closed r.e. sets are not confidently partially learnable.

The next theorem shows that there is nevertheless a large amount of confidently partially learnable classes. As a corollary one obtains that the class of closed sets of any K-r.e. Noetherian matroid is confidently partially learnable.

Theorem 13. *If a class C of A-r.e. sets is confidently explanatorily learnable relative to the oracle K from informant then C is confidently partially learnable from text without any oracle usage.*

Proof. Let M be the K-recursive learner using A-recursive indices from an acceptable numbering of all A-r.e. sets. Without loss of generality, M codes at every mind change into the index e the finite sets $D_{pos(e)}$ and $D_{neg(e)}$ of positive and negative data, respectively, on which the new hypothesis is based. Furthermore, let M_s be an approximation of M for s steps such that M_s has the same behaviour with respect to the indices output as M.

The new confident partial learner N outputs an index e at least n times iff there is a stage $s > n$ and sets D_{pos^*}, D_{neg^*} such that D_{pos^*} is contained in the first s data items of the text, the first n data items of the text are contained in D_{pos^*}, no data item in D_{neg^*} belongs to the first s data items of the text, $\{0, 1, \ldots, n\} \subseteq D_{pos^*} \cup D_{neg^*}$, M_s with inputs D_{pos^*} and D_{neg^*} outputs e, $D_{pos(e)} \subseteq D_{pos^*}$ and $D_{neg(e)} \subseteq D_{neg^*}$.

Now consider any text T and let e be the index to which M converges on T. The index e is also output infinitely often by N. To see this one shows that N outputs e at least n times by taking for n the parameters $D_{pos^*} = (\{0, 1, 2, \ldots, n\} \cap L) \cup D_{pos(e)}$, $D_{neg^*} = (\{0, 1, 2, \ldots, n\} - L) \cup D_{neg(e)}$ and s so large that all elements of D_{pos^*} have appeared among the first s elements of the text and that M_s on the given input has converged to the value of M on this input.

If a further index d is an old hypothesis which is superseded by e then there is an n such that $D_{pos(e)} \cup D_{neg(e)} \subseteq \{0, 1, \ldots, n\}$, all elements of $D_{pos(e)}$ have been observed within the first n elements of the text and for all $s \geq n$ and all disjoint supersets $D_{pos^*} \supseteq D_{pos(e)}$ and $D_{neg^*} \supseteq D_{neg(e)}$, M_s with inputs D_{pos^*} and D_{neg^*} outputs e or outputs a hypothesis d' with $D_{pos(d')} \supseteq D_{pos(e)}$ and $D_{neg(d')} \supseteq D_{neg(e)}$. That hypothesis d' is different from d as $D_{pos(d')} \neq D_{pos(d)} \vee D_{neg(d')} \neq D_{neg(d)}$. It follows that d is output by N at most n times.

If a hypothesis d is output by M_s but not by M on inputs $D_{pos(d)}$ and $D_{neg(d)}$ then N outputs d at most where n is the point from which on M_s is equal to M on the inputs $D_{pos(d)}$ and $D_{neg(d)}$.

If a hypothesis d is output by some M_s on some data $D_{pos(d)}$ and $D_{neg(d)}$ where $D_{neg(d)}$ contains some element within the first n elements of T for some n then d is output at most n times.

From this case distinction follows that every d different from e is output only finitely often by N and hence N is a confident partial learner which converges (in the partial sense on the text) to the same indices as M (in the explanatory sense on an informant for range(T)). ∎

This result together with Theorem 4 directly gives the following corollary.

Corollary 14. *Assume that $A \leq_T K$ and $(\mathbb{N}, \Phi)$ is a Noetherian A-r.e. matroid. Then the class C^A_Φ of the closed sets in this matroid is confidently partially learnable.*

Remark 15. Gold's class [11] consisting of all finite sets and one infinite set can be confidently partially learned by outputting a fixed index of the infinite set when a new element is observed and a canonical index for the current range when no new element is observed. But this class is not confidently learnable from informant relative to any oracle, hence Theorem 13 is not a characterisation.

The next result gives a useful criterion for classes to be reliably partially learnable.

Theorem 16. *Every uniformly K-recursive class is reliably partially learnable using a padded version of the given indexing as hypothesis space.*

Proof. Let $V_0, V_1, \ldots$ be a uniformly K-recursive numbering containing the class C to be learnt and let f be a two-place recursive function with $W^K_{f(d,e)} = V_d$ for all d, e. The second parameter is just a padding parameter used to code from which point onwards the language V_d is different from all $V_{d'}$ with $d' < d$. For the reliable partial learner, it is enough to say how often an index has to be output at least; that information can then be used to generate the learner. So the learner for C will output only indices of the form $f(d, e)$ and it will output on a text $a_0\, a_1\, a_2\, \ldots$ at least n times iff there is a stage $s > n$ such that

- e is the least number such that for all $d' < d$ there is an $x < e$ with $V_{d',s}(x) \neq V_{d,s}(x)$ and
- for all $x < n$, $x \in V_{d,s}$ iff $x \in \{a_0, a_1, \ldots, a_s\}$.

If d is the least index of the set L to be learnt and if e is the least number such that for every $d' < d$ there is an $x < e$ with $V_{d'}(x) \neq V_d(x)$ then there is for every n an $s > n$ where $f(d, e)$ qualifies to be output. Hence $f(d, e)$ is output infinitely often.

If $V_d(x) \neq L(x)$ for some x then for all sufficiently large n and all $s > n$, $V_{d,s}(x) \neq L(x)$ and x has appeared within the first n elements of the text iff x is in L. One can then see that $f(d, e)$ is not output n times for these large n and hence output only finitely often.

If there is $d' < d$ with $V_{d'} = V_d$ then every bound e will qualify only finitely often as for all sufficiently large s there is no $x < e$ with $V_{d',s}(x) \neq V_{d,s}(x)$. Hence each $f(d, e)$ is output only finitely often.

If d is the minimal index of L but e is the incorrect bound then one can again see that $f(d, e)$ is output only finitely often.

This case distinction shows that on an input text for a language L, the index $f(d,e)$ is output infinitely often iff $V_d = L$, d is the least index with this property and e is the least bound witnessing that d is the least index. This shows that the learner is correct and also that the learner is reliable. ∎

Example 17. Not every reliably partially learnable class is uniformly K-recursive. For example, let $T_0, T_1, T_2, \ldots$ be a uniformly recursive sequence of trees over $\mathbb{N}^*$ which have at most one infinite branch and for which the tree is unique to the branch, that is, no two trees have the same infinite branch. Furthermore, let $V_e = \{(x, f_e(x)) : x \in \mathbb{N}\}$ where f_e is the unique infinite branch of T_e. Note that the class $\{V_0, V_1, \ldots\}$ needs not to be uniformly K-recursive as there are recursive trees with exactly one infinite branch f such that f is not even arithmetic, that is, much more complicated than K-recursive. Now let g be a recursive function which translates the V-index into that of an A-r.e. set with respect to some acceptable numbering of these sets where A is a suitably chosen oracle. The learner outputs $g(e)$ at least n times iff after some time for $x = 0, 1, \ldots, n$ exactly one pair (x, y_x) has been observed and $y_0\, y_1\, \ldots\, y_x$ is a node on the tree T_e. It is clear that the learner outputs an index $g(e)$ infinitely often iff the observed data is the set V_e.

4 Learnability Properties of Full and Noetherian Matroids

In this section the two extremes of A-r.e. matroids are investigated: The first case is the full matroid where every A-r.e. set is also closed in the matroid, as Φ is the identity operator. The second case are the Noetherian matroids; these satisfy that the closed sets are all uniformly A-recursive.

For the convenience of the reader, the following well-known facts and properties of Turing degrees are summarised: $A \leq_T K$ means that A is Turing reducible to the halting problem. By Shoenfield's Limit Lemma, this is equivalent to saying that A can be approximated in the limit by a sequence of uniformly recursive sets A_s; that is, if $x \in A$ then $x \in A_s$ for almost all s and if $x \notin A$ then $x \notin A_s$ for almost all s. Furthermore, there is a real $r \equiv_T A$ such that there is some recursive sequence $r_0, r_1, \ldots$ of rationals with $r = \lim_s r_s$ and $\{m : r_m < r\} \equiv_T A$. Here the number r itself can be chosen not to be a rational, so $r \neq r_m$ for all m. In the case that A is not recursive (what is the only interesting case), one can in addition chose r such that $\{q \in \mathbb{Q} : q > r\}$ is not an r.e. set.

The set A' denotes the Turing jump of A or the halting problem relative to A; A' is A-r.e. and for every A-r.e. set B there is a recursive function f with $x \in B \Leftrightarrow f(x) \in A'$. A set A has low Turing degree (or just "A is low") iff $A' \leq_T K$, that is, the halting problem relative to A is not more complicated than the unrelativised halting problem. A is low_2 iff $A'' \leq_T K'$. Note that every low set is low_2, A is low implies $A \leq_T K$ and A is low_2 implies $A \leq_T K'$. Of special interest are those sets A which are low_2 and satisfy in addition that $A \leq_T K$. They coincide with those sets such that the class of all A-recursive sets has a uniformly K-recursive listing.

Note that the full matroid satisfies that every set is closed. Hence the class of its closed A-r.e. sets is exactly the class of all A-r.e. sets. Therefore it is interesting to ask when this class is partially learnable. The following theorem gives an answer.

Theorem 18. *Assume that $A \leq_T K'$. Let C be the class of all A-r.e. sets. Then the following statements are equivalent.*

1. *A is low, that is, $A' \leq_T K$;*
2. *C is reliably partially learnable;*
3. *C is partially learnable.*

Proof. The proof is done by a case distinction over several parts.

First, consider the case that A is low. In this case, the class of A-r.e. sets is uniformly K-recursive and hence by Theorem 16 reliably partially learnable; note there that the set $\{(e,x) : x \in W_e^A\}$ is already K-recursive, hence the reliable partial learner from Theorem 16 can actually be made to use the A-r.e. indices. It follows that in this case all three conditions are true.

Second consider the case where $A \leq_T K$ and C is partially learnable. Now let $a_0, a_1, a_2, \ldots$ be a A-recursive sequence of rationals converging to a real r such that the set $\{m : a_m < r\}$ is A-r.e. and has the Turing degree of A'. Such a sequence is known to exist. As $A \leq_T K$ there is a uniformly recursive approximation to the sequence such that $a_m = \lim_s a_{m,s}$. Now let, for a real number q,

$$B_q = \{(m,s) : a_{m,s} < q \vee \exists t > s\, [a_{m,t} \neq a_{m,s}]\}.$$

Furthermore, there is a locking sequence $\sigma \in B_r^*$ and an index e of B_r such that for every $\tau \in B_r^*$ there is an $\eta \in B_r^*$ with $M(\sigma\tau\eta) = e$. Let $b_0 = \max\{a_0, a_1, a_2, \ldots\}$ and note that $b_0 > r$. Now define the following sequence of rationals using the oracle K: Given $b_k > r$, search for a sequence $\tau \in B_{b_k}^*$ such that there is no $\eta \in B_{b_k}^*$ with $M(\sigma\tau\eta) = e$; this τ must exist as M does not output e infinitely often on an text for the r.e. set B_{b_k} which is a proper superset of B_r; note that $B_{b_k} \in C$ as every r.e. set is an A-r.e. set and M outputs on each text for B_{b_k} some other index than e infinitely often. Having τ, τ cannot be a member of B_r^* by the choice of σ. So there is at least one $(m,s) \in B_{b_k} - B_r$ which occurs in τ. Now let $b_{k+1} = \max\{a_{m,s} : (m,s) \text{ occurs in } \tau \text{ and } a_{m,t} = a_{m,s} \text{ for all } t > s\}$. So the $a_{m,s}$ over which the maximum is taken satisfy all $a_{m,s} < b_k$ and furthermore one of them satisfies $r < a_{m,s}$ as otherwise $\tau \in B_r^*$. It follows that $r < b_{k+1} < b_k$. The sequence of the b_k is K-recursive and a subsequence of $a_0, a_1, \ldots$; hence the sequence $b_0, b_1, \ldots$ converges to r from above. Furthermore, by the choice of the sequence $a_0, a_1, \ldots$ one can approximate r from below with an A-recursive and thus with a K-recursive sequence. Hence $r \leq_T K$ and $\{m : a_m < r\} \leq_T K$. It follows that $A' \leq_T K$ and A is low. Again all three conditions are satisfied.

Third, consider the case where $A \leq_T K'$ and $A \not\leq_T K$. Furthermore, let $r = \sum_n 2^{-n} A(n)$ or $r = \sum_n 2^{-n}(1 - A(n))$, just take that version for which $\{q \in \mathbb{Q} : q > r\}$ is not K-r.e.; note that $\{q \in \mathbb{Q} : q < r\}$ is an A-r.e. set; it is even

A-recursive. There is a K-recursive sequence of rationals such that $r = \lim a_n$ and $\{n : a_n < r\} \equiv_T A \oplus K$. Furthermore, $\{n : a_n > r\}$ is not r.e. relative to K as otherwise $\{q \in \mathbb{Q} : q > r\}$ would be r.e. relative to K. Having this, one makes a proof similar to the previous case. There is a uniformly recursive approximation $a_{m,s}$ to a_m; so for all m and almost all s it holds that $a_m = a_{m,s}$. Now let, for each real q,

$$B_q = \{(m, s) : a_{m,s} < q \vee \exists t > s\, [a_{m,t} \neq a_{m,s}]\}.$$

Note that for every rational q and also for $q = r$ it holds that B_q is A-r.e. and hence there is a learner M which partially learns all sets B_q with $q \in \mathbb{Q}$ and also B_r. Hence, there is a locking sequence $\sigma \in B_r^*$ and an index e of B_r such that for every $\tau \in B_r^*$ there is an $\eta \in B_r^*$ with $M(\sigma\tau\eta) = e$. Let $b_0 = \max\{a_0, a_1, a_2, \ldots\}$ and note that $b_0 > r$. Now define the following sequence of rationals using the oracle K: Given $b_k > r$, search for a sequence $\tau \in B_{b_k}^*$ such that there is no $\eta \in B_{b_k}^*$ with $M(\sigma\tau\eta) = e$; this τ must exist as M does not output e infinitely often on any text for the r.e. set B_{b_k} which is a proper superset of B_r. Having τ, τ cannot be a member of B_r^* by the choice of σ. Hence there is at least one $(m, s) \in B_{b_k} - B_r$ which occurs in τ. Now let $b_{k+1} = \max\{a_{m,s} : (m, s)$ occurs in τ and $a_{m,t} = a_{m,s}$ for all $t > s\}$. So the $a_{m,s}$ over which the maximum is taken satisfy all $a_{m,s} < b_k$ and furthermore one of them satisfies $r < a_{m,s}$ as otherwise $\tau \in B_r^*$. It follows that $r < b_{k+1} < b_k$. The sequence of the b_k is K-recursive and a subsequence of $a_0, a_1, \ldots$; hence the sequence $b_0, b_1, \ldots$ converges to r from above. It follows that $\{m : a_m > r\} = \{m : \exists n\, [b_n < a_m]\}$ is a K-r.e. set in contradiction to its choice. From this contradiction follows that the class of all A-r.e. sets is neither partially learnable nor reliably partially learnable. So all three conditions are not satisfied in this case and hence they are equivalent also in this case. ∎

Theorem 19. *Given a set A, it holds that $A \leq_T K$ iff C_Φ^A is reliably partially learnable for every Noetherian A-r.e. matroid $(\mathbb{N}, \Phi)$.*

Proof. One distinguishes three cases:

- first $A \leq_T K$,
- second $A \not\leq_T K$ and every A-recursive function is majorised by a K-recursive one,
- third there is some $f \leq_T A$ which is not majorised by any K-recursive function.

For the first case, consider $A \leq_T K$. Note that for a Noetherian matroid the class C_Φ^A is uniformly A-recursive and thus uniformly K-recursive. Then C_Φ^A is reliably partially learnable by Theorem 16, using the given A-r.e. indices for the learning process.

For the second part of the proof, one assumes for a contradiction that every Noetherian matroid is reliably partially learnable, that $A \not\leq_T K$ and that every A-recursive function is majorised by a K-recursive one. Now one works over binary strings instead of natural numbers and considers trees of binary strings. One

defines a matroid with two A-recursive sets: The set $Br_A = \{\lambda, A(0), A(0)A(1), A(0)A(1)A(2), \ldots\}$ of all prefixes of A and the full binary set $\{0,1\}^*$; $\Phi(S) = Br_A$ for all $S \subseteq Br_A$ and $\Phi(S) = \{0,1\}^*$ for all other S. Now assume that M is a partial reliable learner of this class which outputs the index e infinitely often on the ascending text of Br_A; e is an index for Br_A and therefore not output infinitely often on the text of any other set. Let $f(n)$ be the number of elements seen of the ascending text of Br_A when M outputs e for the n-th time. The function f is A-recursive and hence majorised by a K-recursive function g, that is, $f(n) \leq g(n)$ for all n. Now define the K-recursive tree T to be the set of all strings σ such that every $n \leq |\sigma|$ with $g(n) \leq |\sigma|$ satisfies that M outputs e at least n times on the input sequence $\overline{\sigma}$ which contains all the prefixes $\lambda, \sigma(0), \sigma(0)\sigma(1), \sigma(0)\sigma(1)\sigma(2), \ldots, \sigma$ of σ in ascending order. Then A is an infinite branch of T as M outputs e at least n times on the $g(n)$ first elements of the ascending text of Br_A; on the other hand no set $B \neq A$ is an infinite branch of T as M outputs e only n times on the ascending text of Br_B for some n and then $B(0)B(1)\ldots B(g(n))$ is not on T by the definition of T. So A is the only infinite branch on T and as $T \leq_T K$, it also holds that $A \leq_T K$, in contradiction to the assumption. Hence there is an Noetherian A-recursive matroid whose closed sets cannot be reliably partially learnt.

Third, assume that $f \leq_T A$ is not majorised by any K-recursive function. In other words, the range of f is a set which is hyperimmune relative to K. Now consider the matroid $(\mathbb{N}, \Phi)$ with $\Phi(B) = \mathbb{N} - \text{range}(f)$ if B is disjoint to the range of f and $\Phi(B) = \mathbb{N}$ if B meets the range of f. Assume that there is a reliable partial learner M, an index e and a locking-sequence $\sigma \in (\mathbb{N} - \text{range}(f))^*$ such that for all $\tau \in (\mathbb{N} - \text{range}(f))^*$ there is an $\eta \in (\mathbb{N} - \text{range}(f))^*$ with $M(\sigma\tau\eta) = e$. There is a K-recursive function g such that for every n and every finite set $D \subseteq \{0,1,\ldots,n\} - \text{range}(\sigma)$ there is a $\tau_D \in (\{0,1,\ldots,g(n)\} - D)^*$ such that $M(\sigma\tau_D\eta) \neq e$ for all $\eta \in (\mathbb{N} - D)^*$. This function exists as M does not output e infinitely often on any text of a confinite set (by reliability) and therefore one can use K to find the τ_D for each finite set $D \subseteq \{0,1,\ldots,n\} - \text{range}(\sigma)$ and then take the $g(n)$ to be the maximum of the ranges of the finitely many strings $\sigma\tau_D$ for these D. Now it must be that for $D = \{0,1,\ldots,n\} - \text{range}(f)$ there is an element of range(f) between n and $g(n)$ as otherwise $\tau_D \in (\mathbb{N} - \text{range}(f))^*$ and $M(\sigma\tau_D\eta) \neq e$ for all $\eta \in (\mathbb{N} - \text{range}(f))^*$ in contradiction to the choice of σ. So it follows that for every $n \in \mathbb{N}$ the intersection $\{n+1, n+2, \ldots, g(n)\} \cap \text{range}(f)$ is not empty and that contradicts the fact that range(f) is hyperimmune relative to K. Thus this case does not occur and the reliable partial learner M cannot exist. So also in this case there exists a Noetherian A-recursive matroid whose closed sets cannot be reliably partially learnt. ∎

The two preceding results give together the following corollary.

Corollary 20. *The class of all A-r.e. sets is reliably partially learnable iff A is low.*

5 The Class of All A-Recursive Sets

The following results deal with the question when the class of all A-recursive sets is partially learnable.

Theorem 21. *The class C of all A-recursive sets is reliably partially learnable iff A is low$_2$ and $A \leq_T K$.*

Proof. If $A \leq_T K$ and A is GL_2 then A is low$_2$ then the class of all A-recursive sets is uniformly K-recursive [14]: One could fix an uniformly A-recursive numbering of all partial A-recursive functions and consider a K-recursive function g which dominates the convergence time of each total function in this list; such a g exists as $A \leq_T K$ and A is low$_2$. Now one defines

$$V_d(x) = \begin{cases} 0 & \text{if } \varphi^A_{g(d+x)}(x)\downarrow = 0 \text{ and } \forall y \leq x\, [\varphi^A_{g(d+x)}(x)\downarrow \in \{0,1\}]; \\ 1 & \text{if } \varphi^A_{g(d+x)}(x)\downarrow = 1 \text{ and } \forall y \leq x\, [\varphi^A_{g(d+x)}(x)\downarrow \in \{0,1\}]; \\ 2 & \text{otherwise.} \end{cases}$$

Note that those indices d where $V_d(x) = 2$ for some x are invalid indices for which the padded versions $f(d,e)$ in Theorem 16 will each be output only finitely often. Hence the learner from Theorem 16 will converge to one index of the form $f(d,e)$ where $V_d(x) \in \{0,1\}$ for all x. This is then a characteristic A-recursive index of the set to be learnt.

It follows from Theorem 19 that whenever $A \not\leq_T K$ then the class of all A-recursive sets is partially reliably learnable.

So consider the last case that $A \leq_T K$ and A is not low$_2$. For the way of a contradiction, assume that M is a reliable partial learner of the class of all A-recursive sets. Let ASC be the set of all strinctly ascending sequences. One can now show the following: $(*)$ For every $\sigma \in ASC$ there is τ_σ with $\sigma\tau_\sigma \in ASC$ such that for all η with $\sigma\tau\eta \in ASC$ it holds that $M(\sigma\tau\eta) > \max(\text{range}(\sigma))$.

If this would not be true, then one could find a $\sigma \in ASC$ such that for every τ with $\sigma\tau \in ASC$ there is an η such that $\sigma\tau\eta \in ASC$ and $M(\sigma\tau\eta) < \max(\text{range}(\sigma))$. This would permit to build more than $\max(\text{range}(\sigma)) + 1$ many recursive ascending texts for pairwise different sets on which M outputs an index below $\max(\text{range}(\sigma))$ infinitely often, in particular one such index would be output infinitely often on at least two different ascending texts of two different sets, thus the condition $(*)$ is true.

There is now a K-recursive function which finds one possible τ_σ for σ. Furthermore, let $\tau_{\sigma,s}$ be an approximation to τ_σ such that $\tau_{\sigma,s} \in ASC$ for all $\sigma \in ASC$ and all s.

Let $g(n)$ be the first s such that for all $t \geq s$ and all $\sigma \in ASC$ with $\max(\text{range}(\sigma)) \leq n$ it holds that $\max(\text{range}(\sigma\tau_{\sigma,s})) < s$ and $\tau_{\sigma,t} = \tau_\sigma$. Note that g in monotonically increasing. As g is a K-recursive function and A is not low$_2$ there is an A-recursive function f such that $f(n) > g(g(n))$ for infinitely many n. Let $T = \lim \sigma_m$ be the ascending text which is obtained by starting with $\sigma_0 = 0$ and taking $\sigma_{m+1} = \sigma_m \tau_{\sigma_m, f(\max(\text{range}(\sigma_m)))}$. Now consider any n

with $f(n) > g(g(n))$. There is a first m such that $\max(\text{range}(\sigma_m)) \geq n$. It follows from the definition of g that $\max(\text{range}(\sigma_m)) \leq g(n)$, $g(\max(\text{range}(\sigma_m))) \leq f(n) \leq f(\max(\text{range}(\sigma_m)))$ and $\sigma_{m+1} = \sigma_m \tau_{\sigma_m}$. So, beyond σ_{m+1}, M does not output again any index below n while processing T. As there are infinitely such n it follows that M outputs on T every index only finitely often. Now T is an A-recursive ascending text, hence $B = \text{range}(T)$ is an A-recursive set which M does not partially learn. It follows that there is no reliable partial learner for the class of all A-recursive sets in this case. ∎

Corollary 22. *The class of all A-r.e. sets is reliably partially learnable from A-recursive texts iff $A \leq_T K$ and A is low$_2$.*

Proof. The class $\tilde{C}$ in Theorem 21 is reliably partially learnable only if $A \leq_T K$ and A is low$_2$. The texts considered there were ascending and, as the sets in $\tilde{C}$ are A-recursive, these texts are also A-recursive. It follows that the class of all A-r.e. sets is reliably partially learnable from A-recursive texts only if $A \leq_T K$ and A is low$_2$.

For the converse direction, note that the class of all A-recursive texts of some A-r.e. sets is uniformly K-recursive and that one can therefore reliably partially learn the underlying class of A-recursive functions. Identifying the index of the e-th text $\tilde{\varphi}_e^A$ and its range W_e^A permits then to convert the function learner into a language learner. ∎

One might ask whether Theorem 21 can be improved to use a characterisation based on partial learning in place of reliable partial learning. The answer is that one cannot do this. The next result exhibits that there are uncountably many oracles A such that the class of all A-recursive sets is partially learnable.

Theorem 23. *There are uncountably many oracles A such that the class of all A-recursive sets is partially learnable.*

Proof. For this result, one needs a co-r.e. tree T without deadends and with a branching node above every internal node such that the following holds for each each two different infinite branches A, B and all sets E:

- $A \not\leq_T B$;
- if A has hyperimmune-free Turing degree and $E \leq_T A$ and E is not recursive then $A \leq_T E$.

Groszek and Slaman [10] give a construction with the second property, but they do not write whether the first is satisfied as well. Furthermore, the construction of the minimal rK-degree by Raichev and Stephan [25] provides a co-r.e. tree T with the second property where the first one could easily be added in, one just has to add a fourth condition that whenever for d, e with $d < e$ and $\sigma, \tau \in \{0,1\}^e$ there are marker positions for the markers m_σ, m_τ from that construction there is an x and there are marker positions η above the current position of m_σ and η' above the current position of m_τ inside the current version of the co-r.e. tree considered with $\varphi_e^\eta(x)\downarrow \neq \eta'(x)\downarrow$ then one moves m_σ to η and m_τ to η' and cuts the laid off branches accordingly. The verification is left to the reader.

Now, given such a co-r.e. tree T, it follows from the Hyperimmune-free Basis Theorem of Jockusch and Soare [15] that T has uncountably many infinite branches A of hyperimmune-free Turing degree; fix such an A. Note that there is no other infinite branch B of T such that $B \leq_T A$. Now let f be a recursive one-one function with three inputs such that the following holds:

- if $j = 0$ then $W^A_{f(i,j,n)}$ is the recursive set $\{x : \forall y < x\, [\varphi_i(y)\downarrow \in \{0,1\}] \wedge \varphi_i(x)\downarrow = 1\}$;
- if $j > 0$ then $W^A_{f(i,j,n)}$ is the A-recursive set $\{x : \forall y < x\, [\varphi^A_i(y)\downarrow \in \{0,1\}] \wedge \varphi^A_i(x)\downarrow = 1\}$.

The learner will on a text for an A-recursive set R output exactly one triple $f(i,j,n)$ infinitely often with the following properties:

- $W^A_{f(i,j,n)} = R$;
- If R is recursive then $j = 0$;
- If R is nonrecursive then $j = k+1$ and $A = \varphi^R_k$ and φ^B_i, φ^B_k are total for all oracles B;
- The number n is the number of times some triple $\langle i', j', n'\rangle$ with $\langle i', j'\rangle < \langle i, j\rangle$ had been output.

In order to make this algorithm work one has to specify when a pair $\langle i, j\rangle$ qualifies such that it happens m times that, with current values of n, the index $f(i,j,n)$ is output; the algorithm is then to output $f(i,j,n)$ at stage s iff $\langle i, j\rangle$ qualifies at this step for another time and the pairs $\langle i', j'\rangle < \langle i, j\rangle$ have altogether qualfied n times up to stage s. For this, one defines that $\langle i, j\rangle$ qualifies at least m times iff the following conditions are true.

- $j = 0$ and $\varphi_i(x)$ is defined for $x = 0, 1, \ldots, m$ and there is a stage $s > m$ such that the set D of the first s elements in the text satisfies $D(x) = \varphi_i(x)$ for all $x \leq m$.
- $j = k+1$ and there are bounds s, t such that the set D of the first t elements from the input text satisfies the following conditions:
  - $\varphi^E_k(x)$ is defined for all oracles E and all $x \leq s$ within t steps of computation;
  - $\varphi^E_i(x)$ is defined for all oracles E and all $x \leq m$ within s steps of computation and $\varphi^E_i(x)$ does not query beyond bound s;
  - $\sigma = \varphi^D_k(0)\varphi^D_k(1)\ldots\varphi^D_k(s)$ is not enumerated into the complement of T within t steps;
  - $D(x) = \varphi^\sigma_i(x)$ for all $x \leq m$.

If R is recursive then the first but not the second condition can qualify infinitely often; the second would require that R computes an infinite branch of T which is impossible. If R is not recursive then the first condition cannot qualify infinitely often while the second qualifies infinitely often iff $j > 0$, φ_i and φ_k with $k = j-1$ are truth-table reductions, that is, Turing reductions defined for all oracles and $A = \varphi^R_k$ and $R = \varphi^A_i$. There is a least pair $\langle i, j\rangle$ which qualifies infinitely often and for this the parameter n converges to a value n which is the number of

times some pair $\langle i', j' \rangle$ qualifies. Then $f(i, j, n)$ is output infinitely often while all $f(i', j', n')$ with $\langle i', j' \rangle > \langle i, j \rangle$ are output only finitely often as the parameter n for those triples does not converge but goes to infinity. Hence exactly one index $f(i, j, n)$ is output infinitely often and that index satisfies that $R = W^A_{f(i,j,n)}$. ∎

If A is one of the oracles for which the class of all A-recursive sets is partially learnable then one has of course that this is also true for the closed sets of every Noetherian A-r.e. matroid. Hence one gets the following corollary which shows that reliably partially learnable cannot be replaced by partially learnable in Theorem 19.

Corollary 24. *There are uncountably many sets A such that C^A_Φ is partially learnable for every Noetherian matroid $(\mathbb{N}, \Phi)$.*

Acknowledgments. The authors would like to thank Matthew de Brecht, Noam Greenberg, Sanjay Jain and Dan Turetsky for discussions and comments.

References

1. Angluin, D.: Inductive inference of formal languages from positive data. Information and Control 45, 117–135 (1980)
2. de Brecht, M., Kobayashi, M., Tokunaga, H., Yamamoto, A.: Inferability of Closed Set Systems from Positive Data. In: Washio, T., Satoh, K., Takeda, H., Inokuchi, A. (eds.) JSAI 2006. LNCS (LNAI), vol. 4384, pp. 265–275. Springer, Heidelberg (2007)
3. de Brecht, M., Yamamoto, A.: Topological properties of concept spaces. Information annd Computation 208, 327–340 (2010)
4. Calude, C.S.: Relativized topological size of sets of partial recursive functions. Theoretical Computer Science 87, 347–352 (1991)
5. Calude, C.S.: Information and Randomness - An Algorithmic Perspective, 2nd edn. Springer, Heidelberg (2002)
6. Case, J., Jain, S.: Synthesizing learners tolerating computable noisy data. Journal of Computer and System Sciences 62, 413–441 (2001)
7. Case, J., Lynes, C.: Machine Inductive Inference and Language Identification. In: Nielsen, M., Schmidt, E.M. (eds.) ICALP 1982. LNCS, vol. 140, pp. 107–115. Springer, Heidelberg (1982)
8. Case, J., Smith, C.H.: Comparison of identification criteria for machine inductive inference. Theoretical Computer Science 25, 193–220 (1983)
9. Fulk, M.: Prudence and other conditions on formal language learning. Information and Computation 85, 1–11 (1990)
10. Groszek, M.J., Slaman, T.A.: Π^0_1 classes and minimal degrees. Annals of Pure and Applied Logic 87, 117–144 (1997)
11. Mark Gold, E.: Language identification in the limit. Information and Control 10, 447–474 (1967)
12. Harizanov, V.S., Stephan, F.: On the learnability of vector spaces. Journal of Computer and System Sciences 73, 109–122 (2007)
13. Jain, S., Osherson, D., Royer, J.S., Sharma, A.: Systems That Learn: An Introduction to Learning Theory, 2nd edn. MIT-Press, Boston (1999)

14. Jockusch, C.: Degrees in which recursive sets are uniformly recursive. Canadian Journal of Mathematics 24, 1092–1099 (1972)
15. Jockusch, C., Soare, R.: Π_1^0 classes and degrees of theories. Transactions of the American Mathematical Society 173, 33–56 (1972)
16. Kalantari, I., Retzlaff, A.: Maximal vector spaces under automorphisms of the lattice of recursively enumerable vector spaces. The Journal of Symbolic Logic 42, 481–491 (1977)
17. Kasprzik, A., Kötzing, T.: String Extension Learning Using Lattices. In: Dediu, A.-H., Fernau, H., Martín-Vide, C. (eds.) LATA 2010. LNCS, vol. 6031, pp. 380–391. Springer, Heidelberg (2010)
18. Metakides, G., Nerode, A.: Recursively enumerable vector spaces. Annals of Mathematical Logic 11, 147–171 (1977)
19. Metakides, G., Nerode, A.: Recursion theory on fields and abstract dependence. Journal of Algebra 65, 36–59 (1980)
20. Minicozzi, E.: Some natural properties of strong-identification in inductive inference. Theoretical Computer Science 2, 345–360 (1976)
21. Odifreddi, P.: Classical Recursion Theory. Studies in Logic and the Foundations of Mathematics, vol. 125. North-Holland, Amsterdam (1989)
22. Odifreddi, P.: Classical Recursion Theory II. Studies in Logic and the Foundations of Mathematics, vol. 143. Elsevier, Amsterdam (1999)
23. Osherson, D.N., Stob, M., Weinstein, S.: Systems That Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists. MIT Press (1986)
24. Osherson, D., Weinstein, S.: Criteria of language learning. Information and Control 52, 123–138 (1982)
25. Raichev, A., Stephan, F.: A minimal rK-degree. Computational Prospects of Infinity, Lecture Notes Series, Institute for Mathematical Sciences, National University of Singapore 15, 261–269 (2008)
26. Rogers, H.: Theory of Recursive Functions and Effective Computability. McGraw-Hill, New York (1967)
27. Soare, R.I.: Recursively enumerable sets and degrees. In: Perspectives in Mathematical Logic, Springer, Berlin (1987)
28. Stephan, F., Ventsov, Y.: Learning algebraic structures from text. Theoretical Computer Science 268, 221–273 (2001)

Invariance and Universality of Complexity

Helmut Jürgensen

Department of Computer Science
The University of Western Ontario
London, Ontario, Canada, N6A 5B7

Abstract. The definition of descriptional complexity or algorithmic information in the sense of Kolmogorov or Chaitin is based on two important properties of computable functions, the existence of universal machines and the invariance under the choice of machine. Recently, the notion of descriptional complexity for finite-state computable functions has been introduced by Calude et al. For the latter theory, one cannot rely on the existence of universal machines, but bases the conclusions on an invariance theorem for finite transducers.

This raises the question, which assumptions in algorithmic information theory are actually needed. We answer this question in a general setting, called encoded function space. Without any assumptions regarding encodings of functions and arguments and without any assumptions about computability or computing models, we introduce the notion of complexity. On this basis alone, a general invariance theorem is proved and sufficient conditions are stated for complexity to be computable. Next, universal functions are introduced, defined by pairing functions. It is shown that properties of the pairing functions, that is, of the joint encodings of functions and their inputs, determine the relation between the complexities measured according to different universal functions. In particular, without any other assumptions, for length-bounded or length-preserving pairing functions one can prove that complexity is independent of the choice of the universal function up to an additive constant. Some of the fundamental results of algorithmic information theory are obtained as corollaries.

1 Introduction

When discussing the *complexity* of some object, be it a computation or a problem or a string of symbols or a political transaction, we need to refer to a *framework* in which we express the task at hand. Thus, what we determine as being the complexity of the object, is really just the complexity of it as expressed within the given framework. This is a very simple idea. Rarely is it made explicit in computability theory or in complexity theory. In fact, simple statements like the following:

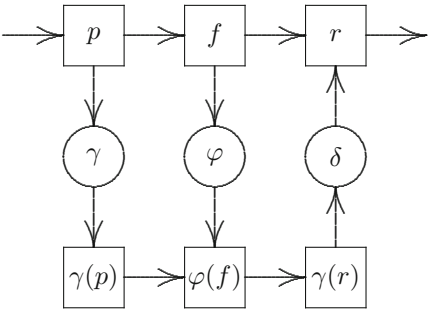
1. the function $f(n, m) = n \cdot m$ of integers is computable;
2. the multiplication of two binary n-bit positive integers can be performed in time $O(n^2)$;

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 140–158, 2012.

3. satisfiability is NP-complete;
4. the string $00\ldots00$ consisting of 32768 consecutive symbols 0 is compressible;

obscure the fact that we do not really express what we are talking about. In (1), we do not state what integers are and how to compute with them as integers. In (2), we implicitly talk about integers represented in binary, but do not say on which kind of computing device the computation is performed. In (3), we do not state which computing device is to be used, nor how the input is presented to the device, nor how the output is obtained from the device. In (4), we do not even give a yard stick by which to measure success.

The point is, of course, that computations work on representations of input objects, yield representations of output objects, and the computing devices themselves are described in a language the interpretation of which is obtained using a meta-language. With a large grain of salt, the situation is captured by the following diagram:



Here p is the problem at hand, f is the function to be applied and $r = f(p)$ is the result. One encodes the problem using γ; the function is defined as $\varphi(f)$ in a formal system according to φ; the encoded result $\gamma(r)$ is decoded by δ. While this diagram seems to capture the intuition well, it actually hides important facts: (1) encodings of problems, functions and results need not be unique; and (2) to describe encodings and decodings, again formal systems are needed which, themselves, rely on encodings and decodings.

These problems have been pointed out frequently, – the coding and decoding issues even lead to fundamental problems in the Gödelization of Turing machines as natural numbers – but, by their very nature, escape a formal treatment. Some aspects of these difficulties are addressed in the context of a general analysis of computability by Hoeberechts in her thesis [11].

In this paper, we try to clarify the rôle of various parameters which contribute to the problem. We pick up some discussions with Calude on the interpretation of Gödel's independence theorems since 1990 [2,5,6], certain of the author's criticism of the usual exposition of recursion theory, some later discussions with Calude and others regarding the foundations of computability and some ideas

presented by Calude et al. in recent papers on complexity with respect to finite automata [7,8,9]. Specifically we are looking at a problem which we consider central to all of computability and complexity theories, that of the effect the encoding has, not only on the results, but on the theories themselves. We can only scratch the surface of the problem, but hope to establish some pointers as to where to look for answers.

To model the general situation, we introduce the notion of an encoded function space. This formalizes one layer of the coding problem by referring, implicitly, to a common meta-language layer. We define the complexity of an object with respect to such a space. Neither the object nor the space need to be effectively given. Without any assumptions about computing models or computability, we prove a *General Invariance Theorem* of which the corresponding theorems for finite transducers and Turing machines are special cases.

The *Computability of Complexity Theorem* states sufficient conditions for complexity to be computable. These conditions are satisfied by the space of finite-state computable functions. The computability of finite-state descriptional complexity [7,8,9] is a corollary to this theorem.

Universal functions for an encoded function space are functions which 'simulate' all functions in the space. A universal function need not be in the space. We then define the complexity of objects with respect to universal functions, when they simulate only functions in the given space. The simulation is based on a pairing function, which can be defined equivalently on the functions and their arguments or on the encodings of the functions and the encodings of the arguments.

In general, there is no connection between the complexity c in the function space and the complexity C with respect to a universal function. Our general results indicate that properties of the pairing function would determine such a connection. We focus on two related natural cases: the pairing function is length-bounded or is length-preserving. Typical machine simulations use encodings of the machine to be simulated and its input which consist of the encoding of the machine, followed by a special symbol, followed by the encoded input. Such a pairing function is length-preserving. Thus, our considerations take into account the usual cases.

In some cases – for instance, when different alphabets are involved – the condition of length-preservation would need to be replaced by a similar, but much weaker condition. We do not know at present, how this would influence the rest of the theory. In encoded function spaces, this issue is addressed by the encodings of functions and arguments. As the complexity is a measure with respect to the encodings, the problem has been shifted to the encoding mappings.

The importance of properties of the pairing function for universality of finite-state functions was explored, albeit in a quite different context not related to complexity theory, in a book by Boucher [1] and in a thesis by Ring [16].

When the pairing function is length-bounded, the universal complexity C is a lower bound to the complexity c. When the pairing function is length-preserving, the two complexities are essentially equal up to an additive constant. This is

stated in the *Universal Complexity Theorem.* Moreover, the choice of the universal function makes no difference. Thus, when only universal functions defined by such pairing functions are considered, there is no need at all to introduce universal functions in the treatment of complexity.

In the case of Turing machines, the universal machines are in the class of machines under consideration. In the case of finite-state transducers, the universal machines are outside that class. In the latter case, the universal machines have no encodings and, within the class, a universal machine cannot simulate another universal machine. This suggests to consider a superspace of the encoded function space under consideration such that the universal functions are in the superspace. Under certain natural conditions, mainly length-boundedness of the pairing functions, the complexities defined by such universal functions are equal up to an additive constant. For the case of Turing-computable functions, one of the fundamental invariance results of algorithmic information theory is a corollary of this theorem.

We emphasize that none of the results of this paper requires computing models, computability or specific representations. Computability and related properties are not needed at this level. The crucial parameter is the pairing function. Its influence warrants further detailed study.

In some expositions on descriptional complexity, the formal definition is preceded by an informal very general motivation[1]. However, the actual formal definitions and results concerning invariance and universality rely on computing models, recursive functions and computability. As we show, these assumptions are not needed. Moreover, taking into account the ongoing philosophical discussions regarding the notion of computability, it seems preferable to develop as much of the theory as possible without such assumptions. We believe that, beyond the results of the present paper, many more of the fundamental properties of complexity can be proved in our general setting.

This paper is structured as follows. We review some basic notions and notation in Section 2. We assume the reader to be familiar with standard issues of the theories of computability, algorithmic information, automata and formal languages. As a standard references, we use [3,4,10,15,17]. In Section 3 we introduce encoded function spaces and the complexity of objects with respect to such spaces. A general invariance theorem of complexity in encoded function spaces is proved in Section 4. Moreover, we state natural sufficient conditions for complexity to be computable. The notion of universal function is explored in Section 5. For this purpose, pairing functions are introduced. For pairing functions satisfying a length-preservation condition, we prove a universal complexity theorem. In Section 6 we strengthen these results by introducing encodings of universal functions. As the properties of pairing functions turn out to be crucial, we discuss some variants of pairing functions and the consequences on the notion of universality in Section 7. A summary, some conclusions and some questions are presented in Section 8.

[1] See, for instance, Chapter 2.1 on the Invariance Theorem of the book [15] by Li and Vitányi or the paper [14] by Kolmogorov.

2 Notation

An *alphabet* is a finite non-empty set the elements of which are called *symbols*. To a avoid trivial exceptions, in this paper every alphabet is assumed to contain at least two symbols. Moreover, without special mention, the symbols a, b, 0 and 1 could be among these. When dealing with different alphabets we sometimes assume that they are subsets of a common, possibly infinite, but countable, set, of which only finite parts are used.

Let X be an alphabet. The set of *words* over X including the *empty word* ε is denoted by X^*. Define $X^+ = X^* \setminus \{\varepsilon\}$. For $w \in X^*$, $|\, w \,|$ is the *length* of w.

We use the common set notation. For singleton sets we often omit the parentheses when there is no risk of confusion. Let S and T be two sets. We write $f : S \overset{\circ}{\to} T$ to indicate that f is a partial mapping of S into T. Then $\mathrm{dom}(f)$ is the subset of S on which f is defined, the *domain* of f, and $\mathrm{codom}(f) = f(\mathrm{dom}(f))$ is the set of images of f, the *co-domain* of f. For $t \in T$, $f^{-1}(t)$ is the set of pre-images of t. When $S = \mathrm{dom}(f)$, f is a total mapping written as $f : S \to T$.

To avoid trivial case distinctions we define[2] $\min \emptyset$ and $\inf \emptyset$ to be ∞.

For sets A, B and C, a *pairing function* from $A \times B$ to C is an injective mapping, denoted by π, of $A \times B$ into C. Often it is convenient to write $\langle a, b \rangle$ instead of $\pi(a, b)$ with $a \in A$ and $b \in B$. Obviously, a pairing function exists if and only if $|\, A \,| \cdot |\, B \,| \leq |\, C \,|$. Given a pairing function, one defines the projections $p_A\big(\langle a, b \rangle\big) = a$ and $p_B\big(\langle a, b \rangle\big) = b$.

By $\mathbb{N}$ we denote the set of positive integers. Then $\mathbb{N}_0 = \mathbb{N} \cup 0$ is the set of non-negative integers. The symbol $\mathbb{R}$ denotes the set of real numbers. For real-valued partial functions f and g we write $f \lesssim g$ (or $f(x) \lesssim g(x)$) if $\mathrm{dom}(g) \subseteq \mathrm{dom}(f)$ and there is a non-negative constant c such that $f(x) \leq g(x) + c$ for all $x \in \mathrm{dom}(g)$. We write $f \sim g$ or $f(x) \sim g(x)$ if $f \lesssim g \lesssim f$, that is, $\mathrm{dom}(f) = \mathrm{dom}(g)$ and, for a constant $c \geq 0$ and all x, $|\, f(x) - g(x) \,| \leq c$.

We say that a function f is *Turing-computable,* if there is a Turing machine, which computes $f(x)$ for input x. A function f is said to be *finite-state computable,* if there is a finite transducer in the sense of [7,8,9], which outputs $f(x)$ for input x. Usually we do not explicitly mention the input and output alphabets of such computations. When necessary, it is assumed the all functions under consideration use the same alphabets, or the same countable sets of alphabets.

3 Function Spaces and Encodings

We introduce encoded function spaces and the complexity functions defined for them. Natural examples of such spaces are sets of functions computable with specific computing devices with encodings establishing the connection between the abstract functions and the devices computing them and between the input values and their representations. However, this intuition is a bit misleading –

[2] This convention is technically not correct because $\min \emptyset \notin \emptyset$. However it avoids case distinctions in the sequel. Using the infimum instead would obscure the fact that one deals with the minimum unless the set is empty.

albeit helpful – as we do not make any assumptions regarding computability, whichsoever.

Definition 1. *An* encoded function space *is a construct*

$$\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$$

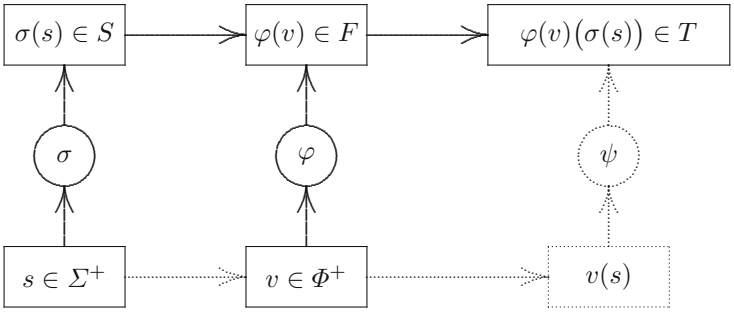
with the following properties.

1. *S is a non-empty countable set, the* source;
2. *T is a non-empty set, the* target;
3. *F is a non-empty countable set of partial mappings of S into T;*
4. *Φ and Σ are alphabets;*
5. *$\varphi : \Phi^+ \overset{\circ}{\to} F$ is a surjective partial mapping, the* encoding of *F.*
6. *$\sigma : \Sigma^+ \overset{\circ}{\to} S$ is a surjective partial mapping, the* encoding of *S;*

The encoded function space $\mathfrak{F}$ is said to be effective *if all items in the construct are effectively given and the mappings σ and φ are computable.*

Calling φ the encoding of F and σ the encoding of S suggests that φ would be a mapping of F into Φ^+ and that σ would be a mapping of S into Σ^+. However, functions in F and values in S could have multiple encodings. For this reason, the mappings φ and σ are defined in the opposite direction, that is, by abuse of language, in terms of decodings.

An encoded function space does not refer to any model of computation nor to any specific syntax for the encoding of arguments or functions. Moreover, for our purposes it is convenient not to consider encodings of results. On the other hand, when one has a specific model of computation in mind, one may consider $\mathrm{dom}(\varphi)$ as the set of syntactically correct descriptions of computers within the model which compute the functions in F; in this case, $\mathrm{dom}(\sigma)$ would be the set of syntactically correct descriptions of inputs to the these computers as shown in the following diagram:



Moreover, in such a context one may write $v(s)$ to denote the result of the computation defined by v when applied to the input described by s, written as a word over some alphabet Ψ. With a suitable decoding $\psi : \Psi^+ \xrightarrow{\circ} T$ one requires $\psi(v(s)) = \varphi(v)\big(\sigma(s)\big)$ as indicated in the diagram by dotted lines.

We now define the complexity of objects in the target set with respect to a given encoded function space $\mathfrak{F}$.

Definition 2. *Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ be an encoded function space and let $t \in T$.*

1. *For $f \in F$, the* f-complexity *of t in $\mathfrak{F}$ is defined as*

$$c_f^{\mathfrak{F}}(t) = \inf\big\{\,|\,s\,| \;\big|\; s \in \Sigma^+, f(\sigma(s)) = t\big\}$$

2. *The* complexity *of t in $\mathfrak{F}$ is defined as*

$$c^{\mathfrak{F}}(t) = \inf\big\{\,|\,v\,| + |\,s\,| \;\big|\; v \in \Phi^+, s \in \Sigma^+, \varphi(v)(\sigma(s)) = t\big\}.$$

In the definition of complexity, one can replace "inf" by "min" as only subsets of $\mathbb{N}_0$ are concerned using the convention of $\min \emptyset = \infty$ introduced above.

Our model of encoded function space is defined so as not to rely on computability, nor on specific representations of data or functions. As a consequence, the definition of complexity can be applied to situations when functions are not computable in the usual sense, to computations with real numbers, to computability with oracles and to many other potentially interesting scenarios. One needs descriptions of the functions and their arguments. These do not have to be constructive. They are just mappings. Of course, functions and arguments can be described in many different ways. For complexity, we look for shortest descriptions. The description of the results of functions applied to arguments is not important for this purpose.

To deal with specific types of functions or computability issues, constraints would be imposed on the components of an encoded function space. We provide a general example of such conditions, which then guarantee the computability of the complexity, below in Theorem 2.

4 Invariance

In [7,9] the authors provide a definition of complexity with respect to finite automata. In that case the functions are language mappings which can be defined by finite transducers. The source and target are sets of words, possibly encoded in binary. The transducers are described in some fixed way as binary words. If v is the description of a transducer then $\varphi(v)$ is the partial function computed by that transducer. The complexity of an output word t is defined as the minimum of all values $|\,s\,| + |\,v\,|$, such that the transducer described by v, when given s as input, computes t as output. One can also define the complexity of t with respect to a fixed transducer as the shortest input length for which

the given transducer computes t. This leads to the following result, referred to as *Invariance Theorem*[3]: For every transducer description v and every word t, the complexity of t does not exceed the sum of $|\, v \,|$ and the complexity of t with respect to the transducer described by v.

It turns out that this result does not depend on any properties of the set of functions under consideration. It is a consequence of the following general observation.

Fact 1. *Let* $P \subseteq \mathbb{R} \times \mathbb{R}$ *and let* $S_P = \{a + b \mid (a, b) \in P\}$. *For* $a \in \mathbb{R}$, *let* $p_a(P) = \{b \mid (a, b) \in P\}$. *Then*

$$\inf S_P \leq a + \inf p_a(P)$$

for all $a \in \mathbb{R}$ *with* $p_a(P) \neq \emptyset$.

Proof. If $P = \emptyset$, nothing needs to be proved. Otherwise, consider $a \in \mathbb{R}$ with $p_a(P) \neq \emptyset$. If $a + \inf p_a(P) < \inf S_P$ then there is $b \in p_a(P)$ such that $a + b < x + y$ for all $(x, y) \in P$. As $(a, b) \in P$, this is impossible. □

Theorem 1. (General Invariance Theorem)
Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ *be an encoded function space. Then, for all* $f \in F$ *and* $v \in \varphi^{-1}(f)$,

$$c^{\mathfrak{F}}(t) \leq c_f^{\mathfrak{F}}(t) + |\, v \,|$$

for all $t \in T$.

Proof. Let $t \in T$ and

$$P(t) = \left\{ \left(|\, v \,|, |\, s \,| \right) \;\middle|\; v \in \Phi^+, s \in \Sigma^+, \varphi(v)(\sigma(s)) = t \right\}.$$

Then

$$c^{\mathfrak{F}}(t) = \inf S_{P(t)}$$

and, for v with $\varphi(v) = f$,

$$c_f^{\mathfrak{F}}(t) = \inf p_{|\, v \,|}(P(t)).$$

By Fact 1 the inequality as claimed is obtained. □

Using Theorem 1, one can state very general sufficient conditions for the complexity to be computable[4].

[3] Theorem 13 of [7], Theorem 3.2 of [9].

[4] Here and in the sequel, by "effectively defined" or "effectively given" we mean that objects are obtained in some constructive way; this excludes, for example, obtaining such objects through non-constructive existential quantifiers. Decidability results are often not stated carefully enough. For example, the statement that "emptiness of a regular set is decidable" is true, if the set is effectively given *as a regular set,* but may be false otherwise. We do not assume any connection between the way such objects are effectively given and their encodings.

Theorem 2. (Computability of Complexity Theorem)
Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ be an encoded function space and let $t \in T$.

1. *For $f \in F$ the complexity $c_f^{\mathfrak{F}}(t)$ is computable, if the following conditions are satisfied:*
   (a) *f is effectively defined and computable.*
   (b) *$\mathrm{dom}(f)$ and $\mathrm{codom}(f)$ are decidable.*
   (c) *S is effectively defined and σ is computable.*
   (d) *t is effectively defined and equality is decidable in T.*
2. *The complexity $c^{\mathfrak{F}}(t)$ is computable, if, in addition to Conditions a–d, also the following conditions are satisfied:*
   (e) *Emptiness of the set $\{f \mid f \in F, t \in \mathrm{codom}(f)\}$ is decidable.*
   (f) *F is effectively given and enumerable, and φ is computable.*
   (g) *For every $c \in \mathbb{N}$, emptiness of the set*

$$F_{c,t} = \{f \mid f \in F, \exists s \in \Sigma^+ : |\, s \,| < c, f(\sigma(s)) = t\}$$

   is decidable.

Proof. First check if $t \in \mathrm{codom}(f)$. If not, $c_f^{\mathfrak{F}}(t) = \infty$. Otherwise, one computes as follows: Enumerate Σ^+ in pseudo-alphabetic order[5] until the first s is found with $f(s) = t$. Then $c_f^{\mathfrak{F}} = |\, s \,|$. This proves the first statement.

For the second statement, first check whether $t \in \mathrm{codom}(f)$ for some $f \in F$. If not, then $c^{\mathfrak{F}}(t) = \infty$. Otherwise, let $c = \infty$ and enumerate Φ^+ until the first v with $c_{\varphi(v)}^{\mathfrak{F}}(t) < c$ is found. Let $c = |\, v \,| + c_{\varphi(v)}^{\mathfrak{F}}(t)$. If $F_{c,t} = \emptyset$, then $c^{\mathfrak{F}}(t) = c$. Otherwise, continue the enumeration as above. By Theorem 1, this procedure computes $c^{\mathfrak{F}}(t)$. □

The case of infinite complexity is usually ruled out immediately by the assumptions. When one considers functions defined by automata or other kinds of formal systems, also the conditions a, c, d, f are trivially satisfied. Only the conditions b and g are crucial. Neither condition is satisfied when F is the set of all Turing computable functions. On the other hand, both are satisfied, when F is the set of finite-state computable functions.

Corollary 1. ([7,9]) *If $\mathfrak{F}$ is the space of finite-state computable functions with transducers as the computer model then $c^{\mathfrak{F}}$ is computable.*

5 Universality

The complexity of a value t with respect to an encoded function space $\mathfrak{F}$ is not uniformly defined as it involves not only the "program" for t but also the description of the function by which to compute t. This is in contrast to algorithmic

[5] There are various names for this ordering of words. First enumerate words by length; then, for any given length, enumerate them lexicographically. This avoids entering infinite branches of a search tree, while other options still need to be explored. Any other order with similar properties could be used as well.

information or descriptional complexity where a specific f_U can be chosen for all computations without distorting the complexity. When F is the set of all partial Turing-computable functions, one chooses f_U as the function computed by some universal Turing machine. By a very simple argument, present already in Kolmogorov's work [14], one shows that the complexity with respect to f_U is a lower bound to the complexity with respect to every Turing-computable function f up to an additive constant depending on f. In the sequel we explore the general structure of this argument.

Definition 3. *Let S and T be non-empty sets and let F be a non-empty set of partial mappings of S into T. Let $\pi : F \times S \to S$ be a pairing function.*

A partial function $g : S \overset{\circ}{\to} T$ is said to universal for F by π, *if $\langle f, s\rangle \in \mathrm{dom}(g)$ and $g(\langle f, s\rangle) = f(s)$ for all $f \in F$ and all $s \in \mathrm{dom}(f)$.*

Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ be an encoded function space. A partial function $g : S \overset{\circ}{\to} T$ is said to universal for $\mathfrak{F}$ by π, *if it is universal for F by π.*

The notion of a universal function as defined in [15] should not be confused with our definition. Those definitions capture different ideas.

A function which is universal for F by π need not be an element of F. One has the following well-known property of Turing-computable functions:

Fact 2. *There are universal Turing-computable functions.*

Of course, the choice of the universal function depends on the pairing function π, that is, on the joint encoding $\langle f, s\rangle$ of the function and its argument. As shown by Boucher [1] and Ring [16], there are some subtle problems arising from the choice of the pairing function, by which such a seemingly convincing definition of universality can be unacceptable in certain cases. We discuss this issue further below in Section 7.

Fact 3. *There is no finite-state computable function which is universal for all finite-state computable functions.*

In contrast to Fact 3 there are deterministic two-way pushdown automata which are universal for the set of all finite-state computable functions [16]. We explain some of the details of this statement in Section 7 below.

Fact 4. *Let S and T be non-empty sets, let F be a non-empty set of mappings of S into T, and let π be a pairing function of $F \times S$ into S. If g is universal for F by π, then g is uniquely defined by*

$$g(s) = p_F(s)\big(p_S(s)\big)$$

for all $s \in \mathrm{codom}(\pi)$ with $p_S(s) \in \mathrm{dom}(p_F)(s)$. For all other values of s, g can be left undefined or defined in an arbitrary way. In particular, a universal function for F exists, if and only if there is a pairing function π.

We extend the definition of complexity as follows:

Definition 4. *Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ be an encoded function space, let $\pi : F \times S \to S$ be a pairing function, let $g : S \overset{\circ}{\to} T$ be universal for $\mathfrak{F}$ by π, and let $t \in T$. The* complexity of t in $\mathfrak{F}$ according to g (or π) *is defined as*

$$C_g^{\mathfrak{F}}(t) = \min\left\{ |\, s \,| \;\middle|\; \begin{array}{l} s \in \Sigma^+, \exists f \in F, \exists s' \in S : \\ \sigma(s) = \langle f, s' \rangle, s' \in \mathrm{dom}(f), f(s') = t \end{array} \right\}.$$

In this definition of complexity the "size" of the encoding of the function f and its arguments enters through a backdoor, which is hidden in the usual definitions. In considering complexity with respect to a universal function g, it is crucial that g "simulates" encoded functions on encoded inputs using fixed alphabets. If the encoding of objects expressed over different alphabets is assumed to be part of the simulation, certain functions may fail to be universal (see [1]) or the complexity measure may fail to be invariant under the choice of universal functions. We give some additional explanations in Section 7 below.

Using the framework of Definition 4, there could be $s \in \Sigma^+$ such that $g\big(\sigma(s)\big) = t$ and $|\, s \,| < C_g^{\mathfrak{F}}(t)$. In this case $\sigma(s) \notin \mathrm{codom}(\pi)$ or $\sigma(s) = \langle f, s' \rangle$ for some $f \in F$ and $s' \in S$ such that $s' \notin \mathrm{dom}(f)$. Hence, one has $C_g^{\mathfrak{F}}(t) \leq |\, s \,|$ for all $s \in \Sigma^+$ with $g\big(\sigma(s)\big) = t$ satisfying the following condition:

$$\sigma(s) \in \mathrm{codom}(\pi) \wedge p_S\big(\sigma(s)\big) \in \mathrm{dom}\big(p_F(\sigma(s))\big).$$

To compare $C_g^{\mathfrak{F}}$ and $c^{\mathfrak{F}}$ we need a connection between the lengths of encodings of pairs $\langle f, s' \rangle$ and the sum of the lengths of encodings of f and s'. To establish such a connection is the next goal.

Fact 5. *Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ be an encoded function space and let $\hat{\pi} : \Phi^+ \times \Sigma^+ \overset{\circ}{\to} \Sigma^+$ be a pairing function. There is a unique pairing function $\pi : F \times S \to S$ such that the following diagram commutes:*

$$\begin{array}{ccccc} \Phi^+ & \times & \Sigma^+ & \xrightarrow{\hat{\pi}} & \Sigma^+ \\ \Big\downarrow{\scriptstyle \varphi} & & \Big\downarrow{\scriptstyle \sigma} & & \Big\downarrow{\scriptstyle \sigma} \\ F & \times & S & \xrightarrow{\pi} & S \end{array}$$

One has $\pi(f, s) = \sigma(\hat{\pi}(v, s'))$ with $f \in F$, $s \in S$, $\varphi(v) = f$ and $\sigma(s') = s$. We say that π is derived from $\hat{\pi}$.

Combining Facts 4 and 5, one finds that pairing functions on encodings define universal functions uniquely up to non-essential values. This proves the following statement.

Theorem 3. *Let* $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ *be an encoded function space, let* $\hat{\pi} : \Phi^+ \times \Sigma^+ \to \Sigma^+$ *be a pairing function, and let* $\pi : F \times S \to S$ *be the pairing function derived from* $\hat{\pi}$*. Let* g_π *be a universal function for* $\mathfrak{F}$ *by* π*. Then*

$$C^{\mathfrak{F}}_{g_\pi}(t) = \min\Big\{ \big|\, \hat{\pi}(v,s) \,\big| \;\Big|\; v \in \Phi^+, s \in \Sigma^+, \varphi(v)\big(\sigma(s)\big) = t \Big\}$$

for all $t \in T$*.*

A connection between the lengths of encodings of pairs $\langle f, s' \rangle$ and the sum of the lengths of encodings of f and s' as required to compare $C^{\mathfrak{F}}_g$ and $c^{\mathfrak{F}}$ is established in the following definition. Obviously other options are possible; this one is chosen to arrive at analogues of the usual universality results of algorithmic information theory.

Definition 5. *An injective partial function* $\hat{\pi} : \Phi^+ \times \Sigma^+ \overset{\circ}{\to} \Sigma^+$ *is* length-bounded *if*

$$|\, \hat{\pi}(u,v) \,| \lesssim |\, u \,| + |\, v \,|$$

for all $(u,v) \in \mathrm{dom}(\hat{\pi})$*. It is said to be* length-preserving *if*

$$|\, \hat{\pi}(u,v) \,| \sim |\, u \,| + |\, v \,|$$

for all $(u,v) \in \mathrm{dom}(\hat{\pi})$*.*

The typical encodings of functions and their arguments used in the literature are length-preserving. For our present purposes this condition is sufficient. As shown in [16], this condition is not sufficient to handle more subtle aspects of universality. We recount some of the details further below in Section 7.

Theorem 4. (Universal Complexity Theorem)
Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ *be an encoded function space. Let* $\hat{\pi} : \Phi^+ \times \Sigma^+ \to \Sigma^+$ *be a length-bounded pairing function and let* $\pi : F \times S \to S$ *be derived from* $\hat{\pi}$*. Let* g_π *be a universal function for* $\mathfrak{F}$ *defined by* π*. Then* $C^{\mathfrak{F}}_{g_\pi} \lesssim c^{\mathfrak{F}}$*. Moreover,* $C^{\mathfrak{F}}_{g_\pi} \sim c^{\mathfrak{F}}$*, if* $\hat{\pi}$ *is length-preserving.*

Proof. By Theorem 3, $C^{\mathfrak{F}}_{g_\pi}(t)$ is the minimum of

$$\big|\, \hat{\pi}(v,s) \,\big|$$

with $v \in \Phi^+$, $s \in \Sigma^+$ and $\varphi(v)\big(\sigma(s)\big) = t$. The fact that $\hat{\pi}$ is length-bounded implies that there is a constant c such that

$$\big|\, \hat{\pi}(v,s) \,\big| \leq |\, v \,| + |\, s \,| + c.$$

The minimum of $|\, v \,| + |\, s \,|$ is $c^{\mathfrak{F}}(t)$. This proves the first claim. When $\hat{\pi}$ is length-preserving, then also

$$|\, v \,| + |\, s \,| - c \leq \big|\, \hat{\pi}(v,s) \,\big| \leq |\, v \,| + |\, s \,| + c,$$

which implies the second statement. □

As a consequence of Theorem 4, when only universal functions defined by length-preserving pairing functions are considered, complexity can be defined without recourse to universal functions. This is important for the interpretation of the results in [7,8,9].

The complexity with respect to different universal functions need not be related at all. In the completely general setting, even length-boundedness of the respective pairing functions seems not to permit a comparison. On the other hand, it is not unexpected, that length-preservation allows for a comparison. Note that most encodings considered in the literature are not just length-bounded, but even length-preserving.

Theorem 5. *Let $\mathfrak{F} = (F, S, T, \Phi, \Sigma, \varphi, \sigma)$ be an encoded function space. For $i = 1, 2$, let $\hat{\pi}_i : \Phi^+ \times \Sigma^+ \to \Sigma^+$ be length-preserving pairing functions and let $\pi_i : F \times S \to S$ be derived from $\hat{\pi}_i$. Let g_{π_i} be a universal function for $\mathfrak{F}$ defined by π_i. Then $C^{\mathfrak{F}}_{g_{\pi_1}} \sim C^{\mathfrak{F}}_{g_{\pi_2}}$.*

Proof. There are constants c_i such that

$$| v | + | s | - c_i \leq \left| \hat{\pi}_i(v, s) \right| \leq | v | + | s | + c_i$$

for all $v \in \Phi^+$ and $s \in \Sigma^+$. Thus $| \hat{\pi}_1(v, s) | \sim | \hat{\pi}_2(v, s) |$. □

Theorem 5 can also be derived as a corollary of Theorem 4. Note that none of the results so far require that the universal function be in the space $\mathfrak{F}$. The results follow solely from properties of the pairing functions involved.

6 Universal Functions Encoded

So far, when considering universal functions for an encoded function space we did not refer to any representation of these. We now require, as an additional assumption, that the universal functions are, themselves, elements of an encoded function space. This permits us to compare the complexities with respect to different universal functions directly.

Definition 6. *For $i = 1, 2$ let $\mathfrak{F}_i = (F_i, S, T, \Phi, \Sigma, \varphi_i, \sigma_i)$ be two encoded function spaces.*

1. *We say that $\mathfrak{F}_1$ is a* subspace *of $\mathfrak{F}_2$, written as $\mathfrak{F}_1 \subseteq \mathfrak{F}_2$, if $F_1 \subseteq F_2$, $\varphi_1 \subseteq \varphi_2$ and $\sigma_1 \subseteq \sigma_2$.*
2. *We say that $\mathfrak{F}_1$ is a* conservative *subspace of $\mathfrak{F}_2$, if $\mathfrak{F}_1 \subseteq \mathfrak{F}_2$ and additionally,*

$$\varphi_1^{-1}(f) = \varphi_2^{-1}(f) \text{ and } \sigma_1^{-1}(s) = \sigma_2^{-1}(s)$$

for all $f \in F_1$ and all $s \in S_1$.

We now consider the following situation: The function space of interest is the space $\mathfrak{F}_1$. The universal functions for $\mathfrak{F}_1$ to be considered are in $\mathfrak{F}_2$, and $\mathfrak{F}_1$ is a conservative subspace of $\mathfrak{F}_2$. Thus, the universal functions have encodings in $\mathfrak{F}_2$,

and the functions of $\mathfrak{F}_1$ and their arguments have exactly the same encodings in both spaces.

Special situations include the following: (1) $\mathfrak{F}_1 = \mathfrak{F}_2$, as is the case for Turing-computable functions; (2) for $\mathfrak{F}_1$ being the finite-state computable functions, $\mathfrak{F}_2$ could be the space of functions computable by deterministic two-way pushdown automata (see [16]) or the space of functions computable by deterministic linearly bounded Turing machines (see [1]).

Theorem 6. *Let $\mathfrak{F}_1$ and $\mathfrak{F}_2$ be encoded function spaces such that $\mathfrak{F}_1$ is a conservative subspace of $\mathfrak{F}_2$. For $i = 1, 2$, let $\hat{\pi}_i : \Phi^+ \times \Sigma^+ \to \Sigma^+$ be length-bounded pairing functions, and let $\pi_i : F_2 \times S \to S$ be derived from $\hat{\pi}_i$. Let g_{π_i} be universal for $\mathfrak{F}_2$ by π_i. If $g_{\pi_1}, g_{\pi_2} \in F_2$, then $C^{\mathfrak{F}_1}_{g_{\pi_1}} \sim C^{\mathfrak{F}_1}_{g_{\pi_2}}$.*

Proof. For $i = 1, 2$, $C^{\mathfrak{F}_1}_{g_{\pi_i}}(t)$ is the minimum of

$$\left| \hat{\pi}_i(v, s) \right|$$

with $v \in \Phi^+$, $s \in \Sigma^+$, $\varphi_2(v)\,(\sigma_2(s)) = t$, $\varphi_2(v) \in F_1$ and $\sigma_2(s) \in S_1$. As $\mathfrak{F}_1$ is a conservative subspace of $\mathfrak{F}_2$, $C^{\mathfrak{F}_1}_{g_{\pi_i}}(t)$ is also the minimum of

$$\left| \hat{\pi}_i(v, s) \right|$$

with $v \in \Phi^+$, $s \in \Sigma^+$, $\varphi_2(v)\,(\sigma_2(s)) = t$, $\varphi_1(v) \in F_1$ and $\sigma_1(s) \in S_1$.

Let $u_i \in \varphi_2^{-1}(g_{\pi_i})$. Then

$$g_1\Big(\sigma_2\big(\hat{\pi}_1(u_2, \hat{\pi}_2(v, s))\big)\Big) = t = g_2\Big(\sigma_2\big(\hat{\pi}_2(u_1, \hat{\pi}_1(v, s))\big)\Big)$$

for v and s as above.

As $\hat{\pi}_1$ and $\hat{\pi}_2$ are length-bounded, there are constants c_1 and c_2 greater than 0 such that

$$\left| \hat{\pi}_1(u_2, \hat{\pi}_2(v, s)) \right| \leq |\, u_2 \,| + |\, \hat{\pi}_2(v, s) \,| + c_1 \leq |\, u_2 \,| + |\, v \,| + |\, s \,| + c_1 + c_2$$

and

$$\left| \hat{\pi}_2(u_1, \hat{\pi}_1(v, s)) \right| \leq |\, u_1 \,| + |\, \hat{\pi}_1(v, s) \,| + c_2 \leq |\, u_1 \,| + |\, v \,| + |\, s \,| + c_1 + c_2.$$

Hence

$$C^{\mathfrak{F}_1}_{g_{\pi_1}}(t) \leq \left| \hat{\pi}_1(u_2, \hat{\pi}_2(v, s)) \right| \leq |\, u_2 \,| + C^{\mathfrak{F}_1}_{g_{\pi_2}}(t) + c_1$$

and

$$C^{\mathfrak{F}_1}_{g_{\pi_2}}(t) \leq \left| \hat{\pi}_2(u_1, \hat{\pi}_1(v, s)) \right| \leq |\, u_1 \,| + C^{\mathfrak{F}_1}_{g_{\pi_1}}(t) + c_2$$

for v and s achieving the minima. Let $c \geq \max\{c_1, c_2\} + \max\{|\, u_1 \,|, |\, u_2 \,|\}$. then

$$\left| C^{\mathfrak{F}_1}_{g_{\pi_1}}(t) - C^{\mathfrak{F}_1}_{g_{\pi_2}}(t) \right| \leq c$$

as was to be proved. □

The proof of Theorem 6 uses the idea underlying Kolmogorov's proof for Turing-computable functions [14]. This idea is also used in most of the literature for the universality theorem.

7 Pairing Functions

In this section we summarize some of the results and observations made by Ring in [16] and by Boucher in [1] concerning the notion of universal function for automata and the rôle of the pairing functions in the definition of this notion. Related considerations can also be found in [13].

In [16], universality is defined with respect to language acceptance rather than output computation. This can be modified easily[6]. To present the concepts introduced in [1,16], we use the notation of the present paper with some liberty. Three concepts of universal function (universal automaton) are distinguished as follows:

1. *Weak universality:* g is weakly universal for $\mathfrak{F}$, if there is an injective function
$$\psi : \big\{(v,s) \;\big|\; v \in \mathrm{dom}(\varphi), s \in \mathrm{dom}(\sigma)\big\} \to \mathrm{dom}(g)$$
such that
$$\varphi(v)\big(\sigma(s)\big) = g\big(\psi(v,s)\big)$$
for all v and s.
2. *Universality:* g is universal for $\mathfrak{F}$, if there are injective partial functions
$$\chi : \Phi^+ \overset{\circ}{\to} \Sigma^+ \text{ and } \psi : \Sigma^+ \overset{\circ}{\to} \Sigma^+$$
such that
$$\mathrm{dom}(\chi) = \mathrm{dom}(\varphi),\ \mathrm{dom}(\psi) = \mathrm{dom}(\sigma)$$
and
$$\varphi(v)\big(\sigma(s)\big) = g\big(\chi(v)\psi(s)\big)$$
for all v and s.
3. *Strong Universality:* g is strongly universal, if there are injective partial functions
$$\chi : \Phi^+ \overset{\circ}{\to} \Sigma^+ \text{ and } \psi : \Sigma^+ \overset{\circ}{\to} \Sigma^+$$
such that
$$\mathrm{dom}(\chi) = \mathrm{dom}(\varphi),\ \mathrm{dom}(\psi) = \mathrm{dom}(\sigma),$$
ψ is a homomorphism and
$$\varphi(v)\big(\sigma(s)\big) = g\big(\chi(v)\psi(s)\big)$$
for all v and s.

[6] See Section 3.5 of [16].

Above we assume that the factorization of the concatenation $\chi(v)\psi(s)$ into $\chi(v)$ and $\psi(s)$ is unique.

In [1], weak universality is called *universalité au sens large* and universality is called *universalité au sens strict;* strong universality is not considered there.

In the context of [16] and for the specific goals of that work, the items under consideration have to satisfy strong computability conditions: $\mathfrak{F}$ must be effective; the mappings ψ and χ must be computable, etc. We summarize several results of [16] omitting some of the more technical assumptions. For precise statements, the reader needs to consult the original.

Theorem 7. ([16])

1. *For every effectively defined set of automata*[7] *recognizing only decidable languages there is a weakly universal finite automaton.*
2. *There is no universal finite automaton for the deterministic finite automata with a fixed non-empty alphabet.*
3. *For every effectively defined set of automata recognizing only decidable languages there is a universal pushdown automaton.*
4. *For every effectively defined set of finite automata there is a strongly universal deterministic two-way pushdown automaton.*
5. *There is no strongly universal one-way push-down automaton for the deterministic finite automata with a fixed non-empty alphabet.*

The cases of weak universality and universality correspond to using arbitrary pairing functions $\hat{\pi}$. With length-bounded or length-preserving pairing functions one encounters situations allowing for strong universality.

Theorem 7 indicates that the three notions of universality are indeed different. For weak universality and universality, the pairing functions – encodings of functions and inputs – permit one to hide much of the computation, even the results, in the input to the (weakly) universal function. Thus, these notions are not satisfactory when one considers the 'computation power' or the 'complexity of computations' (even with oracles) of the function space at hand. The problem is especially evident in the statements (1) and (3) of Theorem 7. For descriptional complexity this seems not to be such a serious issue, as explicitly encoding the result in the input is likely only to lengthen the input.

Typical pairing functions employed to describe the input to a universal machine for the simulation of the computation of a machine v on input s have the form vms or smv where m is a special marker symbol or marker sequence in the sense of [12]. With $\hat{\pi}(v,s) = vms$ one has $|\,\hat{\pi}(v,s)\,| = |\,v\,| + |\,s\,| + |\,m\,|$ where $|\,m\,|$ is a constant. Thus, $\hat{\pi}$ is length-preserving. This type of pairing function can be used in the proof of Theorem 7(4). We state this result using the present notation.

[7] Here and in (3) below, arbitrary types of automata are considered, not just finite automata. In our terminology this would be an effective encoded function space with total computable functions.

Corollary 2. *Let $\mathfrak{F}$ be an encoded function space of finite-state computable functions. There is a length-preserving pairing function $\hat{\pi}$ such that the universal function g_π for $\mathfrak{F}$ can be computed by a deterministic two-way pushdown automaton. Up to an additive constant, the complexity with respect to g_π is independent of the choice of π and, hence, of the corresponding deterministic two-way pushdown automaton.*

Proof. The claim follows from Theorem 7(4) and Theorem 5. □

Theorem 6 is not directly applicable in the case of Corollary 2. The simulation of finite automata by a deterministic two-way pushdown automaton used in [16] works as follows: The input of the machine to be simulated is copied onto the pushdown tape. Then the current state is written to the pushdown tape. Now the transition is looked up on the input tape using the state information and the next input symbol from the pushdown tape. The input symbol is erased from the pushdown tape and the next state is written on the pushdown tape. This simulation works when the machine to be simulated is a finite automaton, but will not work when that machine itself is a pushdown automaton, as would be required for Theorem 6. We do not know, whether there are deterministic two-way pushdown automata which are universal for their class, but doubt it.

A weaker simulation result for finite automata is due to Boucher [1].

Theorem 8. ([1]) *There is a deterministic linearly bounded Turing machine which is universal for the space of finite automata.*

Both in [1] and in [16], infinite classes of automata with different alphabets are considered. Thus the encodings of automata and their inputs have to map these to one or two common alphabets, usually binary. In our concept of encoded function space, we have hidden this issue by relying on the two fixed alphabets Φ and Σ. However, conflicting measurements of space or time or information may result when more, possibly infinitely many, alphabets are involved[8]. Usually, one circumvents this problem, by taking all measurements with respect to the encoded objects.

The results of [1] include the following:

Theorem 9. ([1])

1. *There is a deterministic linearly bounded Turing machine, which is weakly universal for the class of all deterministic linearly bounded Turing machines.*
2. *There is no deterministic linearly bounded Turing machine, which is universal for the class of all deterministic linearly bounded Turing machines.*

The second statement of Theorem 9 is a consequence of the need to simulate deterministic linearly bounded Turing machines with arbitrarily large alphabets. Then the linear bound on the space may be violated. The problem is no longer

[8] Sometimes, but not always, the difference is just a constant factor, for instance, the logarithm of the size of an alphabet; there are situations when even this seemingly trivial issue is crucial.

present when the alphabet is fixed. In that case, deterministic linearly bounded Turing machines satisfy the assumptions of Theorem 6. Thus, any two such machines which are universal for the space of finite-state functions give rise to the same complexity up to an additive constant.

8 Summary, Some Conclusions, Some Questions

We have isolated the rôles of the various assumptions in the universality and invariance results of algorithmic information theory. Computability and computing models are not needed in general unless one aims for statements about the computability of complexity-related properties. A general invariance theorem for complexity can be proved without any specific assumptions.

Rather weak sufficient conditions are derived for complexity to be computable. These conditions can be relativized easily to include oracles.

It turns out that the properties of pairing functions influence what can be said about complexity with respect to universal functions. Length-boundedness or length-preservation entail the invariance of complexity with respect to universal functions up to additive constants. This holds true even when the universal functions are outside the class of functions under consideration.

The fact that length-boundedness and length-preservation are important is a direct consequence of the logarithmic nature of complexity or information measures.

In summary: To arrive at some of the very fundamental statements about descriptional complexity, computing models are not needed at all. Only two general concepts are needed: encoded function spaces and pairing functions.

These results give rise to several questions, for example: (1) Which of the other fundamental results of algorithmic information theory can be proved with such a reduced set of assumptions? (2) Are encoded function spaces $\mathfrak{F}_1$ and $\mathfrak{F}_2$ satisfying the conditions of Theorem 6 rare? (3) Which properties of pairing functions are required to guarantee the invariance under the choice of the universal function?

We emphasize that none of the results of this paper requires computing models, computability or specific representations. The crucial parameter is the pairing function. In a way, it represents a specific view of the space under consideration. Its influence warrants further detailed study.

References

1. Boucher, C.: Leçons sur la théorie des automates mathématiques. Lecture Notes in Operations Research and Mathematical Systems, vol. 46. Springer, Berlin (1971)
2. Calude, C.S., Jürgensen, H., Zimand, M.: Is independence an exception? Applied Mathematics and Computation 66, 63–76 (1994)
3. Calude, C.S.: Theories of Computational Complexities. North-Holland, Amsterdam (1988)
4. Calude, C.S.: Information and Randomness – An Algorithmic Perspective, 2nd edn. Springer, Berlin (2002)

5. Calude, C.S., Jürgensen, H.: Randomness as an invariant for number representations. In: Maurer, H., Karhumäki, J., Rozenberg, G. (eds.) Results and Trends in Theoretical Computer Science. LNCS, vol. 812, pp. 44–66. Springer, Berlin (1994)
6. Calude, C.S., Jürgensen, H.: Is complexity a source of incompleteness? Advances in Appl. Math. 35, 1–15 (2005)
7. Calude, C.S., Salomaa, K., Roblot, T.K.: Finite-state complexity and randomness. Research Report CDMTCS-374, Centre for Discrete Mathematics and Theoretical Computer Science, Auckland, New Zealand (June 2010)
8. Calude, C.S., Salomaa, K., Roblot, T.K.: Finite-state complexity and the size of transducers. In: McQuillan, I., Pighizzini, G. (eds.) Proceedings of the Twelfth Annual Workshop on Descriptional Complexity of Formal Systems, Saskatoon, Canada, August 8–10, Technical Report 2010-02, 50–61, Department of Computer Science, University of Saskatchewan, Saskatoon (2010); Also published as Electronic Proceedings in Theoretical Computer Science (EPTCS) 31, 38–47 (2010), doi:10.4204/EPTCS.31
9. Calude, C.S., Salomaa, K., Roblot, T.K.: Finite-state complexity. Theoretical Comput. Sci. 412, 5668–5677 (2011)
10. Chaitin, G.J.: Algorithmic Information Theory, 3rd edn. Cambridge Tracts in Theoretical Computer Science, vol. 1. Cambridge University Press, Cambridge (1987/1990)
11. Hoeberechts, M.: On the Foundations of Computability Theory. PhD Thesis, The University of Western Ontario (2009)
12. Jürgensen, H.: Marks of change in sequences. In: Proceedings of the 37th International Conference on Applications of Mathematics in Engineering and Economics, AMEE 2011, Sozopol, Bulgaria, June 8–13 (to appear, 2011)
13. Kaufholz, G.: Der programmierbare endliche Automat. Acta Inform. 1, 225–241 (1971/1972)
14. Kolmogorov, A.N.: Три подхода к определению понятия «количество информации» Three approaches for defining the concept of 'information quantity'. Problemy Peredachi Informatsii 1(1), 3–11 (1965); English translation: Three approaches to the quantitative definition of information. Problems Inform. Transmission 1, 1–7 (1966); Also published as: Three approaches to the definition of the notion of amount of information. In: [18], pp. 184–193
15. Li, M., Vitányi, P.: An Introduction to Kolmogorov Complexity and Its Applications, 2nd edn. Springer, Berlin (1997)
16. Ring, H.: Universelle erkennende Automaten für reguläre Sprachen. Dissertation, Universität Karlsruhe (TH) (1973)
17. Rozenberg, G., Salomaa, A. (eds.): Handbook of Formal Languages. Springer, Berlin (1997)
18. Shiryayev, A.N. (ed.): Selected Works of A. N. Kolmogorov, III. Kluwer Academic Publishers, Dordrecht (1993); Annotated translation of Теория информации и теория алгоритмов, Nauka, Moscow (1987); translated by A.B. Sossinsky

Demuth's Path to Randomness*

Antonín Kučera[1] and André Nies[2]

[1] Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic
antonin.kucera@mff.cuni.cz
[2] Department of Computer Science, University of Auckland, Auckland, New Zealand
andre@cs.auckland.ac.nz

Abstract. Osvald Demuth (1936–1988) studied constructive analysis in the Russian style. For this he introduced notions of effective null sets which, when phrased in classical language, yield major algorithmic randomness notions. He proved several results connecting constructive analysis and randomness that were rediscovered only much later.

We give an overview in mostly chronological order. We sketch a proof that Demuth's notion of Denjoy sets (or reals) coincides with computable randomness. We show that he worked with a test notion that is equivalent to Schnorr tests relative to the halting problem. We also discuss the invention of Demuth randomness, and Demuth's and Kučera's work on semigenericity.

1 Who Was Demuth?

The mathematician Osvald Demuth worked mainly on constructive analysis in the Russian style, which was initiated by Markov, Šanin, Ceĭtin, and others. Demuth was born 1936 in Prague. In 1959 he graduated from the Faculty of Mathematics and Physics at Charles University, Prague with the equivalent of a masters degree. Thereafter he studied constructive mathematics in Moscow under the supervision of A. A. Markov jr., where he successfully defended his doctoral thesis (equivalent to a PhD thesis) in 1964. After that he returned to Charles University, where he worked, mostly in isolation, until the end of his life in 1988.

2 Demuth's World

Demuth used the Russian style terminology of constructive mathematics, adding some of his own terms and notions. In this paper, his definitions will be phrased

* A. Kučera is partially supported by the Research project of the Ministry of Education of the Czech Republic MSM0021620838. Nies is partially supported by the Marsden Foundation of New Zealand under grant 09-UOA-184. We wish to dedicate this paper to Cris Calude on the occasion of his 60th birthday. We are very grateful to Cris because he is the person who started the study of algorithmic randomness in New Zealand. Today it is one of the most active fields in logic and computer science. Contributions by New Zealand-based researchers were essential.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 159–173, 2012.

in the language of modern computable analysis, developed for instance in [34,6]. We will also use present-day terminology in algorithmic randomness as in [28].

From the beginning through the 1970s, in line with Russian style constructivism, Demuth only believed in computable reals, which he called constructive real numbers, and sometimes, simply, numbers.

Definition 1. A *computable real* z is given by a computable Cauchy name, i.e., a sequence $(q_n)_{n\in\mathbb{N}}$ of rationals converging to z such that $|q_r - q_n| \leq 2^{-n}$ for each $r \geq n$.

Demuth still accepted talking about Δ^0_2 reals, which he called pseudo-numbers. They are given as limits of computable sequences of rationals, so it was not necessary to view them as entities of their own. Later on, in the 1980s, he relaxed his standpoint somewhat, also admitting arithmetical reals.

The following is a central notion of Russian-style constructivism. Since in that context only computable reals actually exist, it is the most natural notion of computability for a function.

Definition 2. A function g defined on the computable reals is called *Markov computable* if from an index for a computable Cauchy name for x one can compute an index for a computable Cauchy name for $g(x)$.

Demuth called such functions *constructive*. By a *c-function* he meant a constructive function that is constant on $(-\infty, 0]$ and on $[1, \infty)$. This in effect restricts the domain to the unit interval (but a constructivist cannot write that into the definition since it is not decidable whether a given computable real is negative). By a result of Ceĭtin, and also a similar result of Kreisel, Shoenfield and Lacombe, each c-function is continuous on the computable reals. However, since such a function only needs to be defined on the computable reals, it is not necessarily uniformly continuous.

A *modulus of uniform continuity* for a function f is a function θ on positive rationals such that $|x-y| \leq \theta(\epsilon)$ implies $|f(x)-f(y)| \leq \epsilon$ for each rational $\epsilon > 0$. If a c-function is uniformly continuous (or equivalently, if it can be extended to a continuous function on $[0, 1]$) then it has a modulus of uniform continuity that is computable in $\emptyset'$. Demuth also considered $\emptyset$-uniformly continuous c-functions, i.e. c-functions which even have a computable modulus of uniform continuity; this is equivalent to computable functions on the unit interval in the usual sense of computable analysis (see [34,6]).

3 The Denjoy Alternative, and Pseudo-differentiability

The Denjoy alternative motivated a lot of Demuth's work on algorithmic randomness.

3.1 Background

For a function f, the *slope* at a pair a, b of distinct reals in its domain is

$$S_f(a,b) = \frac{f(a) - f(b)}{a - b}.$$

Recall that if z is in the domain of f then

$$\overline{D}f(z) = \limsup_{h \to 0} S_f(z, z+h)$$
$$\underline{D}f(z) = \liminf_{h \to 0} S_f(z, z+h)$$

Note that we allow the values $\pm\infty$. By the definition, a function f is differentiable at z if $\underline{D}f(z) = \overline{D}f(z)$ and this value is finite.

One simple version of the Denjoy alternative for a function f defined on the unit interval says that

$$\text{either } f'(z) \text{ exists, or } \overline{D}f(z) = \infty \text{ and } \underline{D}f(z) = -\infty. \tag{1}$$

It is a consequence of the classical Denjoy (1907), Young (1912), and Saks (1937) Theorem that for *any* function defined on the unit interval, the Denjoy alternative holds at almost every z. The full result is in terms of right and left upper and lower Dini derivatives denoted $D^+f(z)$ (right upper) etc. Denjoy himself obtained the Denjoy alternative for continuous functions, Young for measurable functions, and Saks for all functions. For a proof see for instance Bogachev [5, p. 371]. One application of this result is to show that f' is Borel (as a partial function) for any function f. A paper by Alberti et al. [1] revisits the Denjoy alternative. They provide a version that is in a sense optimal.

3.2 Pseudo-differentiability

If one wants to study the Denjoy alternative for Markov computable functions, one runs into the problem that they are only defined on computable reals. So one has to introduce upper and lower"pseudo-derivatives" at a real z, taking the limit of slopes close to z where the function is defined. This is presumably what Demuth did. Consider a function g defined on $I_{\mathbb{Q}}$, the rationals in $[0,1]$. For $z \in [0,1]$ let

$$\widetilde{D}g(z) = \limsup_{h \to 0^+} \{S_g(a,b) \colon a, b \in I_{\mathbb{Q}} \wedge\ a \le x \le b \wedge\ 0 < b - a \le h\}.$$
$$\underset{\sim}{D}g(z) = \liminf_{h \to 0^+} \{S_g(a,b) \colon a, b \in I_{\mathbb{Q}} \wedge\ a \le x \le b \wedge\ 0 < b - a \le h\}.$$

Definition 3. We say that a function f with domain containing $I_{\mathbb{Q}}$ is *pseudo-differentiable at* x if $-\infty < \underset{\sim}{D}f(x) = \widetilde{D}f(x) < \infty$.

Since Markov computable functions are continuous on the computable reals, it does not matter which dense set of computable reals one takes in the definition of these upper and lower pseudo-derivatives. For instance, one could take all computable reals, or only the dyadic rationals. For a total continuous function g, we have $\underset{\sim}{D}g(z) = \underline{D}g(z)$ and $\widetilde{D}g(z) = \overline{D}g(z)$. The last section of [7] contains more detail on pseudo-derivatives.

Definition 4. Suppose the domain of a function f contains $I_{\mathbb{Q}}$. We say that the *Denjoy alternative* holds for f at z if

$$\text{either } \widetilde{D}f(z) = \underset{\sim}{D}f(z) < \infty, \text{ or } \widetilde{D}f(z) = \infty \text{ and } \underset{\sim}{D}f(z) = -\infty. \tag{2}$$

This is equivalent to (1) if the function is total and continuous.

4 Martin-Löf Randomness and Differentiability

Demuth introduced a randomness notion equivalent to Martin-Löf (ML) randomness in the paper [10]. He was not aware of Martin-Löf's earlier definition in [27]. Among other things, Demuth gave his own proof that there is a universal Martin-Löf test.

The notion was originally only considered for pseudo-numbers (i.e., Δ^0_2 reals). As a constructivist, Demuth found it more natural to define the *non*-Martin-Löf random pseudo-numbers first. He called them Π_1 numbers. Pseudonumbers that are not Π_1 numbers were called Π_2 numbers. Thus, in modern language, the Π_2 numbers are exactly the Martin-Löf random Δ^0_2-reals.

As already noted, from around 1980 on Demuth also admitted arithmetical reals (possibly in parallel with the decline of communism, and thereby its background of philosophical materialism). In [14] he called the arithmetical *non*-ML-random reals $\mathcal{A}_1$ numbers, and the arithmetical ML-random reals $\mathcal{A}_2$ numbers. For instance, the definition of $\mathcal{A}_1$ can be found in [14, page 457]. By then, Demuth knew of Martin-Löf's work: he defined $\mathcal{A}_1$ to be $\bigcap_k [W_{\langle g \rangle (k)}]$, where g is a computable function determining a universal ML-test, and $[X]$ is the set of arithmetical reals extending a string in X. In the English language papers such as [18], the non-ML random reals were called AP (for approximable, or approximable in measure), and the ML-random reals were called NAP (for non-approximable).

Demuth needed Martin-Löf randomness for his study of differentiability of Markov computable functions (Definition 2), which he called constructive. The abstract of the paper [11], translated literally, is as follows:

> It is shown that every constructive function f which cannot fail to be a function of weakly bounded variation is finitely pseudo-differentiable on each Π_2 number.
>
> For almost every pseudo-number ξ there is a pseudo-number which is a value of pseudo-derivative of function f on ξ, where the differentiation is almost uniform.

Converted into modern language, the first paragraph says that each Markov computable function of bounded variation is (pseudo-)differentiable at each Martin-Löf random real. We do not know how Demuth proved this. However, his result has been recently reproved in [7] in an indirect way, relying on a similar result on computable randomness in the same paper [7]: each Markov computable nondecreasing function is differentiable at each computably random real.

The first part of the second paragraph expresses that for almost every Δ^0_2 real z, the derivative $f'(z)$ is also Δ^0_2. It is not clear what Demuth means by the

second part, that "the differentiation is almost uniform". One might guess it is similar to the definition of Markov computablility: from an index for z as a limit of a computable sequence of rationals, one can compute such an index for $f'(z)$.

The notion that a property $\mathcal{S}$ holds for "almost every" pseudo-number (i.e., Δ^0_2 real) is defined in [11, page 584]; see Figure 1.

Определние. Пусть $\mathcal{S}$ свойство псевдочисел, а $x \Delta y$ сегмент. Мы скажем, что $\mathcal{S}(\xi)$ выполнено для почти всех псевдочисел ξ (соотв. ξ из $x \Delta y$), если для всякого НЧ m существуют последовательность последовательностей рациональных сегментов $\{\{Q^m_k\}_k\}_m$ и последовательность неинфинитных рекурсивно перечислимых (р.п.) множеств НЧ $\{D_m\}_m$ такие, что $\forall m\, l \left(\sum_{1 \le k \le l \,\&\, \neg(k \in D_m)} |Q^m_k| < \frac{1}{2^{m+m}}\right)$ и для всякого ПЧ ξ (соотв. ПЧ ξ из $x \Delta y$) верно $(\neg \exists m\, k (\neg (k \in D_m) \,\&\, \xi \in Q^m_k) \supset \mathcal{S}(\xi))$ (см. лемму 7 из [2]).

Fig. 1. [11, page 584]: Definition of interval sequence tests

We rephrase this definition in modern (but classical) language. Demuth introduces a notion of tests; let us call them *interval sequence tests*. In the following let m, r, k range over the set $\mathbb{N}^+$ of positive integers. An interval sequence test uniformly in a number $m \in \mathbb{N}^+$ provides a computable sequence of rational intervals $(Q^m_r(k))_{r,k\in\mathbb{N}^+}$, and a uniformly c.e. sequence of finite sets $(E^m_r)_{r\in\mathbb{N}^+}$, such that

$$\lambda\Big(\bigcup\{Q^m_r(k) \colon k \notin E^m_r\}\Big) \le 2^{-(m+r)} \tag{3}$$

(where λ denotes Lebesgue measure). A real z *fails* the test if for each m there is r such that for some $k \notin E^m_r$ we have $z \in Q^m_r(k)$. In other words, for each m,

$$z \in \bigcup_r \bigcup_{k \notin E^m_r} Q^m_r(k). \tag{4}$$

Note that the class in (4) has measure at most 2^{-m}, hence the reals z failing the test form a null set. If z does not fail the test we say that z *passes* the test. Demuth says that a property $\mathcal{S}$ holds for *almost all* reals z if there is an interval sequence test (depending on $\mathcal{S}$) such that $\mathcal{S}$ holds for all z passing the test.

Recall that a Schnorr test is a Martin-Löf test $(G_m)_{m\in\mathbb{N}^+}$ such that λG_m is a computable real uniformly in m. We say that a real z fails the Schnorr test if $z \in \bigcap_m G_m$. (See [28, 3.5.8].)

Corollary 5 (with Hirschfeldt). *Interval sequence tests are uniformly equivalent to Schnorr tests relative to $\emptyset'$. That is, given a test of one kind, we can effectively determine a test of the other kind so that every real fails the first test if and only if it fails the second test.*

Proof. Firstly, suppose we are given an interval sequence test

$$(Q_r^m(k))_{r,k\in\mathbb{N}^+},\ (E_r^m)_{r\in\mathbb{N}^+}\ (m\in\mathbb{N}^+).$$

Let G_m be the class in (4). Then G_m is $\Sigma_1^0(\emptyset')$ uniformly in m, and λG_m is computable relative to $\emptyset'$ by (3).

Secondly, suppose we are given a Schnorr test $(G_m)_{m\in\mathbb{N}^+}$ relative to $\emptyset'$. Uniformly in m, using $\emptyset'$ as an oracle we can compute λG_m for each $m\in\mathbb{N}^+$. Hence we can for each $r,m\in\mathbb{N}^+$ determine $u_r\in\mathbb{N}$ and, by possibly splitting into pieces some intervals from G_m, a finite sequence of rational intervals $P_r^m(i)$, $u_r<i\leq u_{r+1}$, such that $\lambda(\bigcup_{u_r<i\leq u_{r+1}} P_r^m(i))\leq 2^{-(m+r)}$ and $G_m=\bigcup_r\bigcup_{u_r<i\leq u_{r+1}} P_r^m(i)$. By the Limit Lemma we have a computable sequence of intervals $P_r^m(i,t)$ and a computable sequence $u_r(t)$, $t\in\mathbb{N}$, such that for large enough t, $u_r(t)=u_r$ and $P_r^m(i,t)=P_r^m(t)$ for $i\leq u_r$. From this we can build an interval sequence test as required: the uniformly c.e. finite sets E_r^m correspond to the intervals we want to remove because of the mind changes of the approximations $u_r(t)$ and $P_r^m(i,t)$ for $i\leq u_r(t)$.

Above we quoted the abstract of the paper [11]. The first part of the second paragraph asserts that for almost every Δ_2^0 real z, the derivative $f'(z)$ is also Δ_2^0. Since f is Markov computable, it is easy to verify that

$$f'(z)\leq_T z',$$

namely, the value of the pseudo-derivative of f at z is computable in the Turing jump of z, whenever this pseudo-derivative exists. Thus $f'(z)$ is Δ_2^0 whenever z is low. By [18, Remark 10, part 3b], or [28, 3.6.26], there is a single Schnorr test relative to $\emptyset'$ (in fact, a Demuth test as defined in 11 below) such that each real z passing it is generalized low (i.e., $z'\leq z\oplus\emptyset'$). Thus, we know how to obtain the first part of that paragraph; the point is the *second* part, that the derivative can be obtained uniformly.

5 Denjoy Alternative and Denjoy Sets

For any function $g\colon[0,1]\to\mathbb{R}$, the reals z such that $\underline{D}g(z)=\infty$ form a null set. This well-known fact from classical analysis is usually proved via covering theorems, such as Vitali's or Sierpinskis's. Cater [8] has given an alternative proof of a stronger fact: the reals z where the right lower derivative $D_+(z)$ is infinite form a null set.

Demuth knew results of this kind. He studied the question which type of null class is needed to make an analog of this classic fact hold for Markov computable functions (see Definition 2). The following definition originates in [13]. As usual, for functions not defined everywhere we have to work with pseudo-derivatives defined in Subsection 3.2.

Definition 6. A real $z \in [0,1]$ is called Denjoy random (or a Denjoy set) if for no Markov computable function g we have $\underset{\sim}{D}g(z) = \infty$.

The paper [13] is entitled "The constructive analogue of a theorem by Garg on derived numbers". Garg's Theorem, a variant of the Denjoy-Young-Saks theorem discussed in Subsection 3.1, has the somewhat obscure reference [22].

The work of Demuth on the Denjoy alternative for effective functions is described in the preprint survey "Remarks on Denjoy sets" [17]. This is based on a talk Demuth gave at the Logic Colloquium 1988 in Padova, Italy (close to the end of communist era in 1989, it became easier to travel to the "West"). He later turned the preprint survey into the paper [19] with the same title, but it contains only part of the preprint survey.

In the preprint survey [17, page 6] it is shown that if $z \in [0,1]$ is Denjoy random, then for every computable $f\colon [0,1] \to \mathbb{R}$ the Denjoy alternative (1) holds at z. Combining this with the results in [7] we can now figure out what Denjoy randomness is, and also obtain a pleasing new characterization of computable randomness of reals through differentiability of computable functions. Joseph S. Miller also contributed to the theorem.

Theorem 7. *The following are equivalent for a real $z \in [0,1]$*

(i) z is Denjoy random.
(ii) z is computably random
(iii) for every computable $f\colon [0,1] \to \mathbb{R}$ the Denjoy alternative (1) holds at z.

Proof. (i)→(iii) is Demuth's result. For (iii)→(ii), let f be a nondecreasing computable function. Then f satisfies the Denjoy alternative at z. Since $\underline{D}f(z) \geq 0$, this means that $f'(z)$ exists. This implies that z is computably random by [7, Thm. 4.1].

The implication (ii)→(i) is proved by contraposition: if g is Markov computable and $\underset{\sim}{D}g(z) = \infty$ then one builds a computable betting strategy showing that z is not computably random. See [4, Thm. 15] or Section 2 of the Logic Blog [2] for proofs.

Remark 8. For the contraposition of the implication (ii)→(i), actually the weaker hypothesis on g suffices that $g(q)$ is a computable real uniformly in a rational $q \in I_{\mathbb{Q}}$.

We do not know at present how Demuth obtained (i)→(iii) of the theorem; a proof of this using classical language would be useful. However, a direct proof of the contraposition of (i)→(ii) is in [7, Thm. 3.6]: if z is not computably random then a martingale M with the so-called "savings property" succeeds on (the binary expansion of) a real z. The authors now build in fact a computable function g such that $\underline{D}g(z) = \underset{\sim}{D}g(z) = \infty$. Together with Remark 8 we obtain:

Corollary 9. *The following are equivalent for a real z:*

(i) For no function g such that $g(q)$ is uniformly computable for $q \in I_{\mathbb{Q}}$ we have $\underset{\sim}{D}g(z) = \infty$.

(ii) z is Denjoy random, i.e., for no Markov computable function g we have $\underset{\sim}{D}g(z) = \infty$.
(iii) For no computable function g we have $\underline{D}g(z) = \infty$.

This implies that the particular choice of Markov computable functions in Definition 6 is irrelevant. Similar equivalences stating that the exact level of effectivity of functions does not matter have been obtained in the article [7]. For instance, in [7, Thm. 7.3], extending a result of Demuth [11] the authors characterize Martin-Löf randomness via differentiability of effective functions of bounded variation. This works with any of the three particular effectiveness properties above: computable, Markov computable, and uniformly computable on the rationals. For nondecreasing *continuous* functions, the three effectiveness properties coincide as observed in [7, Prop. 2.2]

Because of Theorem 7 one could assert that Demuth studied computable randomness indirectly via his Denjoy sets. Presumably he didn't know the notion of computable randomness, which was introduced by Schnorr in [32], a monograph in German (see [28, Ch. 7]). Demuth also proved in [18, Thm. 2] that every Denjoy set that is AP (i.e., non ML-random) must be high. The analogous result for computable randomness was later obtained in [30]. There the authors also show a kind of converse: each high degree contains a computably random set that is not ML-random. This fact was apparently not known to Demuth.

6 Demuth Randomness and Weak Demuth Randomness

As told above, Demuth knew that Denjoy randomness of a real z implies the Denjoy alternative at z for all computable functions. The next question for Demuth to ask was the following:

How much randomness for a real z is needed to ensure the Denjoy alternative at z for all Markov computable functions?

Demuth showed the following (see preprint survey, page 7, Thm 5, item 4), which refers to [12].

Corollary 10. *There is a Markov computable function f such that the Denjoy alternative fails at some ML-random real z. Moreover, f is extendable to a continuous function on* $[0, 1]$.

This has been reproved by Bienvenu, Hölzl, Miller and Nies [4]. In their proof, z can be taken to be the least element of an arbitrary effectively closed set of reals containing all the ML-random reals but no computable reals. In particular, one can make z left-c.e.

6.1 The Definition of (Weak) Demuth Randomness

It was now clear to Demuth that a randomness notion stronger than Martin-Löf's was needed. Weak 2-randomness may have seemed ignoble to him because a Δ^0_2 real cannot be weakly 2-random. He needed a notion compatible with being Δ^0_2.

Such a notion was introduced in the paper "Some classes of arithmetical reals" [14, page 458]. The definition is reproduced in the preprint survey [17, page 4].

For X a set of binary strings, let $[X]^{\prec}$ denote the collection of infinite binary sequences (sets) extending a string in X. In modern (but classical) language the definitions are as follows.

Definition 11. *A* Demuth test *is a sequence of c.e. open sets* $(S_m)_{m\in\mathbb{N}}$ *such that* $\forall m\, \lambda S_m \leq 2^{-m}$*, and there is a function* $f\colon \mathbb{N} \to \mathbb{N}$ *with* $f \leq_{\mathrm{wtt}} \emptyset'$ *such that* $S_m = [W_{f(m)}]^{\prec}$.

A set Z passes *the test if* $Z \notin S_m$ *for almost every* m*. We say that* Z *is* Demuth random *if* Z *passes each Demuth test.*

Recall that $f \leq_{\mathrm{wtt}} \emptyset'$ if and only if f is ω-c.e., namely, $f(x) = \lim_t g(x,t)$ for some computable function g such that the number of stages t with $g(x,t) \neq g(x,t-1)$ is bounded in x. Hence the intuition is that we can change the m-th component S_m for a computably bounded number of times.

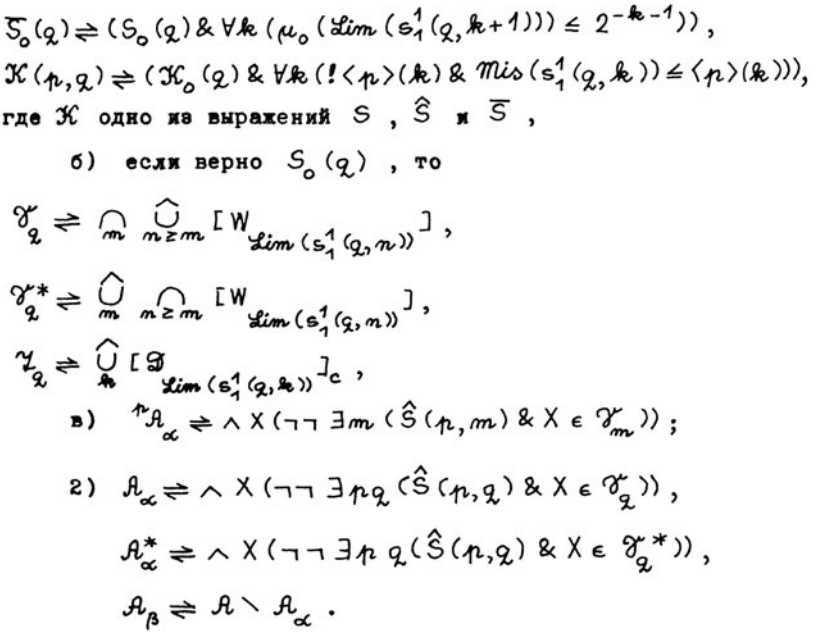


Fig. 2. [14, page 458]: $\mathcal{A}_\beta$ is the definition of Demuth randomness

Fig. 2 shows what the definition of Demuth randomness looks like in the 1982 paper [14, p. 458]. Demuth first defines tests via certain conditions γ_q, where q is an index for a binary computable function $\phi_q(x,k)$. The condition γ_q holds for a real z if

$$\forall m \exists k \geq m \; z \text{ is in } [W_{\lim(s_1^1(q,k))}]$$

(where his notation $[X]$ is equivalent to our notation $[X]^{\prec}$). The expression in the subscript in the same line simply means $\lim_x \phi_q(x,k)$, which is the final version r of the test. A further condition $\mathcal{K}(p,q)$, involving an index p for a computable unary function, yields the bound $\phi_p(k)$ on the number of changes. The bound 2^{-k} on measures of the k-th component can be found in the top line. The notation $Mis(s_1^1(q,k))$ in Fig. 2 refers to the number of "mistakes", i.e. changes, and Demuth requires it be bounded by $\langle p\rangle(k)$, meaning $\phi_p(k)$.

If we apply the usual passing condition for tests, we obtain the following notion which only occurs in [14, page 458].

Definition 12. *We say that a set $Z \subseteq \mathbb{N}$ is* weakly Demuth random *if for each Demuth test $(S_m)_{m\in\mathbb{N}}$ there is an m such that $Z \notin S_m$.*

In [14] this is given by conditions γ_q^*, where the quantifiers are switched compared to γ_q:

$$\exists m \forall k \geq m \ z \text{ is in } [W_{\lim(s_1^1(q,k))}].$$

The class of arithmetical non-Demuth randoms is called $\mathcal{A}_\alpha$, and the class of arithmetical non-weakly Demuth randoms is called $\mathcal{A}_\alpha^*$. The complement of $\mathcal{A}_\alpha$ within the arithmetical reals is called $\mathcal{A}_\beta$ and, similarly, the complement of $\mathcal{A}_\alpha^*$ within the arithmetical reals is called $\mathcal{A}_\beta^*$. Later on, in the preprint survey, Demuth used the terms WAP sets (weakly approximable) for the non-Demuth randoms, and NWAP for the Demuth randoms and analogously, in an obviuous sense, the terms WAP* sets and NWAP* sets.

6.2 The Denjoy Alternative for Markov Computable Functions

In the preprint survey [17, page 7, Thm 5, item 5)], Demuth states that Demuth randomness is sufficient to get the Denjoy alternative for Markov computable functions. This refers to the paper [15].

Corollary 13. *Let z be a Demuth random real. Then the Denjoy Alternative holds at z for every Markov computable function.*

This result is actually hard to pin down in [15]. Theorem 2 on page 399 comes close, but has some extra conditions not present in the original Denjoy alternative.

Remark 14. Franklin and Ng [21] introduced difference randomness, a concept much weaker than even weak Demuth randomness, but still stronger than Martin-Löf randomness. Bienvenu, Hölzl and Nies [4, Thm. 1] have shown that difference randomness is sufficient as a hypothesis on the real z in Theorem 13. No converse holds. They also show that the "randomness notion" to make the Denjoy Alternative hold for each Markov computable function is incomparable with ML-randomness!

6.3 Demuth Randomness Finds Itself

We have seen that Demuth randomness of a real is way too strong for its original purpose, ensuring that the Denjoy alternative holds at this real for all Markov computable functions. However, Demuth randomness has recently turned out to be a very interesting notion on its own. Since it is stronger than ML-randomness but still allows the real to be Δ^0_2, it interacts nicely with computability theoretic notions. For instance, Kučera and Nies [25] proved that every c.e. set Turing below a Demuth random is strongly jump traceable (see [28, Section 8.4] for a definition of this lowness notion). Greenberg and Turetsky have recently provided a converse of this result of Kučera and Nies: every c.e. strongly jump traceable set has a Demuth random set Turing above. Nies [29] showed that each base for Demuth randomness is strongly jump traceable. Greenberg and Turetsky proved that this inclusion is proper.

Lowness for Demuth randomness and weak Demuth randomness have been characterized by Bienvenu et al. [3]. The former is given by a notion called BLR-traceability, in conjunction with being computably dominated. The latter is the same as being computable.

7 Late Work Related to Computability Theory

In the 1980s the mathematics department at Charles University had a seminar on recursion theory, which was based on Rogers' book [31] and some draft of Soare's book [33]. Because of this, Demuth became more interested in computability theory and the computational complexity of random sets.

7.1 Randomness and Computational Complexity

Demuth proved the following.

Corollary 15. *(i) Each Demuth random real z satisfies $z' \leq z \oplus \emptyset'$.*
(ii) Each Demuth random set is of hyperimmune T-degree.

(i). Demuth [18, Remark 10, part 3b] gives a sketch of a proof. As mentioned, a full proof can be found in [28, 3.6.26].
(ii). Only a sketch of a proof is given in Remark 2 and Remark 11 of the preprint survey. It seems that a single Demuth test is sufficient here. An alternative proof can be derived from (i) and the result of Miller and Nies [28, Thm. 8.1.19] that no GL_1 set of hyperimmune-free degree is d.n.c.

It is of interest that Kučera and Demuth ([20], Theorem 18) proved a result very similar to a later result of Miller and Yu (see, [28], 5.1.14). For a Turing functional Φ and $n > 0$, consider the open set

$$S^A_{\Phi,n} = [\{\sigma\colon\ A\restriction_n \preceq \Phi^\sigma\}]^\prec.$$

If A is ML-random then there is a constant c such that $\forall n\, \lambda S^A_{\Phi,n} \leq 2^{-n+c}$.

7.2 Work on Semigenericity

The following direction of Demuth's late work is only loosely connected to randomness. An incomputable set Z is called *semigeneric* [16] if every Π^0_1 class containing Z has a computable member. Any ML-random set is contained in a whole Π^0_1 class of ML-randoms, and is therefore not semigeneric. Intuitively, to be semigeneric means to be close to computable in the sense that the set cannot be separated from the computable sets by a Π^0_1 class.

Demuth proved in [16, Thm. 9] that if a set Z is semigeneric then any set B such that $\emptyset <_{tt} B \leq_{tt} Z$ is also semigeneric. In particular, its tt-degree only contains semigeneric sets.

Demuth and Kučera [20] studied semigenericity and its relationship with other types of genericity. We review some of their results.

Ceĭtin's notion of strong undecidability. Ceĭtin [9] called a set Z *strongly undecidable* if there is a computable function p such that for any computable set M and any index v of its characteristic function, $p(v)$ is defined and $Z\restriction_{p(v)} \neq M\restriction_{p(v)}$.

By Demuth and Kučera [20, Cor. 2], an incomputable set Z is semigeneric if and only if Z is not strongly undecidable. Furthermore, strong undecidability can be characterized by some kind of "uniform non-hyperimmunity": by [20, Thm. 5], a set Z is strongly undecidable if and only if there is a computable function f such that for each computable set M and any index v of its characteristic function, the symmetric difference $M \triangle Z$ is infinite, and its listing in order of magnitude dominated by the computable function with index $f(v)$.

Demuth and Kučera [20, Thm. 14] characterize the sets Z such that the Turing-degree of Z contains a strongly undecidable set: this happens precisely when there is a Π^0_2 class containing Z but no computable sets. So we have a weaker form of separation from the computable sets than for incomputable sets that are not semigeneric (i.e. strongly undecidable sets per se), where the separating class is Π^0_1 by definition.

The result [20, Thm. 14] was actually proved in terms of so-called V-coverings (where V stands for Vitali). A set Z is V-covered by a c.e. set of strings A if for all k there is a string $\sigma \in A$ such that $|\sigma| \geq k$ and $\sigma \prec Z$. It is easy to see that a class of sets $\mathcal{A}$ is a Π^0_2 class if and only if there is a c.e. set of strings B such that $\mathcal{A}$ is equal to the class of sets V-covered by B (see [28, 1.8.60]).

Connection to weak 1-genericity and hyperimmunity. Recall that a set Z is weakly 1-generic if Z is in each dense Σ^0_1 class (see [28, 1.8.47]). Clearly any weakly 1-generic set is semigeneric. The converse fails.

Demuth [16, Thm. 16] showed that a set Z is weakly 1-generic if and only if for any computable set M the symmetric difference $M \triangle Z$ is hyperimmune. Kurtz [23,24] proved that a Turing-degree contains a weakly 1-generic set if and only if it is hyperimmune. It follows from Kurtz's results, using a fact of Martin-Miller [26], that the weakly 1-generic T-degrees are closed upwards. As a corollary we have that there are weakly 1-generic Turing degrees which do contain ML-random sets and, thus, they can compute d.n.c. functions. On the other hand,

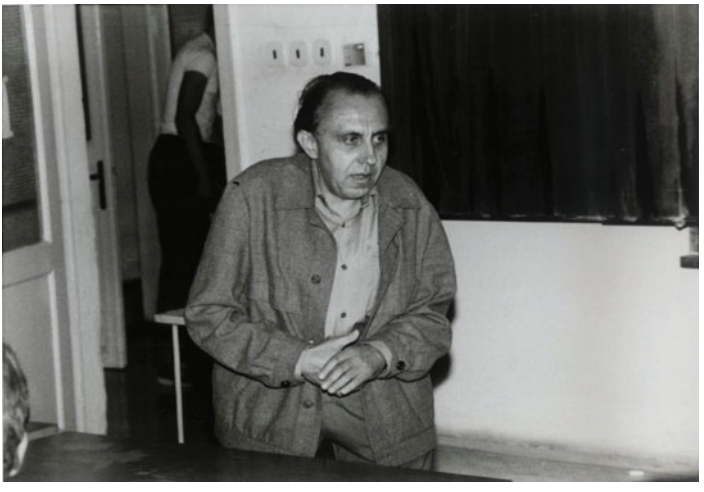


Fig. 3. Demuth by the blackboard

Kučera and Demuth showed that the classes of 1-generic Turing degrees and of Turing degrees of d.n.c. functions are disjoint. In fact, they proved in [20, Cor. 2] that no d.n.c. function (and, thus, no ML-random set) is computable in a 1-generic set (also see [28, 4.1.6]).

Demuth [16, Cor. 12] proved that any hyperimmune or co-hyperimmune set is semigeneric. Furthermore, he showed in [16, Thm. 21] that there is a semigeneric set E (even hypersimple) such that no set $A \leq_{tt} E$ is weakly 1-generic.

Final remarks. The searchable database at `http://www.dml.cz` contains most papers of Demuth. We plan to submit an extended journal version of this paper to the Bull. Symb. Logic in 2012.

References

1. Alberti, G., Csörnyei, M., Laczkovich, M., Preiss, D.: The Denjoy-Young-Saks theorem for approximate derivatives revisited. Real Anal. Exchange 26(1), 485–488 (2000/2001)
2. Various Authors. Logic Blog (2011), `http://dl.dropbox.com/u/370127/Blog/Blog2011.pdf`
3. Bienvenu, L., Downey, R., Greenberg, N., Nies, A., Turetsky, D.: Lowness for Demuth randomness. Unpublished, 20xx
4. Bienvenu, L., Hoelzl, R., Miller, J., Nies, A.: The Denjoy alternative for computable functions (2012), `http://dl.dropbox.com/u/370127/papers/BHMNStacsFinal.pdf`
5. Bogachev, V.I.: Measure theory, vol. I, II. Springer, Berlin (2007)
6. Brattka, V., Hertling, P., Weihrauch, K.: A tutorial on computable analysis. In: Barry Cooper, S., Löwe, B., Sorbi, A. (eds.) New Computational Paradigms: Changing Conceptions of What is Computable, pp. 425–491. Springer, New York (2008)

7. Brattka, V., Miller, J., Nies, A.: Randomness and differentiability (submitted)
8. Cater, F.S.: Some analysis without covering theorems. Real Anal. Exchange 12(2), 533–540 (1986)
9. Ceĭtin, G.S.: On upper bounds of recursively enumerable sets of constructive real numbers. Proc. Steklov Inst. Math. 113, 119–194 (1970)
10. Demuth, O.: Constructive pseudonumbers. Comment. Math. Univ. Carolinae 16, 315–331 (1975)
11. Demuth, O.: The differentiability of constructive functions of weakly bounded variation on pseudo numbers. Comment. Math. Univ. Carolin. 16(3), 583–599 (1975) (Russian)
12. Demuth, O.: The constructive analogue of the Denjoy-Young theorem on derived numbers. Comment. Math. Univ. Carolinae 17(1), 111–126 (1976)
13. Demuth, O.: The constructive analogue of a theorem by Garg on derived numbers. Comment. Math. Univ. Carolinae 21(3), 457–472 (1980)
14. Demuth, O.: Some classes of arithmetical real numbers. Comment. Math. Univ. Carolin. 23(3), 453–465 (1982)
15. Demuth, O.: On the pseudodifferentiability of pseudo-uniformly continuous constructive functions from functions of the same type. Comment. Math. Univ. Carolin. 24(3), 391–406 (1983)
16. Demuth, O.: A notion of semigenericity. Comment. Math. Univ. Carolin. 28(1), 71–84 (1987)
17. Demuth, O.: Remarks on Denjoy sets, preprint (1988), http://dl.dropbox.com/u/370127/DemuthPapers/Demuth88PreprintDenjoySets.pdf
18. Demuth, O.: Remarks on the structure of tt-degrees based on constructive measure theory. Comment. Math. Univ. Carolin. 29(2), 233–247 (1988)
19. Demuth, O.: Remarks on Denjoy sets. In: Mathematical logic, pp. 267–280. Plenum, New York (1990)
20. Demuth, O., Kučera, A.: Remarks on 1-genericity, semigenericity and related concepts. Commentationes Mathematicae Universitatis Carolinae 28(1), 85–94 (1987)
21. Franklin, J.N.Y., Ng, K.M.: Difference randomness. Proceedings of the American Mathematical Society (to appear)
22. Garg, K.M.: An analogue of Denjoy's theorem. Ganita 12, 9–14 (1961)
23. Kurtz, S.: Randomness and genericity in the degrees of unsolvability. Ph.D. Dissertation, University of Illinois, Urbana (1981)
24. Kurtz, S.: Notions of weak genericity. J. Symbolic Logic 48, 764–770 (1983)
25. Kučera, A., Nies, A.: Demuth randomness and computational complexity. Ann. Pure Appl. Logic 162, 504–513 (2011)
26. Martin, D.A., Miller, W.: The degrees of hyperimmune sets. Z. Math. Logik Grundlag. Math. 14, 159–166 (1968)
27. Martin-Löf, P.: The definition of random sequences. Inform. and Control 9, 602–619 (1966)
28. Nies, A.: Computability and randomness. Oxford Logic Guides, vol. 51. Oxford University Press, Oxford (2009)
29. Nies, A.: Computably enumerable sets below random sets. Ann. Pure Appl. Logic (to appear, 2011)
30. Nies, A., Stephan, F., Terwijn, S.: Randomness, relativization and Turing degrees. J. Symbolic Logic 70(2), 515–535 (2005)

31. Rogers Jr., H.: Theory of Recursive Functions and Effective Computability. McGraw-Hill, New York (1967)
32. Schnorr, C.P.: Zufälligkeit und Wahrscheinlichkeit. Eine algorithmische Begründung der Wahrscheinlichkeitstheorie. Lecture Notes in Mathematics, vol. 218. Springer, Berlin (1971)
33. Soare, R.I.: Recursively Enumerable Sets and Degrees. Perspectives in Mathematical Logic, Omega Series. Springer, Heidelberg (1987)
34. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)

A Computability Challenge: Asymptotic Bounds for Error-Correcting Codes

Yuri I. Manin

Max–Planck–Institut für Mathematik, Bonn, Germany
manin@mpim-bonn.mpg.de

Dedicated to Professor C. S. Calude on his 60th birthday

Abstract. Consider the set of all error-correcting block codes over a fixed alphabet with q letters. It determines a recursively enumerable set of points in the unit square with coordinates $(R, \delta) :=$ *(relative transmission rate, relative minimal distance).* Limit points of this set form a closed subset, defined by $R \leq \alpha_q(\delta)$, where $\alpha_q(\delta)$ is a continuous decreasing function called *asymptotic bound.* Its existence was proved by the author in 1981, but all attempts to find an explicit formula for it so far failed.

In this note I consider the question whether this function is computable in the sense of constructive mathematics, and discuss some arguments suggesting that the answer might be negative.

1 Introduction

1.1. Notation. This paper is a short survey focusing on an unsolved problem of the theory of error-correcting codes (cf. the monograph [VlaNoTsfa]).

Briefly, we choose and fix an integer $q \geq 2$ and a finite set, *alphabet* A, of cardinality q. An (unstructured) *code* C is defined as a non-empty subset $C \subset A^n$ of words of length $n \geq 1$. Such C determines its *code point* $P_C = (R(C), \delta(C))$ in the (R, δ)-plane, where $R(C)$ is called *the transmission rate* and $\delta(C)$ is *the relative minimal distance of the code.* They are defined by the formulas

$$\delta(C) := \frac{d(C)}{n(C)}, \quad d(C) := \min\{d(a,b) \,|\, a, b \in C, a \neq b\}, \quad n(C) := n,$$

$$R(C) = \frac{k(C)}{n(C)}, \quad k(C) := \log_q \operatorname{card}(C), \tag{1.1}$$

where $d(a, b)$ is the Hamming distance

$$d((a_i), (b_i)) := \operatorname{card}\{i \in (1, \ldots, n) \,|\, a_i \neq b_i\}.$$

In the degenerate case $\operatorname{card} C = 1$ we put $d(C) = 0$. We will call the numbers $k = k(C)$, $n = n(C)$, $d = d(C)$, *code parameters* and refer to C as an $[n, k, d]_q$-code.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 174–182, 2012.

A considerable bulk of research in this domain is dedicated either to the construction of (families of) "good" codes (e.g. algebraic-geometric ones), or to the proof that "too good" codes do not exist. A code is good if in a sense it maximizes simultaneously the transmission rate and the minimal distance. To be useful in applications, a good code must also come with feasible algorithms of encoding and decoding. The latter task includes the problem of finding a closest (in Hamming's metric) word in C, given an arbitrary word in A^n that can be an output of a noisy transmission channel (error correction). Feasible algorithms exist for certain classes of *structured* codes. The simplest and most popular example is that of *linear codes*: A is endowed with a structure of a finite field $\mathbf{F}_q$, A^n becomes a linear space over $\mathbf{F}_q$, and C is required to be a linear subspace.

1.2. Asymptotic Bounds. Since the demands of good codes are mutually conflicting, it is natural to look for the bounds of possible.

A precise formulation of the notion of good codes can be given in terms of two notions: *asymptotic bounds* and *isolated codes.*

Fix q and denote by V_q the set of all points P_C, corresponding to all $[n, k, d]_q$-codes. Define *the code domain* U_q as *the set of limit points of* V_q.

It was proved in [Man1] that U_q consists of all points in $[0, 1]^2$ lying below the graph of a certain continuous decreasing function α_q:

$$U_q = \{(R, \delta) \mid R \leq \alpha_q(\delta)\}. \tag{1.2}$$

Moreover, $\alpha_q(0) = 1, \alpha_q(\delta) = 0$ for $1 - q^{-1} \leq \delta \leq 1$, and the graph of α_q is tangent to the R-axis at $(1, 0)$ and to the δ-axis at $(0, 1 - q^{-1})$. This curve is called *the asymptotic bound*. (In fact, [Man1] considered only linear codes, and the respective objects are now called $V_q^{lin}, U_q^{lin}, \alpha_q^{lin}$; the unstructured case can be treated in the same way with minimal changes: cf. [ManVla] and [ManMar]).

Now, a code can be considered a good one, if its point either lies in U_q and is close to the asymptotic bound, or is *isolated*, that is, lies above the asymptotic bound.

1.3. Computability Problems. There is an abundant literature establishing upper and lower estimates for asymptotic bounds, and providing many isolated codes. However, not only "exact formulas" for asymptotic bounds are unknown, but even the question, whether $\alpha_q(\delta)$ is differentiable, remains open (of course, since this function is monotone and continuous, it is differentiable *almost everywhere*.) Similarly, the structure of the set of isolated code points is a mystery: for example, *are there points on* $R = \alpha_q(\delta), 0 < R < 1 - q^{-1}$, *that are limit points of isolated codes?*

The principle goal of this report is to discuss weaker versions of these problems, replacing "exact formulas" by "computability". In particular, we try to elucidate the following

QUESTION. Is the function $\alpha_q(\delta)$ *computable?*

As our basic model of computability we adopt the one described in [BratWe] and further developed in [BratPre], [Brat], [BratMiNi]. In its simplest concrete version, it involves approximations of closed subsets of $\mathbf{R}^2$, such as U_q or graph of α_q, by unions of computable sets of rational coordinate squares, "pixels" of varying size.

The following mental experiment suggests that the answer to this computability problem may not be obvious, and that α_q might even be *uncomputable* and by implication not expressible by any reasonable "explicit formula".

Imagine that a computer is drawing finite approximations $V_q^{(N)}$ to the set of code points V_q by plotting all points with $n \leq N$ for a large N (appropriately matching a chosen pixel size). What will we see on the screen?

Conjecturally, we will *not* see a dark domain approximating U_q with a cloud of isolated points above it, but rather an eroded version of the *Varshamov–Gilbert curve* lying (at least partially) strictly below $R = \alpha_q(\delta)$:

$$R = \frac{1}{2}\left(1 - \delta\log_q(q-1) - \delta\log_q\delta - (1-\delta)\log_q(1-\delta)\right) \tag{1.3}$$

In fact, "most" code points lie "near" (1.3): cf. Exercise 1.3.23 in [VlaNoTsfa] and some precise statements in [BaFo] (for $q = 2$.)

By contrast, a statistical meaning of the asymptotic bound does not seem to be known, and this appears as the intrinsic difficulty for a complete realization of the project started in [ManMar]: interpreting asymptotic bound as a "phase transition" curve. Hopefully, a solution might be found if we imagine plotting code points in the order of their *growing Kolmogorov complexity*, as was suggested and used in [Man3] for renormalization of halting problem. For the context of constructive mathematics, cf. [CaHeWa] and references therein.

In any case, it is clear that code domains represent an interesting testing ground for various versions of computability of subsets of $\mathbf{R}^n$, complementing the more popular Julia and Mandelbrot fractal sets (cf. [BravC] and [BravYa]).

2 Code Parameters and Code Points: A Summary

2.1. Constructive Worlds of Code Parameters. Denote the set of all triples $[n, q^k, d] \in \mathbf{N}^3$ corresponding to all (resp. linear) $[n, k, d]_q$-codes by P_q (resp. P_q^{lin}). Clearly, P_q and P_q^{lin} are infinite decidable subsets of $\mathbf{N}^3$. Therefore they admit natural recursive and recursively invertible bijections with $\mathbf{N}$ ("admissible numberings"), defined up to composition with any recursive permutation $\mathbf{N} \to \mathbf{N}$. Hence P_q and P_q^{lin} are infinite *constructive worlds* in the sense of [Man3], Definition 1.2.1.

If X, Y are two constructive worlds, we can unambiguously define the notions of (partial) recursive maps $X \to Y$, enumerable and decidable subsets of X, Y, $X \times Y$ etc., simply pulling them back to the numberings. For a more developed categorical formalism, cf. [Man3].

2.2. Constructive World $\mathbf{S} = [0,1]^2 \cap \mathbf{Q}^2$. The set of all rational points of the unit square in the (R, δ)-plane also has a canonical structure of a constructive world.

2.3. Enumerable Sets of Code Points. Code points (1.1) of linear codes all lie in $\mathbf{S}$. To achieve this for unstructured codes, we will slightly amend (1.1) and define the map $cp: P_q \to \mathbf{S}$ (cp stands for "code point") by

$$cp([n, q^k, d]) := \left(\frac{[k]}{n}, \frac{d}{n}\right) \tag{2.1}$$

where $[k]$ denotes the integer part of the (generally real) number k. On $P_q^{lin} \subset P_q$ it coincides with (1.1).

The motivation for choosing (2.1) is this: in the eventual study of computability properties of the graph $R = \alpha_q(\delta)$, it is more transparent to approximate it by points with rational coordinates, rather than logarithms.

Let V_q (resp. V_q^{lin}) be the image $cp(P_q)$ (resp. $cp(P_q^{lin})$) i.e. the respective set of code points in $\mathbf{S}$. Since cp is a total recursive function both on P_q and P_q^{lin}, V_q and V_q^{lin} are recursively enumerable subsets of $\mathbf{S}$.

2.4. Limit Code Points. Let U_q (resp. U_q^{lin}) be the closed sets of limit points of V_q (resp. V_q^{lin}). We will call *limit code points* elements of $V_q \cap U_q$ (resp. $V_q^{lin} \cap U_q^{lin}$). The remaining subset of *isolated code points* is defined as $V_q \setminus V_q \cap U_q$, and similarly for linear codes.

Notice that we get one and the same set U_q, using transmission rates (1.1) or (2.1). In fact, for any infinite sequence of pairwise distinct code parameters $[n_i, q^{k_i}, d_i]$, $i = 1, 2, \ldots$ we have $n_i \to \infty$, hence the convergence of the sequence of code points (1.1) is equivalent to that of (2.1), and they have a common limit. The resulting sets of isolated code points differ depending on the adopted definition (1.1) or (2.1), however, the set of *isolated codes*, those whose code points are isolated, remains the same.

Our main result in this section is the following characterization of limit and isolated code points in terms of the recursive map cp rather than topology of the unit square.

We will say that a code point $x \in V_q$ has *infinite* (resp. *finite*) *multiplicity*, if $cp^{-1}(x) \subset P_q$ is infinite (resp. finite). The same definition applies to V_q^{lin} and P_q^{lin}.

Theorem 2.5. *(a) Code points of infinite multiplicity are limit points. Therefore isolated code points have finite multiplicity.*

(b) Conversely, any point (R_0, δ_0) with rational coordinates satisfying the inequality $0 < R_0 < \alpha_q(\delta_0)$ (resp. $0 < R_0 < \alpha_q^{lin}(\delta_0)$) is a code point (resp. linear code point) of infinite multiplicity.

This (actually, slightly weaker) result, seemingly, was first explicitly noticed in [ManMar]. It makes me suspect that *distinguishing between limit and isolated*

code points might be algorithmically undecidable, since in general it is algorithmically impossible to decide, whether a given recursive function takes one of its values at a finite or infinitely many points.

Similarly, one cannot expect *a priori* that limit and isolated code points form two recursively enumerable sets, but this must be true, if α_q is computable: see Theorem 3.4 below.

For completeness, I will reproduce the proof of Theorem 2.5 here. It is based on the same "Spoiling Lemma" that underlies the only known proof of existence of the asymptotic bounds α_q and α_q^{lin}.

Proposition 2.6. (Numerical spoiling.) *If there exists a linear $[n,k,d]_q$-code, then there exist also linear codes with the following parameters:*

(i) $[n+1,k,d]_q$ *(always).*

(ii) $[n-1,k,d-1]_q$ *(if $n>1, k>0$.)*

(iii) $[n-1,k-1,d]_q$ *(if $n>1,\ k>1$)*

In the domain of unstructured codes statements (i) and (ii) remain true, whereas in (iii) one should replace $[n-1,k-1,d]_q$ *by* $[n-1,k',d]_q$ *for some* $k-1\le k'<k$.

For a proof of Proposition 2.6, see e. g. [VlaNoTsfa] (linear codes) and [ManMar] (unstructured codes).

Proof of Theorem 2.5. (a) We first check that if a code point $(R_0,\delta_0)\in\mathbf{Q}^2$ is of infinite multiplicity, then it is a limit point. In fact, let $[n_i,q^{k_i},d_i]$ be an infinite sequence of pairwise distinct code parameters, $i\ge 1$, such that $[k_i]/n_i = R_0, d_i/n_i=\delta_0$ for all i. Then codes with parameters $[n_i+1,q^{k_i},d_i]$ (cf. 2.6 (i)) produce infinitely many pairwise distinct code points converging to (R_0,δ_0).

(b) Now consider a rational point $(R_0,\delta_0)\in\mathbf{Q}^2\cap(0,1)^2$ (unstructured or linear), lying strictly below the respective asymptotic bound. Then there exists a code point (R_1,δ_1) also lying strictly below the asymptotic bound, with $R_1>R_0$ and $\delta_1>\delta_0$, because functions α_q and α_q^{lin} decrease. Hence in the part of U_q (resp. U_q^{lin}) where $R\ge R_1,\delta\ge\delta_1$) there exists an infinite family of pairwise distinct code points (R_i,δ_i), $i\ge 1$, coming from a family of unstructured (resp. linear) $[N_i,K_i,D_i]_q$-codes.

Let $(R_0,\delta_0)=(k/n,d/n)$. Divide N_i by n with a remainder term, i.e. put $N_i=(a_i-1)n+r_i$, $a_i\ge 1, 0\le r_i<n$. Using repeatedly 2.6 (i), spoil the respective $[N_i,K_i,D_i]_q$-code, replacing it by some $[a_in,K_i,D_i]_q$-code. Its code point will have slightly smaller coordinates than the initial (R_i,δ_i), however for N_i large enough, it will remain in the domain $R>R_0,\delta>\delta_0$. Hence we may and will assume from the start that in our sequence of $[N_i,K_i,D_i]_q$-codes all N_i's are divisible by n:

$$N_i=a_in\,. \tag{2.2}$$

In order to derive by spoiling from this sequence another sequence of pairwise distinct codes, all of which have one and the same code point $(R_0,\delta_0)=(k/n,d/n)$,

we will first consider the case of linear codes where the procedure is neater, because $[K_i] = K_i$. Since we have $K_i/N_i > k/n, D_i/N_i > d/n$, we get

$$K_i > a_i k, \quad D_i > a_i d.$$

To complete the proof, it remains to reduce the parameters K_i, D_i to $a_i k, a_i d$ respectively, without reducing $N_i = a_i n$. In the linear case, this is achieved by application of several steps 2.6 (ii), 2.6 (iii), followed by steps 2.6 (i).

In the unstructured case reducing D_i can be done in the same way. It remains to reduce $[K_i]$ to $a_i k$. One application of the step 2.6 (iii) produces K_i' such that either $[K_i'] = [K_i] - 1$, or $[K_i'] = [K_i]$. In the latter case, after restoring N_i to its former value, one must apply 2.6 (iii) again. After a finite number of such substeps, we will finally get $[K_i] - 1$.

QUESTION. Can one find a recursive function $b(n, k, d, q)$ such that if an $[n, k, d]_q$-code is isolated, and $a > b(n, k, d, q)$, there is no code with parameters $[an, ak, ad]_q$?

3 Codes and Computability

In this section, I will discuss computability of two types of closed sets in $[0, 1]^2$: U_q and Γ_q:= the graph of α_q, as well as their versions for linear codes. I will start with the brief summary of basic definitions of [BratWe] in our context.

3.1. Effective Closed Sets. First, we will consider $[0, 1]^2$, U_q and Γ_q as closed subsets in a larger square, say $X := [-1, 2]^2$, with its structure of compact metric space given by $d((a_i), (b_i)) := \max |a_i - b_i|$. The set of *open balls $\mathcal{B}$ with rational centers and radii* in this space has a natural structure of a constructive world (cf. 2.1). Hence we may speak about (recursively) enumerable and decidable subsets of $\mathcal{B}$.

Following [BratWe] and [La], we will consider three types of effectivity of closed subsets $Y \subset X$:

(i) Y is called *recursively enumerable*, if the subset

$$\{I \in \mathcal{B} \,|\, I \cap Y \neq \emptyset\} \subset \mathcal{B} \tag{3.1}$$

is recursively enumerable in $\mathcal{B}$.

(ii) Y is called *co-recursively enumerable*, if the subset

$$\{I \in \mathcal{B} \,|\, \overline{I} \cap Y = \emptyset\} \subset \mathcal{B} \tag{3.2}$$

is recursively enumerable in $\mathcal{B}$ (here $\overline{I}$ is the closure of I).

(iii) Y is called *recursive*, if it is simultaneously recursively enumerable and co-recursively enumerable.

As a direct application of [BratWe] we find:

Proposition 3.2. *The closures $\overline{V}_q$ and $\overline{V}_q^{lin}$ are recursively enumerable.*

Proof. In fact, range of the function *cp* (see 2.3) is dense in $\overline{V}_q$, resp. $\overline{V}_q^{lin}$, and we can apply [BratWe], Corollary 3.13(1)(d).

3.3. Problem of Computability of the Asymptotic Bound. Referring to the Corollary 7.3 of [Brat], we will call α_q (resp. α_q^{lin}) *computable*, if its graph Γ_q (resp. Γ_q^{lin}) is co-recursively enumerable.

Theorem 3.4. *Assume that α_q is computable. Then each of the following sets is recursively enumerable:*

(a) Code points lying strictly below the asymptotic bound.

(b) Isolated code points.

The same is true for linear codes, if α_q^{lin} is computable.

Proof. We start with the following remark. Choose any integer $N \geq 1$ and consider the set $\Gamma_q^{(N)}$ which is the union of closed balls of the form

$$\overline{I} = \left[\frac{p}{N}, \frac{p+1}{N}\right] \times \left[\frac{p}{N}, \frac{p+1}{N}\right] \subset X \tag{3.3}$$

satisfying $p \in \mathbf{N}$, $\overline{I} \cap \Gamma_q \neq \emptyset$. Then we have:

(i) *The boundary of $\Gamma_q^{(N)}$ consists of two vertical (parallel to the R-axis) segments at the ends and two piecewise linear connected closed curves: $\Gamma_{q+}^{(N)}$ lying above $\Gamma_{q-}^{(N)}$.*

(ii) The distance of any point $x \in \Gamma_{q-}^{(N)}$ to $\Gamma_{q+}^{(N)}$ does not exceed $2/N$, and similarly with $+$ and $-$ reversed.

Let us call an *N-strip* any connected closed set satisfying these conditions.

Now, assuming α_q (resp. α_q^{lin}) computable, that is, Γ_q co-recursively enumerable, choose N and run the algorithm generating in some order all rational closed balls $\overline{I}$ such that $\overline{I} \cap \Gamma_q = \emptyset$. Wait until their subset consisting of balls of the form (3.3) covers the whole square $[0,1]^2$ with exception of a set whose closure is an N-strip. This strip will then be an approximation to Γ_q (resp. Γ_q^{lin}) containing the respective graph in the subset of its inner points.

Run parallelly an algorithm generating all code points and divide each partial list of code points into three parts depending on N: points lying below $\Gamma_q^{(N)}$, above $\Gamma_q^{(N)}$, and inside $\Gamma_q^{(N)}$.

When N grows, the growing first and second parts respectively will recursively enumerate code points below and above the asymptotic bound.

Remark. This reasoning also shows, in accordance with [Brat], that if we assume Γ_q only co-recursively enumerable, it will be automatically recursively enumerable and therefore recursive.

Theorem 3.5. *Assume that U_q is recursive in the sense of 3.1(iii). Then α_q is computable. The similar statement holds for linear codes.*

Proof. Consider first a closed ball $\overline{I}$ as in (3.3) that intersects U_q whereas its inner part I does not intersect U_q. A contemplation will convince the reader that the left lower boundary point of this "ball" (a square in the Euclidean metric) is precisely the intersection point $\overline{I} \cap \Gamma_q$. Call such a ball *an exceptional N-ball.* Since α_q is decreasing, we have

(a) *Each horizontal strip $p/N \le R \le (p+1)/N$ and each vertical strip $q/N \le \delta \le (q+1)/N$ can contain no more than one exceptional N-ball.*

(b) If one exceptional N-ball lies to the right of another one, then it also lies lower than that one.

Generally, call a set of N-balls N-*admissible*, if it satisfies (a) and (b).

Now, assuming U_q recursive and having chosen N, we can run parallelly two algorithms: one generating closed balls (3.3) non-intersecting U_q and another, generating open balls (3.3) intersecting U_q. Run them until all N-balls are generated, with a possible exception of an N-admissible subset $X_q^{(N)}$, then stop generation. Let $U_{q+}^{(N)}$ be the union of all balls generated by the first algorithm, and $U_{q-}^{(N)}$ the union of all balls generated by the second algorithm.

Look through all the balls in $X_q^{(N)}$ in turn. If there are elements in it whose closure does not intersect the closure of $U_{q-}^{(N)}$, delete them from $X_q^{(N)}$ and put it into $U_{q+}^{(N)}$. Similarly, if there are elements in it whose closure does not intersect (initial) $U_{q+}^{(N)}$, delete them from $X_q^{(N)}$ and put them into $U_{q-}^{(N)}$.

Keep the old notations $U_{q-}^{(N)}$, $U_{q+}^{(N)}$, $X_q^{(N)}$ for these amended sets.

Now, the union of the lower boundary of $U_{q+}^{(N)}$ and the upper boundary of $U_{q-}^{(N)}$ will approximate Γ_q from two sides, with error not exceeding N^{-1}. (Here a "boundary" means the respective set of boundary squares).

Clearly, this reasoning shows also also computability of α_q in the sense of 3.3.

References

[BaFo] Barg, A., Forney, G.D.: Random codes: minimum distances and error exponents. IEEE Transactions on Information Theory 48(9), 2568–2573 (2002)

[Brat] Brattka, V.: Plottable real functions and the computable graph theorem. SIAM J. Comput. 38(1), 303–328 (2008)

[BratMiNi] Brattka, V., Miller, J.S., Nies, A.: Randomness and differentiability. arXiv:1104.4456

[BratPre] Brattka, V., Preser, G.: Computability on subsets of metric spaces. Theoretical Computer Science 305, 43–76 (2003)

[BratWe] Brattka, V., Weihrauch, K.: Computability on subsets of Euclidean space I: closed and compact subsets. Theoretical Computer Science 219, 65–93 (1999)

[BravC] Braverman, M., Cook, S.: Computing over the reals: foundations for scientific computing. Notices AMS 53(3), 318–329 (2006)

[BravYa] Braverman, M., Yampolsky, M.: Computability of Julia sets. Moscow Math. Journ. 8(2), 185–231 (2008)

[CaHeKhWa] Calude, C.S., Hertling, P., Khoussainov, B., Wang, Y.: Recursively enumerable reals and Chaitin Ω numbers. Theor. Comp. Sci. 255, 125–149 (2001)

[La] Lacombe, D.: Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles., I–III. C. R. Ac. Sci. Paris 240, 2478–2480, 241, 13–14, 151–153 (1955)

[Man1] Manin, Y.I.: What is the maximum number of points on a curve over $\mathbf{F}_2$? J. Fac. Sci. Tokyo, IA 28, 715–720 (1981)

[Man2] Manin, Y.I.: Renormalization and computation I: motivation and background. Preprint math. QA/0904.4921

[Man3] Manin, Y.I.: Renormalization and Computation II: Time Cut-off and the Halting Problem. Preprint math. QA/0908.3430

[ManMar] Manin, Y.I., Marcolli, M.: Error-correcting codes and phase transitions. arXiv:0910.5135

[ManVla] Manin, Y.I., Vladut, S.G.: Linear codes and modular curves. J. Soviet Math. 30, 2611–2643 (1985)

[TsfaVla] Tsfasman, M.A., Vladut, S.G.: Algebraic-geometric Codes. Kluwer (1991)

[VlaNoTsfa] Vladut, S.G., Nogin, D.Y., Tsfasman, M.A.: Algebraic Geometric Codes: Basic Notions. Mathematical Surveys and Monographs, vol. 139. American Mathematical Society, Providence (2007)

Some Transfinite Generalisations of Gödel's Incompleteness Theorem

Jacques Patarin

University of Versailles
45 Avenue des États-Unis, 78035 Versailles Cedex, France
jacques.patarin@prism.uvsq.fr

Abstract. Gödel's incompleteness theorem can be seen as a limitation result of usual computing theory: it does not exist a (finite) software that takes as input a first order formula on the integers and decides (after a finite number of computations and always with a right answer) whether this formula is true or false. There are also many other limitations of usual computing theory that can be seen as generalisations of Gödel incompleteness theorem: for example the halting problem, Rice theorem, etc. In this paper, we will study what happens when we consider more powerful computing devices: these "transfinite devices" will be able to perform α classical computations and to use α bits of memory, where α is a fixed infinite cardinal. For example, $\alpha = \aleph_0$ (the countable cardinal, i.e. the cardinal of $\mathbb{N}$), or $\alpha = \mathfrak{C}$ (the cardinal of $\mathbb{R}$). We will see that for these "transfinite devices" almost all Gödel's limitations results have relatively simple generalisations.

1 Introduction

Gödel's famous incompleteness theorem was first presented on October 7, 1930, at the first international conference of mathematics philosophy, at Königsberg. This result can be seen as a limitation result of usual computing theory: it does not exist a (finite) software that takes as input a first order formula on the integers and decides (after a finite number of computations and always with a right answer) whether this formula is true or false. In 1930 no real computer existed yet, but the mathematical analysis of the functions that can be effectively computed with (finite) software (i.e. "recursive functions") had began. Gödel was studying sets of axioms for which there is an—effective, finite, recursive—computing way to know if a given formula was a member of these axioms or not.

What will happen if we consider more powerful computing devices? For example, if we include in the set of axioms all first order formulas that are true in $\mathbb{N}$ (with the standard interpretation of addition and multiplication) we will obtain a complete set of axioms (i.e. with no undecidable and contradictory formulas); however, it is not possible with a classical software to know if a given formula is one of the axioms or not.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 183–193, 2012.

In this paper we will study what happens when we use "transfinite software", i.e. software that can be run on "transfinite computers", generalised computers that can perform α classical computations and use α bits of memory, where α is a fixed infinite cardinal. These transfinite computers are able to compute more than classical computers, but, are they limited by "transfinite questions" that can be seen as generalisations of classical computations questions? In fact, as we will see, it is possible to generalise almost all classical results within this framework. Such generalisation is not totally new. In [6] and in some references mentioned in [6], problems linked to "Totality, Knowledge and Truth", and "Incompleteness" are mentioned, and it is clearly stated that some limitation results can be generalised beyond the classical theory of computation; a relativised version of incompleteness was proved in [3]. It seems, however, that an explicit description of the main limitation theorems presented in the framework of "transfinite computers" has not been done yet.

In [4] it is proved that any Turing machine that uses only a finite computational space for every input cannot solve any undecidable problem even when it runs in accelerated mode (unlike as in this paper where the memory will be infinite). A natural continuation of this work, that we hope to obtain in the near future, is the generalisation of the results of [2] for "transfinite software".

2 A Transfinite Computing Model

2.1 General Remarks

We will use the transfinite computing model described in [14]. To make this paper self contained we will explain in this section the model.

It is also important to notice that our limitation proofs and results below are very stable and generally will not depend on the chosen transfinite computing model, as long as the model is reasonably natural and uses sets (working on classes instead of sets may involve a specific analysis which will not be presented in this paper).

2.2 Transfinite Computations

Let α be a fixed infinite cardinal. For example $\alpha = \aleph_0$ (the countable cardinal, i.e. the cardinal of $\mathbb{N}$), or $\alpha = \mathfrak{C}$ (the continuum, i.e. cardinal of $\mathbb{R}$). In "α-software" we use "transfinite computers" able to perform α computations with α bits of memory. It is possible to describe precisely this model of computations, see [15], [16], [7], [9], [10], for example. The general idea is to follow a generalisation of the Church's Thesis: as soon as a computation will be clearly feasible with α bits of memory and α computations, we will include it in the model. Moreover, the results of this paper will be very stable with respect to small changes in the infinite computation model.

Readers familiar with Ordinal Turing Machines, (OTM), with tapes whose cells are indexed by ordinals, as described in [9], can just go directly to section 3. We will speak of "α-programs" or "α-softwares". We can assume that

the memory is separated in 4 zones of bits: the input memory, the program memory, the variables of computation memory, and the output memory. Without loss of generality we can assume that the input memory is made of 1, or 2 (or more but $\leq \alpha$) inputs of α bits. The program memory contains a well ordered set of α elementary operations. Thanks to the fact that the program memory is well ordered, we can know at each "time" of the computation which is the next operation to perform. The word "time" is of course here a generalised word: it means that when any set of operations has been performed, we know precisely what is the next operation to be performed. More precisely than "time", it is the succession of some ordinals that we will use. To each operation T at a certain place in the program we will associate an ordinal β, so we can say that T is the operation number β, or of position β. Each elementary operation can be of two kinds: simple, or GOTO. A simple elementary operation is a classic operation present in any computer language (such as C, on two words of 64 bits, for example) such that these two words are chosen at the addresses a and b of the memory and the result is stored at the address c of the memory. Here a and b can be addresses of the input memory, or of the variables of computation memory, c can be an address of the variables of computation memory, or of the output memory; a, b, and c are addresses of at most α bits and are associated with the current operation. So each instruction of the program (operation and its position or number) can memorise a, b, c, and the operation to be performed. Of course, it is possible to use any other classical computer language instead of the C language, or to use words of 32 bits (or another length) instead of 64 bits. This will not change the set of functions that we can compute. A special instruction is the "stop instruction". When this instruction is performed, the program stops and the output of the program is the value stored in the α bits of the output memory. The GOTO operation is an operation of the form (if $X = k$) then GOTO β, where β is an ordinal. Thus this GOTO instruction says that the next instruction to be performed is the instruction number β (or of position β), if a variable X of α bits is equal to the value k of α bits. (Note that β can be any ordinal smaller than the ordinal of the current GOTO instruction performed.) If $X \neq k$, to determine the next instruction we will follow, as for the simple instructions, the usual order of the ordinals of the instructions. It is also possible to describe our model of transfinite computations with generalised Turing machines.

2.3 Coding the Instruction Ordering

In the α bits of the program memory zone there are various simple ways to describe the ordering (well ordering) of the instructions. Let us give here an example for $\alpha = \aleph_0$. (It is easy to generalise this example for any cardinal α.) Let P be an α program. By definition, we will call "ordinal of P" the ordinal of the (well ordered) set of all instructions of P. For example, if $\alpha = \aleph_0$, this ordinal may be ω, or ω^3. A countable ordinal can be described as a good ordering on $\mathbb{N}$. So each countable ordinal can be written as a set of $\aleph_0$ integers: for each integer n, we will give the list of all the integers m such that $m < n$ for

this ordering. We need for this $\leq \aleph_0 \times \aleph_0$ bits. The "infinite processor" can find the first instruction (no instruction is strictly smaller), and then, at each step, it can check all the integers in order to find the next instruction to be performed.

Remark. A classical result on ordinals says that the set of countable ordinals has cardinal $\aleph_1$ (i.e. the smallest non countable infinite cardinal). We know that $\aleph_1 \leq \mathfrak{C}$. (However we do not know if $\aleph_1 = \mathfrak{C}$ or not, this is the famous undecidable problem called "the continuum hypothesis".) Moreover each real number can be given by $\aleph_0$ bits. Therefore, each countable ordinal can be given by $\aleph_0$ bits. This is what we do here for the ordinal of program P. The method used for $\alpha = \aleph_0$ can also be extended to any infinite cardinal α since, adopting the axiom of choice, for any infinite cardinal α, we have $\alpha^2 = \alpha$.

Example. The function $x \rightarrow x^2$ on $\mathbb{Q}$ is a one way function in the model of infinite computation of [17]. In our model of infinite computations, this function however is not a one way function. In order to find a rational (or a real) x such that $x^2 = y$ with $\aleph_0$ computations and $\aleph_0$ bits of memory, we can, for example, find all the bits of x, one by one. If we know that x is a rational number, then we can also try all the rational numbers one by one (card $\mathbb{Q} = \aleph_0$), square them, and check whether we get y. Here again we need "only" $\aleph_0$ computations and $\aleph_0$ bits of memory.

Remark. On classical computers bits can have the value 0, or the value 1. In our model of computation, it is possible to assume that the values can be 0, 1, or "not fixed". The value "not fixed" will be obtained for example when the bit has flipped from 0 to 1 and from 1 to 0 infinitely many times, without being fixed to 0 or 1. However, it is possible to prove that if this value "not fixed" is changed to 0 (or 1), the infinite model of computation will be the same (i.e. we will be able to compute exactly the same functions); in this case the model may be slightly less natural. (A variable B can be at $11 \ldots 1 \ldots$ with an infinity of 1 if and only if a bit b has changed an infinity of times its value.)

Remark. As pointed out by an anonymous referee of WTCS2012, our transfinite computing model is in fact similar to admissible recursion on cardinals (which is equivalent to running ordinal Turing machines). Admissible recursion has been well-developed since the 60s in work of Platek (1966), Kripke (1964) and Sacks (cf [12], p.443).

3 α-Recursive Sets, α-Recursively Enumerable Sets

We start with a few definitions.

Definition 1. We say that an α-software **stops** or **gives the output after** α **computations** when the α-software stops after performing at most α computations.

Definition 2. We denote by $I_\alpha = \{0,1\}^\alpha$ the set of all sequences of α bits. Therefore $I_{\aleph_0}$ can be identified with the set $\mathbb{R}$ of all the real numbers, or with $[0,1]$ for example.

Definition 3. Let A be a subset of I_α. We say that A is α-**recursive** if there exists at least one α-software P such that when we give $n \in I_\alpha$ as input to P, then P will be able to answer after at most $\leq \alpha$ operations if $n \in A$ or $n \notin A$.

Definition 4. We say that A is α-**recursively enumerable** if there exists at least one α-software P such that when we give $n \in I_\alpha$ as input of P:

1. if $n \in A$, then P will be able to answer $n \in A$ after at most α operations.
2. if $n \notin A$, then P does not answer after α operations, or P will answer $n \notin A$.

Definition 5. Let f be an application $I_\alpha \to I_\alpha$. We say that f is α-**recursive** if there exists at least one α-software P such that for all $n \in I_\alpha$, when n is given as input to P, P will give the output $f(n)$ after performing at most α computations.

Remark. There are α^α applications from I_α to I_α, and the number of α-softwares is $\leq \alpha$. Since $\alpha^\alpha \geq 2^\alpha > \alpha$ (Cantor Theorem), it follows that some applications are neither α-recursive nor α-recursively enumerable.

Definition 6. Let α be a cardinal. Put $I_{limit\ \alpha} = \cup_{\beta<\alpha} I_\beta$. We define an α **-limit-software** as a α-software such that:

1. The inputs are the elements of $I_{limit\ \alpha}$.
2. The variables of computation memory are in $I_{limit\ \alpha}$.
3. The program memory is in I_β with $\beta < \alpha$.
4. The input is an element of $I_{limit\ \alpha}$.

Definition 7. Let $A \subset I_{limit\ \alpha}$. We say that A is α-**limit-recursive** if there exists at least one α-limit-software P such that when we give $n \in I_{limit\ \alpha}$ as input of P, P will be able to answer after at most β operations, $|\beta| < |\alpha|$, if $n \in A$ or $n \notin A$.

Definition 8. Let $A \subset I_{limit\ \alpha}$. We will say that A is α**-limit recursively-enumerable** if and only of it exists at least one α-limit-software P such that: when we give $n \in I_{limit\ \alpha}$ as input of P,

• if $n \in A$ then P will be able to answer after at most β operations, $|\beta| < |\alpha|$, that $n \in A$.
• if $n \notin A$ then P will not answer or P will $n \notin A$.

Remark

1. The above definitions are generalisations of the classical definition of recursiveness, i.e. $\aleph_0$-limit-recursive = recursive (usual meaning) and $\aleph_0$-limit-recursively-enumerable = recursively enumerable (usual meaning).

2. The definitions of α-limit-recursiveness may be interesting when α is a cardinal with no predecessor (as $\aleph_0$), i.e. when there is no cardinal $\beta < \alpha$ such that α is the smallest cardinal $> \beta$, because if α has a predecessor β, then α-limit recursiveness is simply β recursively.

4 A Generalisation of "Recursively Enumerable and Not Recursive Sets"

To achieve the main aim of this paper, i.e. to show that that most of the classical limitation results of logic can be generalised in the model of transfinite computations, with almost the same proofs as in the case of classical case, we will follow here one of the classical ways to obtain such limitation results (as in [11] or [13]). Of course, one can possibly obtain the same results following other proofs.

4.1 α-Code of a α-Software

To each α-software T we can associate in an injective and "simple" way an element of I_α, called its α-code, and denoted $\lceil T \rceil$. By "simple" we mean that there exists an α-software that takes $\lceil T \rceil$ as input, and then produces the sequence of α instructions of T.

Example: If $\alpha = \aleph_0$, the indices of the instructions of T are countable ordinals, and the set of these indices is countable. They form a well-ordering and this can be seen as a well-ordering on $\mathbb{N}$. Such a well-ordering on $\mathbb{N}$ can be described as follows: to each natural integer we associate the of integers that are smaller than the given integer (for the well-ordering). There are $\aleph_0$ such integers. Then for each elementary instruction, we can associate 2 or 3 real numbers (for example the instruction "if $(X = K)$ then GOTO β"). Thus we can associate to such a software T, an injective and "simple" application from $\mathbb{N}$ to $\mathcal{P}(\mathbb{N}) \times \mathbb{R}^3$. Let B be the set of applications from $\mathbb{N}$ to $\mathcal{P}(\mathbb{N}) \times \mathbb{R}^3$. $|B| = |\mathbb{R}|$; there exist "simple" bijections from B to $[0, 1]$. So to each $\aleph_0$-software T, we can associate in an injective and "simple" way an element of $I_{\aleph_0}$ which is its code.

Remark. The above method works for any $\alpha \geq \aleph_0$.

4.2 Software Simulation

If B is an α-software and x an element of I_α, we denote by $B(x)$ the result of software B when x is the input: i.e, the value of the output memory (it is also an element of I_α) when the software stops after $\leq \alpha$ operations.

There exists an α-software P which, when it is given $x \in I_\alpha$ as input:

1. P "finds" the α-software X such that $\lceil X \rceil = x$ in case such an α-software X exists. This comes from the fact that the coding is "simple" (cf. above).
2. P executes the same instructions on x as X would execute with x as input. Thus, for all $x \in I_\alpha$, $P(x) = X(x)$ provided there exists an α-software X such that $\lceil X \rceil = x$.

4.3 The Basic Theorem

Theorem 1. *There exists $A \subset I_\alpha$, such that A is α-recursively enumerable, but A is not α-recursive.*

Proof: Let P be the α-software previously defined such that $P(x) = X(x)$, whenever there exists an α-software X with code x. Let

$$A = \{x \in I_\alpha \mid P(x) \text{ is computed in } \leq \alpha \text{ computations.}\}$$

1. Since A is defined by the α-software P, A is α-recursively enumerable.
2. Assume that A is α-recursive and q is the code of an α-software Q such that:

$$x \notin A \Leftrightarrow Q(x) \text{ is computed in } \leq \alpha \text{ computations.}$$

Then

$$q \in A \Leftrightarrow P(q) \text{ is computed in } \leq \alpha \text{ computations} \quad (\text{by definition of } A)$$

i.e.

$$q \in A \Leftrightarrow Q(q) \text{ is computed in } \leq \alpha \text{ computations} \quad (\text{by definition of } P)$$

i.e.

$$q \in A \Leftrightarrow q \notin A \quad (\text{by definition of } Q).$$

This is not possible. Thus A is not α-recursive. □

5 The Decision Problem, the Halting Problem and Ordinal Length of Computations for α-Softwares

We generalise the undecidability of the above problems for α-softwares.

5.1 The Decision Problem

Theorem 2. *There is no general algorithm, programmable with α-software, which could, using always $\leq \alpha$ computations, to decide whether a mathematical proposition on elements of I_α is true or not.*

Proof: It is enough to consider all the propositions of the form $n \in A$, where $n \in I_\alpha$, and A is the set defined in Theorem 1 above. Since A is not α recursive, there exists no α-software which, when applied to one of these propositions $n \in A$ can decide, using $\leq \alpha$ computations, whether this proposition is true or false. □

Remark

1. These mathematical propositions can be written with quantifiers $\forall$, $\exists$, the usual logic symbol and the operators $+$, $-$, $\times$, $\div$ and with $\leq \alpha$ elementary finite formulas. We then get a generalisation of Gödel's Incompleteness Theorem. (We just have to write the α-software with such formulas, which are generalisations of first order classical formulas with α characters, which is always possible).
2. Some properties that are true on sequences of α bits are lost if we are limited to α computations and α bits of memory, for any infinite cardinal α.

5.2 The Halting Problem

Theorem 3. *There exists no α-software which can decide with $\leq \alpha$ computations whether an arbitrary α-software will stop or not in $\leq \alpha$ operations.*

Proof: If such α-software existed, then we could use it to write an α-software which, when presented with an $x \in I_\alpha$ as input could decide in $\leq \alpha$ operations whether $P(x)$ stops after $\leq \alpha$ computed or not. But A is not α recursive, thus such an α-software does not exist. □

5.3 Ordinal Length of Computations for α-Softwares

Theorem 4. *There exists no general α-software taking as input the code of a program T which stops in $\leq \alpha$ computations, and gives as output an ordinal ω_α such that the cardinal of ω_α is $\leq \alpha$ and the ordinal of the number of computations performed before T stops is $\leq \omega_\alpha$.*

Proof: If such a program exists, we could know with $\leq \alpha$ computations if $n \in A$ or $n \notin A$. This is in contradiction with the fact that A is not α-recursive (A comes from Theorem 1). In that case, it would be sufficient to stop after ω_α computations to conclude that the program does not answer in $\leq \alpha$ computations. □

6 The Fixed Point Theorem on α-Softwares

Let $z, x, y \in I_\alpha$, such that there exists an α-software Z with code z and two entries: x and y. We denote $z[x, y]$ the output of the software Z on entries x and y when this software stops in $\leq \alpha$ computations.

Remark. If Z does not stop after $\leq \alpha$ computations, we can consider that $z[x, y]$ is the information "z does not stop" after $\leq \alpha$ computations.

Theorem 5 (Iteration Theorem). *There is an α-recursive application of two variables $s(x, y)$ such that:*

$$\forall z, x, y \in I_\alpha,\ z[x, y] = s(z, y)[x].$$

Proof: We consider the α-software s that performs the following operations when it is given z and y as inputs:

1. Finds the sequence of instructions of the α-software Z with code z. (This is possible since the coding is "simple").
2. Computes the code of an α-software which on input $x \in I_\alpha$ simulates Z on the inputs x and y. (Again this is possible since the coding is "simple").

Then this α-software computes $s(z, y)$ such that

$$\forall z, x, y \in I_\alpha,\ z[x, y] = s(z, y)[x].$$

□

Theorem 6 (Fixed point theorem on α-softwares). *For every α-recursive application h, there exists $e \in I_\alpha$ such that:*

$$\forall x \in \mathbb{N},\ e[x] = h(e)[x].$$

Remark. Every α-software can be written using a program with a single input ($\alpha^2 = \alpha$ since α is an infinite cardinal). Thus the fixed point theorem on α-softwares can be written in the form: If "h is an α-recursive application, there exists always an α-software with code e and an α-software with code $h(e)$ which on any input $x \in I_\alpha$ gives the same output (and does not give any output).

Indeed, let $f(x, y) = h(y)[x]$. Since $s(z, y)$ is α-recursive, there exists d the code of an α-software with computes $f(x, s(y, y))$. For all $x \in I_\alpha$, et $\forall y \in I_\alpha$, we have:

$$f(x, s(y, y)) = \begin{cases} d[x, y], & \text{by definition of } d, \\ d(d, y)[x], & \text{by definition of } s\,. \end{cases}$$

Let $e = d(d, d)$. With $y = d$, we get that for all $n \in I_\alpha$, $f(x, e) = s(d, d)[x] = e[x]$. Thus (by definition of f), $\forall x \in I_\alpha$, $h(e)[x] = e[x]$. □

7 Rice Theorem on α-Softwares

A function f from D_f to I_α, where $D_f \subset I_\alpha$, is called **α-recursive semi-function** if there exists an α-software which computes $f(x)$ when given an input $x \in D_f$ in $\leq \alpha$ computations, and does not answer in $\leq \alpha$ computations when it is given $x \notin D_f$.

Theorem 7 (Rice theorem on α-softwares). *Let F be a non-empty proper subset of all α-recur- sive semi-functions. Then the set*

$$A = \{n \in I_\alpha \mid n \text{ is the code of an } \alpha - \text{recursive semi-function of } F\}$$

is not α-recursive.

Proof: By definition of F, $A \neq \emptyset$ and $A \neq I_\alpha$. Let a and $\bar{a}$ be two elements of I_α such that $a \in A$ and $\bar{a} \notin A$. We set

$$h : \begin{cases} I_\alpha & \to I_\alpha \\ x \in A & \mapsto \bar{a} \\ x \notin A & \mapsto a \end{cases}$$

Suppose that A α is α-recursive. Then, h is a α-recursive application. It follows, from the fixed point theorem, that there exists $e \in \mathbb{N}$ such that the α-programs coded by e and $h(e)$ compute the same semi-function. So $e \in A \Leftrightarrow h(e) \in A$ (by definition of A and e). But by definition of h, $e \in A \Leftrightarrow h(e) \notin A$. Thus $h(e) \in A \Leftrightarrow h(e) \notin A$, a contradiction. This shows that A is not α-recursive. □

This generalised Rice theorem shows that there exists no α-software which decides:

1. If two α-softwares compute the same function. (Choose a singleton for F.)
2. If an α-software will always answer 0 on any input. (Choose F that contains only the null function.)
3. If an α-software will always give an answer. (Choose for F the set of recursive semi-functions defined on I_α.)
4. If an α-software will always return values in a given subset B. (Choose for F the set of semi-functions whose output is in B.)

This generalised Rice Theorem shows that the problem of "debugging" an α-software, or the understanding of what a α-software is doing, generally uses more than α computations.

8 Conclusion

We have shown that most of the logic limitation results proved in the classical theory of computation can be generalised to the case when the computing devices are able to perform α computations and use α bits of memory, where α is a given fixed cardinal. It is expected that some recent results of limitations, such as those of [2], can also be generalised in this framework. This can be the subject of further work.

Acknowledgements. I would like to thank C. Calude for his help in the final redaction of this paper, and an anonymous referee of WTCS2012 for pointing out the previous work of Platek, Kripke and Sacks on admissible recursion on cardinals (which is equivalent to running ordinal Turing machines).

References

1. Calude, C., Jürgensen, H., Zimand, M.: Is independence an exception? Appl. Math. Comput. 66, 63–76 (1994)
2. Calude, C.S., Jürgensen, H.: Is complexity a source of incompletness? Advances in Applied Mathematics 35, 1–15 (2005)
3. Calude, C.S., Rudeanu, S.: Proving as a computable procedure. Fundamenta Informaticae 64(1-4), 43–52 (2005)
4. Calude, C.S., Staiger, L.: A Note on Accelerated Turing Machines. Math. Struct. in Comp. Sciences 20, 1011–1017 (2010)
5. Graham, L., Kantor, J.M.: Naming Infinity. The Belknap Press of Harward University Press (2009)
6. Grim, P.: The Incomplete Universe, Totality, Knowledge and Truth. A Bradford Book, The MIT Press (1991)
7. Hamkins, J.D., Lewis, A.: Infinite time Turing machines. Journal of Symbolic Logic 65(2), 567–604 (2000)
8. Knuth, D.: Surreal Numbers: How two ex-student turned to pure mathematics and found total happiness. Addison-Wesley (1974)
9. Koepke, P.: Turing Computations on Ordinals. The Bulletin of Symbolic Logic 11(3), 377–397 (2005)

10. Koepke, P., Koerwien, M.: Ordinal Computations. Mathematical Structures in Computer Science 16(5), 867–884 (2006)
11. Mendelson: Introduction to Mathematical Logic. Wadsworth and Brooks/Cole Advanced Books of Software
12. Odifreddi, P.G.: Classical Recursion Theory. Elsevier (1989)
13. Patarin, J.: Logique Mathématique et Théorie des Ensembles. Polycopié de cours publié par l'École Centrale de Paris puis l'Université de Versailles-Saint-Quentin (1991)
14. Patarin, J.: Transfinite Cryptography. In: HyperNet 2010, Tokyo (2010)
15. Syropoulos, A.: Hypercomputation. Springer, Heidelberg (2008)
16. Welch, P.D.: Turing Unbound: Transfinite Computation. In: Cooper, S.B., Löwe, B., Sorbi, A. (eds.) CiE 2007. LNCS, vol. 4497, pp. 768–780. Springer, Heidelberg (2007)
17. Woodruff, D., van Dijk, M.: Cryptography in an Unbounded Computational Model. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 149–164. Springer, Heidelberg (2002)

On Oscillation-Free Chaitin h-Random Sequences

Ludwig Staiger

Institut für Informatik, Martin-Luther-Universität Halle-Wittenberg
von-Seckendorff-Platz 1, D-06099 Halle, Germany
staiger@informatik.uni-halle.de

Abstract. The present paper generalises results by Tadaki [12] and Calude et al. [1] on oscillation-free partially random infinite strings. Moreover, it shows that oscillation-free partial Chaitin randomness can be separated from oscillation-free partial strong Martin-Löf randomness by Π^0_1-definable sets of infinite strings.

In the papers [11] and [2] several relaxations of randomness were defined. Subsequently, in [8] these were shown to be essentially different. The variants of partial randomness were characterised by different means such as Martin-Löf tests [11,2], Solovay tests [11,8] and prefix [11] or a priori complexity [2]. Using description complexity partial randomness of an infinite string ξ was defined by linear lower bounds on the complexity of the n-length prefix $\xi \restriction n$, that is, an infinite string was referred to as ε-random provided the complexity of $\xi \restriction n$ was lower bounded by $\varepsilon \cdot n - O(1)$. In general, the mentioned papers did not require an upper bound on the complexity, except for [11] where an asymptotic upper bound was considered.

For the case of a priori complexity, the papers [9,7] gave a description of infinite oscillation free ε-random strings where the upper complexity bound matches the lower bound up to an additive constant. For the case of prefix complexity the construction of similar infinite strings was accomplished in [12,1]. The construction in [1] uses ε-universal prefix machines. Here it was observed in Theorem 6 that there are different (inequivalent) types of ε-universal machines.

In recent publications, based on Hausdorff's original paper [5] the concept of partial randomness was refined to functions of the logarithmic scale [6] or to more general gauge functions [10]. Here we showed that for a priori complexity and computable gauge functions $h : \mathbb{Q} \to \mathbb{R}$ there are oscillation-free h-random infinite strings.

The aim of the present paper is to show that, similarly to the results of [10], also in the case of prefix complexity one can refine ε-randomness to oscillation-free h-randomness. Moreover, our investigations reveal the reason of the paradox of [1, Theorem 6].

Cast into the language of gauge functions (cf. [4,10]) the papers [12,1] considered only the scale $h(t) = t^\varepsilon$, $\varepsilon \in (0, 1)$ computable, which results in complexity bounds of the form $\varepsilon \cdot n + O(1)$. The present paper refines this scale to a much larger class of gauge functions including also non-computable ones.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 194–202, 2012.

The paper is organised as follows. First we introduce some notation and consider the concept of gauge functions. In the second section we investigate, for gauge functions h, h-universal machines as a generalisation of the ε-universal machines of [1]. In this section we also explain the paradox of [1, Theorem 6]. Then, in Section 3, we continue with further generalising results of [12,1] to oscillation-free h-randomness for prefix complexity, and in the last part we show that oscillation-free h-random infinite sequences w.r.t. a priori complexity can be separated by Π^0_1-definable sets from oscillation-free h-random infinite sequences w.r.t. prefix complexity.

1 Notation and Preliminaries

In this section we introduce the notation used throughout the paper. By $\mathbb{N} = \{0, 1, 2, \ldots\}$ we denote the set of natural numbers and by $\mathbb{Q}$ the set of rational numbers. Let $X = \{0, 1, \ldots, r-1\}$ be an alphabet of cardinality $|X| = r \geq 2$. By X^* we denote the set of finite words on X, including the *empty word* e, and X^ω is the set of infinite strings (ω-words) over X. Subsets of X^* will be referred to as *languages* and subsets of X^ω as *ω-languages*.

For $w \in X^*$ and $\eta \in X^* \cup X^\omega$ let $w \cdot \eta$ be their *concatenation*. This concatenation product extends in an obvious way to subsets $W \subseteq X^*$ and $B \subseteq X^* \cup X^\omega$.

We denote by $|w|$ the *length* of the word $w \in X^*$ and $\mathbf{pref}(B)$ is the set of all finite prefixes of strings in $B \subseteq X^* \cup X^\omega$. We shall abbreviate $w \in \mathbf{pref}(\eta)$ ($\eta \in X^* \cup X^\omega$) by $w \sqsubseteq \eta$, and $\eta \upharpoonright n$ is the n-length prefix of η provided $|\eta| \geq n$. A language $W \subseteq X^*$ is referred to as *prefix-free* if $w \sqsubseteq v$ and $w, v \in W$ imply $w = v$.

For a computable domain $\mathcal{D}$, such as $\mathbb{N}$, $\mathbb{Q}$ or X^*, we refer to a function $f : \mathcal{D} \to \mathbb{R}$ as *left computable* (or *approximable from below*) provided the set $\{(d, q) : d \in \mathcal{D} \wedge q \in \mathbb{Q} \wedge q < f(d)\}$ is computably enumerable. Accordingly, a function $f : \mathcal{D} \to \mathbb{R}$ is called *right computable* (or *approximable from above*) if the set $\{(d, q) : d \in \mathcal{D} \wedge q \in \mathbb{Q} \wedge q > f(d)\}$ is computably enumerable, and f is *computable* if f is right and left computable. Accordingly, a real number $\alpha \in \mathbb{R}$ as left computable, right computable or computable provided the constant function $f_\alpha(t) = \alpha$ is left computable, right computable or computable, respectively.

1.1 Gauge Functions

A function $h : (0, \infty) \to (0, \infty)$ is referred to as a *gauge function* provided h is right continuous and non-decreasing.[1] If not stated otherwise, we will always assume that $\lim_{t \to 0} h(t) = 0$. As in [10] with a gauge function we associate a *modulus function* $g : \mathbb{N} \to \mathbb{N}$ which, roughly speaking, satisfies $h(r^{-g(n)}) \approx r^{-n}$ or, more precisely, $|-\log_r h(r^{-g(n)}) - n| = O(1)$.

We may define the modulus as follows

[1] In fact, since we are only interested in the values $h(r^{-n})$, $n \in \mathbb{N}$, the requirement of right continuity is just to conform with the usual meaning (cf. [4]).

Definition 1. $g(n) := \sup\{m : m \in \mathbb{N} \wedge r^{-n} < h(r^{-m})\}$

Here we use the convention $\sup \emptyset = 0$. Then we have

$$h(r^{-g(n)}) > r^{-n} \text{ if } g(n) \neq 0\,. \tag{1}$$

Moreover, the following holds true.

Lemma 1. *If for all $j \in \mathbb{N}$ there is an $m \in \mathbb{N}$ satisfying $r^{-j} < h(r^{-m}) \leq r^{-j+1}$ then $h(r^{-g(n)}) \leq r^{-n+1}$, for all $n \in \mathbb{N}$.*

The assumption of Lemma 1 implies $h(r^{-n}) \geq r^{-n}$ and $h(r^{-(n+c)}) \geq h(r^{-n}) \cdot r^{-c}$. It is, in particular, satisfied if the function h is upwardly convex on $(0, 1)$ and $h(1) \geq 1$ (see [10, Lemma 3]).

For computable gauge functions $h : \mathbb{Q} \to \mathbb{R}$, relaxing Eq. (1) we obtain a corresponding computable modulus function.

Lemma 2 ([10, Lemma 4]). *Let $h : \mathbb{Q} \to \mathbb{R}$ be a computable gauge function satisfying the conditions that $1 < h(1) < r$ and for every $j \in \mathbb{N}$ there is an $m \in \mathbb{N}$ such that $r^{-j} < h(r^{-m}) \leq r^{-j+1}$. Then there is a computable strictly increasing function $g : \mathbb{N} \to \mathbb{N}$ such that $r^{-n-1} < h(r^{-g(n)}) < r^{-n+1}$, for all $n \in \mathbb{N}$.*

2 Universal Machines

In this section we introduce and study the notion of h-universal machine.

A *machine* T is a partial computable function from X^* to X^*. We use machine and function synonymously.

A prefix-free machine is a machine whose domain is a prefix-free language. The prefix complexity of a word w induced by a prefix-free machine T, $H_T(w)$, is $H_T(w) = \inf\{|\pi| : T(\pi) = w\}$. From now on all machines will be prefix-free and will be referred to simply as machines.

In analogy with [1] we say that a machine U is *h-universal* for a gauge function h if for all machines T there exists a constant $c_{U,T}$ such that for each program $\sigma \in X^*$ there exists a program $\pi \in X^*$ such that $U(\pi) = T(\sigma)$ and $-\log_r h(r^{-|\pi|}) \leq |\sigma| + c_{U,T}$. If $h(t) = t$ we get the classical notion of universal machine. Observe that, for gauge functions h, the function $\ell_h(n) := -\log_r h(r^{-n})$ is non-decreasing.

A machine U is *strictly h-universal* if U is h-universal but not h'-universal for any gauge function h' with $\lim_{n\to\infty} \frac{h'(r^{-n})}{h(r^{-n})} = 0$.

We fix a gauge function h and a universal machine T. We say that an ω-word ξ is *Chaitin h-random* if $H_T(\xi \upharpoonright n) \geq -\log_r h(r^{-n}) - O(n)$, and we say that ξ is *strictly Chaitin h-random* if ξ is Chaitin h-random and is not Chaitin h'-random for all gauge functions h' with $\lim_{n\to\infty} \frac{h'(r^{-n})}{h(r^{-n})} = 0$.

If T is universal and $h(t) = t^\varepsilon$, then we get Tadaki's definition of weak Chaitin ε-randomness (see [11,2]), if $h(t) = t$, then we get the classical definition of randomness.

Lemma 3. *The machine U is h-universal if and only if there exists a universal machine T and a constant $c_{U,T}$ such that $-\log_r h(r^{-H_U}) \leq H_T(w) + c_{U,T}$ for all $w \in X^*$.*

In [1] ε-universal machines were obtained from universal machines by padding the inputs. The next theorem shows that the same construction works also in the case of h-universal machines.

Theorem 1. *Let $h : \mathbb{Q} \to \mathbb{R}$ be a computable gauge function, and let $g : \mathbb{N} \to \mathbb{N}$ be a corresponding computable modulus function. If, for a universal machine T, we define $T_h(\pi \cdot 0^{g(|\pi|)-|\pi|}) := T(\pi)$ then $|-\log_r h(r^{-H_{T_h}(w)}) - H(w)| = O(1)$ and T_h is a strictly h-universal machine.*

Proof. Let π be a minimal description of w, that is, $|\pi| = H(w)$. Then $\pi \cdot 0^{g(|\pi|)-|\pi|}$ is a minimal description of w w.r.t. T_h. Consequently, Lemma 2 proves $|-\log_r h(r^{-H_{T_h}(w)}) - H(w)| = O(1)$. This also implies that T_h is h-universal.

Assume now T_h to be h'-universal for some h' tending faster to 0 than h. Then, on the one hand, $-\log_r h'(r^{-H_{T_h}(w)}) \leq H(w) + c$ for some constant c and, on the other hand, for every $i \in \mathbb{N}$ there is an n_i such that $-\log_r h'(r^{-H_{T_h}(w)}) \geq -\log_r h(r^{-H_{T_h}(w)}) + i$ for $H(w) \geq n_i$. This contradicts the relation $|-\log_r h(r^{-H_{T_h}(w)}) - H(w)| = O(1)$. □

Next we give examples for Chaitin h-random ω-words. We follow the line of Theorem 3 of [1] and define for a machine U the ω-word $\Omega_U \in X^\omega$ as the r-ary expansion of the halting probability of a machine U, that is, $0.\,\Omega_U := \sum_{w \in \mathrm{dom}(U)} r^{-|w|}$.

Theorem 2. *Let h be a computable gauge function satisfying the hypothesis of Lemma 2 and let U be an h-universal machine. Then Ω_U is Chaitin h-random.*

Proof. As in the proof of Theorem 3 of [1] one defines a partial computable function T for which $\mathbf{pref}(\Omega_U) \subseteq \mathrm{dom}(T)$ and $H_U(T(\Omega_U \restriction m)) \geq m$. From $H_U(v) \geq m$, we obtain $-\log_r h(r^{-m}) \leq -\log_r h(r^{-H_U(T(v))}) \leq H(T(v)) \leq H(v) + c$ whenever $v \in \mathrm{dom}(T)$, and the assertion follows. □

We conclude this section by considering the paradox of Theorem 6 of [1]. Here inequivalent ε-universal machines $V_{\varepsilon,k}, k = 0, 1, \ldots$ were defined. The machines $V_{\varepsilon,k}$ had the property that $\lim_{|w|\to\infty} H_{V_{\varepsilon,k}}(w) - H_{V_{\varepsilon,k+1}}(w) = \infty$.

Recall the definition of $V_{\varepsilon,k}$. In terms of modulus function $g_k : \mathbb{N} \to \mathbb{N}$ they can be described as $V_{\varepsilon,k}(\pi 0^{g_k(|\pi|)-|\pi|}) := T(\pi)$ where T is a universal machine and $g_k(n) := \max\{n, \lfloor \frac{1}{\varepsilon} \cdot n - k \cdot \log_r n \rfloor\}$. In contrast to [1] where all machines $V_{\varepsilon,k}$ were ε-universal our Theorem 2 states that only $V_{\varepsilon,k}$ is strictly h_k-universal for gauge functions satisfying $h_k(r^{-g_k(n)}) = r^{-n}$. Since $g_k(n) - g_{k+1}(n)$ tends to infinity as n grows, the function h_{k+1} tends faster to 0 than h_k and, consequently, $V_{\varepsilon,k}$ is not h_{k+1}-universal.

The paradox of Theorem 6 of [1] occurs because the family of gauge functions $\left(t^\varepsilon\right)_{\varepsilon\in(0,1)}$ admits intermediate computable functions, e.g. functions of the logarithmic scale like $h(t) = t^\varepsilon \cdot \left(\log_r \frac{1}{t}\right)^k$ (see [5]) but these functions were not taken into consideration in the definition of ε-universality.

3 Oscillation-Freeness

The aim of this section is to show that for a large class of gauge function there exist oscillation-free Chaitin h-random ω-words, that is, $\xi \in X^\omega$ such that $|H(\xi \upharpoonright n) + \log_r h(r^{-n})| = O(1)$.

We start with a generalisation of [1, Proposition 9].

Proposition 1. *Let $h : \mathbb{Q} \to \mathbb{N}$ be a gauge function such that for every $d \in \mathbb{N}$ there is an ℓ_d such that the inequality*

$$H(n) + d - 1 \le -\log_r \tfrac{h(r^{-(n+\ell)})}{h(r^{-\ell})} \le n - \big(H(n) + d - 1\big) \tag{2}$$

holds for all $\ell \ge \ell_d$ and, depending on the value of d, for all sufficiently large $n \in \mathbb{N}$.

Then there are $c, \ell' \in \mathbb{N}$ such that for all words $w \in X^*$, $|w| \ge \ell'$, exist words $v, u \in X^c$ such that

$$H(wu) + \log_r h(r^{-(|w|+c)}) \le H(w) + \log_r h(r^{-|w|}) - 1, \text{ and} \tag{3}$$
$$H(wv) + \log_r h(r^{-(|w|+c)}) \ge H(w) + \log_r h(r^{-|w|}) + 1\,. \tag{4}$$

Proof. As in the proof of Proposition 9 of [1], given $w \in X^*$ and $c \in \mathbb{N}$, one finds strings v, $|v| = c$ and $u = 0^c$ such that

$$H(wv) \ge H(w) + c - H(c) - d \text{ and } |H(w0^c) - H(w)| \le H(c) + d \tag{5}$$

where the constant d is independent of w and c.

Now, depending on d, choose a sufficiently large ℓ' and c, and the inequalities follow from Eqs. (5) and (2). □

Remark 1. The assumption of Proposition 1 is a little bit involved. Due to the fact that $H(n)$ is a slowly growing function one easily observes that Eq. (2) is satisfied whenever there are real numbers $\gamma, \overline{\gamma}$, $r^{-1} < \gamma \le \overline{\gamma} < 1$, such that $\gamma^n \le \frac{h(r^{-(n+\ell)})}{h(r^{-\ell})} \le \overline{\gamma}^n$ for all $\ell, n \in \mathbb{N}$. The latter is satisfied, in particular, for all length-invariant unbounded (p, q)-premeasures in the sense of [8].

The next theorem is an existence theorem for oscillation-free Chaitin h-random ω-words where h is a gauge functions fulfilling Eq. (2). This, in particular, guarantees that for arbitrary $\varepsilon \in (0, 1)$ oscillation-free Chaitin ε-random ω-words exist. The subsequent theorem will then consider the constructive case.

Theorem 3. *Let $h : \mathbb{Q} \to \mathbb{R}$ be a gauge function satisfying Eq. (2) and the assumption of Lemma 1. Then there is an ω-word $\xi \in X^\omega$ and a constant c_h such that $|H(\xi \upharpoonright n) - (-\log_r h(r^{-n}))| \le c_h$.*

Proof. We proceed as in the proof of Theorem 10 of [1]. We choose sufficiently large constants c and ℓ' from Proposition 1 such that $H(w) - c < H(w0^c)$,

$H(wv) \leq H(w) + 2c$ for $|v| = c$, $w \in X^*$ and, in view of $h(r^{-n}) \geq r^{-n}$, also $-\log_r h(r^{-\ell'}) < H(w)$ for some $w, |w| = \ell'$. Then we define $W \subseteq X^{\ell'} \cdot (X^c)^*$ as follows.

$$W := \left\{ w : w \in X^{\ell'} \cdot (X^c)^* \wedge \forall v \big(v \in W \wedge v \sqsubseteq w \to H(v) > -\log_r h(r^{-|v|}) \big) \right\}.$$

By the choice of ℓ' there is a $w \in X^{\ell'}$ with $-\log_r h(r^{-\ell'}) < H(w)$, and we have $W \neq \emptyset$. Eq. (4) shows that every $w \in W$ has an extension $wv \in W$ where $|v| = c$. Thus W contains infinite chains $w_0 \sqsubset w_1 \sqsubset \cdots \sqsubset w_i \sqsubset$ where $|w_{i+1}| - |w_i| = c$. Moreover, since h is non-decreasing and $H(wv) \leq H(w) + 2c$, $H(wv) + \log_r h(r^{-|wv|}) \leq H(w) + \log_r h(r^{-|w|}) + 2c$ for $|v| = c$.

Let $H(w) > -\log_r h(r^{-|w|}) + c + 1$. The function h satisfies the assumption of Lemma 1. Consequently, $-\log_r h(r^{-|w|+c}) \leq -\log_r h(r^{-|w|}) + c$, and Eq. (3) shows $-\log_r h(r^{-|w|+c}) + 1 < H(w0^c)$, that is, $w0^c \in W$. Finally Eq. (4) shows that then $H(w0^c) + \log_r h(r^{-|w|+c}) < H(w) + \log_r h(r^{-|w|})$.

Thus there is an infinite sequence $w_0 \sqsubset w_1 \sqsubset \cdots \sqsubset w_i \sqsubset$ of words in W such that $|w_{i+1}| - |w_i| = c$ and the differences $|H(w_i) + \log_r h(r^{-|w_i|})|$ remain bounded. □

Now consider the language W defined in the preceding proof. If h is a computable function, W is the complement of an computably enumerable language. Hence the infinite paths through W build a Π^0_1-definable ω-language $F \subseteq X^\omega$. Then the leftmost w.r.t. the lexicographical ordering ω-word ξ_{left} in F defines a left computable real $0.\xi_{\text{left}}$.

We show that ξ_{left} is oscillation-free h-random. Since $H(w) > -\log_r h(r^{-|w|})$ for $w \in W$, it suffices to verify that $H(\xi_{\text{left}} \upharpoonright n) + \log_r h(r^{-n}) \leq c_h$ for some constant c_h. We use the parameters c and ℓ' from the proof of Theorem 3.

Let $\mathbf{pref}(\xi_{\text{left}}) \cap W = \{w_i : i \in \mathbb{N} \wedge |w_i| = \ell' + i \cdot c\}$ where w_0 is the leftmost word in W. Choose a constant $c_h > \max\{H(w_0) + \log_r h(r^{-\ell'}), 4c\}$. Then $H(w_0) + \log_r h(r^{-|w_0|}) \leq c_h$. Assume that this relation holds for $j = 0, \ldots, i$. If $H(w_i) + \log_r h(r^{-|w_i|}) \leq 2c$ then $H(w_i v) + \log_r h(r^{-|w_i v|}) \leq 4c$ for all $v \in X^c$. Thus $H(w_{i+1}) + \log_r h(r^{-|w_{i+1}|}) \leq c_h$. If $2c < H(w_i) + \log_r h(r^{-|w_i|}) \leq c_h$ then $w_i 0^c$ is the leftmost successor of w_i in W and $0 < H(w_i) + \log_r h(r^{-|w_i|}) - 2c \leq H(w_i 0^c) + \log_r h(r^{-|w_i|+c}) < H(w_i) + \log_r h(r^{-|w_i|}) \leq c_h$.

This proves the following constructive version of Theorem 3.

Theorem 4. *Let $h : \mathbb{Q} \to \mathbb{R}$ be a computable gauge function which satisfies Eq. (2) and the hypothesis of Lemma 2. Then there exists an oscillation-free Chaitin h-random ω-word ξ such that $0.\xi$ is a left computable real.*

4 A Separation Theorem

In the preceding section we showed the existence of oscillation-free Chaitin h-random ω-words. For the gauge functions fulfilling the assumption of Lemma 2 we proved the existence of Π^0_1-definable ω-languages containing such ω-words as leftmost ones.

In a recent paper [10] we proved that, for a different kind of h-randomness (strong Martin-Löf randomness in the sense of [2]), there are Π^0_1-definable ω-languages containing oscillation-free h-random ω-words. We obtained these ω-languages by diluting ω-words. The concept of strong Martin-Löf randomness can be defined using the a priori complexity of words KA. For a definition of KA see [13,9,10] or [3][2]. We mention here only the following properties of KA.

Property 1. 1. An ω-word ξ is random if and only if $|\mathrm{KA}(\xi \restriction n) - n| = O(1)$,
2. $\mathrm{KA}(wv) \geq \mathrm{KA}(w) - O(1)$, for $w, v \in X^*$, and
3. $H(w) \geq \mathrm{KA}(w) - O(1)$ where the difference is unbounded.

For dilution we use prefix monotone mappings. Every prefix-monotone mapping $\varphi : X^* \to X^*$ defines as a limit a partial mapping $\overline{\varphi} :\subseteq X^\omega \to X^\omega$ in the following way: $\mathbf{pref}(\overline{\varphi}(\xi)) = \mathbf{pref}(\varphi(\mathbf{pref}(\xi)))$ whenever $\varphi(\mathbf{pref}(\xi))$ is an infinite set, and $\overline{\varphi}(\xi)$ is undefined when $\varphi(\mathbf{pref}(\xi))$ is finite.

If a (modulus) function $g : \mathbb{N} \to \mathbb{N}$ is strictly increasing we define a dilution function $\varphi : X^* \to X^*$ as follows.

$$\begin{aligned} \varphi(e) &:= 0^{g(0)} \text{ and} \\ \varphi(wx) &:= \varphi(w) \cdot x \cdot 0^{g(n+1)-g(n)-1} \text{ for } w \in X^* \text{ and } x \in X \end{aligned} \tag{6}$$

If φ is a dilution function then φ and also $\overline{\varphi}$ are one-to-one mappings. If, moreover, g is computable then φ is also computable and $\overline{\varphi}(X^\omega)$ is a Π^0_1-definable ω-language.

It holds the following estimate on the a priori complexity of a diluted string (see [9, Theorem 3.1]).

Lemma 4. *Let g be a computable strictly increasing modulus function and let φ be defined via Eq. (6). Then*

$$\big|\mathrm{KA}\big(\overline{\varphi}(\xi \restriction g(n))\big) - \mathrm{KA}\big(\xi \restriction n\big)\big| \leq O(1) \text{ for all } \xi \in X^\omega \ .$$

From Lemmata 4 and 2 and the above Property 1,2 we obtain immediately the following (cf. also [9, Theorem 3.3]).

Proposition 2. *Let h be a computable gauge function, g a corresponding computable modulus function and let φ be defined via Eq. (6). Then $|\mathrm{KA}(\xi \restriction (n+1)) - \mathrm{KA}(\xi \restriction n)| = O(1)$ implies $|\mathrm{KA}(\overline{\varphi}(\xi) \restriction g(n)) + \log_r h(r^{-n})| = O(1)$.*

Property 1,1 shows that Proposition 2 holds for random ω-words ξ. In that case $\overline{\varphi}(\xi)$ is strongly Martin-Löf h-random. Next we consider the situation for prefix complexity. Here we have $|H(w) - H(\varphi(w))| = O(1)$ whenever φ is a partial computable one-to-one function. Thus we obtain a theorem analogous to Lemma 4 for prefix complexity H.

Lemma 5. *Let g be a computable strictly increasing modulus function and let φ be defined via Eq. (6). Then*

$$\big|H\big(\overline{\varphi}(\xi \restriction g(n))\big) - H\big(\xi \restriction n\big)\big| \leq O(1) \text{ for all } \xi \in X^\omega \ .$$

This much preparation allows us to prove our separation theorem.

[2] In [3] a priori complexity is denoted by KM.

Theorem 5. *Let $h : \mathbb{Q} \to \mathbb{R}$ be a computable gauge function which satisfies Eq. (2) and the hypothesis of Lemma 2. Then there exists a Π^0_1-definable ω-language which contains an oscillation-free strongly Martin-Löf h-random ω-word ξ but no oscillation-free Chaitin h-random ω-word.*

Proof. From Lemma 2 we obtain a computable strictly increasing modulus function g such that $|-\log_r h(r^{-g(n)}) - n| \leq 1$. Define φ according to Eq.(6) and choose an arbitrary random ω-word $\zeta \in X^\omega$. Then Proposition 2 shows that $\overline{\varphi}(\zeta)$ is oscillation-free strongly Martin-Löf h-random.

Next we show that the Π^0_1-definable ω-language $\overline{\varphi}(X^\omega)$ does not contain any oscillation-free Chaitin h-random ω-word.

Assume that, for some $\xi \in X^\omega$, the ω-word $\overline{\varphi}(\xi)$ is oscillation-free Chaitin h-random. Then $|H(\overline{\varphi}(\xi) \upharpoonright g(n)) + \log_r h(r^{-g(n)})| = O(1)$, and, consequently, $|H(\xi \upharpoonright n) - n| = O(1)$. But this is impossible as $H(\xi \upharpoonright n) \geq n - c$, for all $n \in \mathbb{N}$, implies $\lim_{n\to\infty} H(\xi \upharpoonright n) - n = \infty$. □

References

1. Calude, C.S., Hay, N.J., Stephan, F.: Representation of left-computable ε-random reals. J. Comput. System Sci. 77(4), 812–819 (2011), http://dx.doi.org/10.1016/j.jcss.2010.08.001
2. Calude, C.S., Staiger, L., Terwijn, S.A.: On partial randomness. Ann. Pure Appl. Logic 138(1-3), 20–30 (2006), http://dx.doi.org/10.1016/j.apal.2005.06.004
3. Downey, R.G., Hirschfeldt, D.R.: Algorithmic Randomness and Complexity. Theory and Applications of Computability. Springer, New York (2010)
4. Graf, S., Mauldin, R.D., Williams, S.C.: The exact Hausdorff dimension in random recursive constructions. Mem. Amer. Math. Soc. 71(381), x+121 (1988)
5. Hausdorff, F.: Dimension und äußeres Maß. Math. Ann. 79(1-2), 157–179 (1918), http://dx.doi.org/10.1007/BF01457179
6. Mielke, J.: Verfeinerung der Hausdorff-Dimension und Komplexität von ω-Sprachen. Ph.D. thesis, Martin-Luther-Universität Halle-Wittenberg (2010), http://nbn-resolving.de/urn:nbn:de:gbv:3:4-2816 (in German)
7. Mielke, J., Staiger, L.: On oscillation-free ε-random sequences II. In: Bauer, A., Hertling, P., Ko, K.I. (eds.) Computability and Complexity in Analysis. Dagstuhl Seminar Proceedings, vol. 09003, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany (2009), http://drops.dagstuhl.de/opus/volltexte/2009/2269
8. Reimann, J., Stephan, F.: Hierarchies of randomness tests. In: Goncharov, S.S., Downey, R., Ono, H. (eds.) Mathematical logic in Asia, pp. 215–232. World Scientific, Hackensack (2006)
9. Staiger, L.: On oscillation-free ε-random sequences. Electr. Notes Theor. Comput. Sci. 221, 287–297 (2008)
10. Staiger, L.: Constructive Dimension and Hausdorff Dimension: The Case of Exact Dimension. In: Owe, O., Steffen, M., Telle, J.A. (eds.) FCT 2011. LNCS, vol. 6914, pp. 252–263. Springer, Heidelberg (2011)
11. Tadaki, K.: A generalization of Chaitin's halting probability Ω and halting self-similar sets. Hokkaido Math. J. 31(1), 219–253 (2002)

12. Tadaki, K.: A New Representation of Chaitin Ω Number Based on Compressible Strings. In: Calude, C.S., Hagiya, M., Morita, K., Rozenberg, G., Timmis, J. (eds.) UC 2010. LNCS, vol. 6079, pp. 127–139. Springer, Heidelberg (2010)
13. Uspensky, V.A., Shen, A.: Relations between varieties of Kolmogorov complexities. Math. Systems Theory 29(3), 271–292 (1996), http://dx.doi.org/10.1007/BF01201280

Phase Transition between Unidirectionality and Bidirectionality

Kohtaro Tadaki

Research and Development Initiative, Chuo University, JST CREST
1–13–27 Kasuga, Bunkyo-ku, Tokyo 112-8551, Japan
tadaki@kc.chuo-u.ac.jp
http://www2.odn.ne.jp/tadaki/

Abstract. The notion of weak truth-table reducibility plays an important role in recursion theory. In this paper, we introduce an elaboration of this notion, where a computable bound on the use function is explicitly specified. This elaboration enables us to deal with the notion of asymptotic behavior in a manner like in computational complexity theory, while staying in computability theory. We apply the elaboration to sets which appear in the statistical mechanical interpretation of algorithmic information theory. We demonstrate the power of the elaboration by revealing a critical phenomenon, i.e., a phase transition, in the statistical mechanical interpretation, which cannot be captured by the original notion of weak truth-table reducibility.

1 Introduction

The notion of weak truth-table reducibility plays an important role in recursion theory (see e.g. [12,9]). For any sets $A, B \subset \mathbb{N}$, we say that A *is weak truth-table reducible to* B, denoted $A \leq_{wtt} B$, if there exist an oracle Turing machine M and a total recursive function $g\colon \mathbb{N} \to \mathbb{N}$ such that A is Turing reducible to B via M and, on every input $n \in \mathbb{N}$, M only queries natural numbers at most $g(n)$. In this paper, we introduce an elaboration of this notion, where the total recursive bound g on the use of the reduction is explicitly specified. In doing so, in particular we try to follow the fashion in which computational complexity theory is developed, while staying in computability theory. We apply the elaboration to sets which appear in the theory of program-size, i.e., algorithmic information theory (AIT, for short) [7,8,1,12,9]. The elaboration, called *reducibility in query size* f, is introduced as follows.

Definition 1 (Reducibility in Query Size f). *Let $f\colon \mathbb{N} \to \mathbb{N}$, and let $A, B \subset \{0,1\}^*$. We say that A is reducible to B in query size f if there exists an oracle Turing machine M such that*

(i) A is Turing reducible to B via M, and
(ii) on every input $x \in \{0,1\}^*$, M only queries strings of length at most $f(|x|)$.

□

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 203–223, 2012.

For any fixed sets A and B, the above definition allows us to consider the notion of asymptotic behavior for the function f which bounds the use of the reduction, i.e., which imposes the restriction on the use of the computational resource (i.e., the oracle B). Thus, by the above definition, even in the context of computability theory, we can deal with the notion of asymptotic behavior in a manner like in computational complexity theory. Recall here that the notion of input size plays a crucial role in computational complexity theory since computational complexity such as time complexity and space complexity is measured based on it. This is also true in AIT since the program-size complexity is measured based on input size. Thus, in Definition 1 we consider a reduction between subsets of $\{0,1\}^*$ and not a reduction between subsets of $\mathbb{N}$ as in the original weak truth-table reducibility. Moreover, in Definition 1 we require the bound $f(|x|)$ to depend only on input size $|x|$ as in computational complexity theory, and not on input x itself as in the original weak truth-table reducibility. We pursue a formal correspondence to computational complexity theory in this manner, while staying in computability theory.

In this paper we demonstrate the power of the notion of reducibility in query size f in the context of AIT. In [7] Chaitin introduced Ω number as a concrete example of random real. His Ω is defined as the probability that an optimal prefix-free machine U halts, and plays a central role in the development of AIT. Here the notion of *optimal prefix-free machine* is used to define the notion of *program-size complexity* $H(s)$ for a finite binary string s. The first n bits of the base-two expansion of Ω solve the halting problem of the optimal prefix-free machine U for all binary inputs of length at most n. Using this property, Chaitin showed Ω to be a random real. Let $\mathrm{dom}\, U$ be the set of all halting inputs for U. Calude and Nies [4], in essence, showed the following theorem on the relation between the base-two expansion of Ω and the halting problem $\mathrm{dom}\, U$.

Theorem 1 (Calude and Nies [4]). *Ω and $\mathrm{dom}\, U$ are weak truth-table equivalent. Namely, $\Omega \leq_{wtt} \mathrm{dom}\, U$ and $\mathrm{dom}\, U \leq_{wtt} \Omega$.* □

In [15] we generalized Ω to $Z(T)$ in such a way that the partial randomness of $Z(T)$ equals to T if T is a computable real with $0 < T \leq 1$.[1] Here the notion of *partial randomness* of a real is a stronger representation of the compression rate of the real by means of program-size complexity. The real function $Z(T)$ of T is a function of class C^∞ on $(0,1)$ and an increasing continuous function on $(0,1]$. In the case of $T = 1$, $Z(T)$ results in Ω, i.e., $Z(1) = \Omega$. We can show Theorem 2 below for $Z(T)$. This theorem follows immediately from stronger results, Theorems 21 and 22, which are two of the main results of this paper.

Theorem 2. *Suppose that T is a computable real with $0 < T < 1$. Then $Z(T)$ and $\mathrm{dom}\, U$ are weak truth-table equivalent.* □

When comparing Theorem 1 and Theorem 2, we see that there is no difference between $T = 1$ and $T < 1$ with respect to the weak truth-table equivalence

[1] In [15], $Z(T)$ is denoted by Ω^T.

between $Z(T)$ and $\mathrm{dom}\, U$. In this paper, however, we show that there is a critical difference between $T = 1$ and $T < 1$ in the relation between $Z(T)$ and $\mathrm{dom}\, U$ from the point of view of the reducibility in query size f. Based on the notion of reducibility in query size f, we introduce the notions of *unidirectionality* and *bidirectionality* between two sets A and B in this paper. These notions enable us to investigate the relative computational power between A and B.

Theorems 6 and 7 below are two of the main results of this paper. Theorem 6 gives a succinct equivalent characterization of f for which Ω is reducible to $\mathrm{dom}\, U$ in query size f and reversely Theorem 7 gives a succinct equivalent characterization of f for which $\mathrm{dom}\, U$ is reducible to Ω in query size f, both in a general setting. Based on them, we show in Theorem 8 below that the computation from Ω to $\mathrm{dom}\, U$ is unidirectional and the computation from $\mathrm{dom}\, U$ to Ω is also unidirectional. On the other hand, Theorems 21 and 22 below are also two of the main results of this paper. Theorem 21 gives a succinct equivalent characterization of f for which $Z(T)$ is reducible to $\mathrm{dom}\, U$ in query size f and reversely Theorem 22 gives a succinct equivalent characterization of f for which $\mathrm{dom}\, U$ is reducible to $Z(T)$ in query size f, both in a general setting, in the case where T is a computable real with $0 < T < 1$. Based on them, we show in Theorem 23 below that the computations between $Z(T)$ and $\mathrm{dom}\, U$ are bidirectional if T is a computable real with $0 < T < 1$. In this way the notion of reducibility in query size f can reveal a critical difference of the behavior of $Z(T)$ between $T = 1$ and $T < 1$, which cannot be captured by the original notion of weak truth-table reducibility.

Recall that the weak truth-table reducibility is defined for two subsets of $\mathbb{N}$. Thus, when we apply the notion of weak truth-table reducibility to a real α, we regard α as a subset of $\mathbb{N}$ whose characteristic sequence equals to the base-two expansion of α. In fact, in Theorem 1, the real Ω is regarded as a subset of $\mathbb{N}$ in this manner. On the other hand, the notion of reducibility in query size f is defined for two subsets of $\{0,1\}^*$. Thus, when we apply this notion to a real, we have to somehow regard it as a subset of $\{0,1\}^*$. We do this by using the following notion of prefixes of a real.

Definition 2 (Prefixes of Real). *For each $\alpha \in \mathbb{R}$, the prefixes* $\mathrm{Pf}(\alpha)$ *of α is the subset of $\{0,1\}^*$ defined by $\mathrm{Pf}(\alpha) = \{\alpha\restriction_n \mid n \in \mathbb{N}\}$, where $\alpha\restriction_n$ is the first n bits of the base-two expansion of the fractional part $\alpha - \lfloor\alpha\rfloor$ of α.* □

The notion of prefixes of a real is a natural notion in AIT. For example, the randomness of a real α can be rephrased as that there exists $d \in \mathbb{N}$ such that, for every $x \in \mathrm{Pf}(\alpha)$, $|x| \leq H(x) + d$. The notion of prefixes of a real helps us see the aforementioned unidirectionality and bidirectionality.

In [16] we introduced and developed a *statistical mechanical interpretation of AIT*. We there introduced the notion of *thermodynamic quantities at temperature T* such as partition function, free energy, energy, statistical mechanical entropy, and specific heat into AIT. Among these thermodynamic quantities, $Z(T)$ is the partition function at temperature T. The work [16] showed that the values of all the thermodynamic quantities diverge when the temperature T exceeds 1. This phenomenon might be regarded as some sort of phase transition in statistical

mechanics. Thus, this paper reveals a new aspect of the phase transition by showing the critical difference of the behavior of $Z(T)$ between $T = 1$ and $T < 1$ in terms of the notion of reducibility in query size f.

In our former work [17] we considered some elaboration of weak truth-table equivalence between Ω and $\mathrm{dom}\, U$ and showed the unidirectionality between them in a certain form. Compared with this paper, however, the treatments of [17] were insufficient in the correspondence to computational complexity theory. In this paper, based on the notion of reducibility in query size f, we sharpen the results of [17] with a thorough emphasis on a formal correspondence to computational complexity theory.

The paper is organized as follows. We begin in Section 2 with some preliminaries to AIT and partial randomness. In Section 3 we investigate simple properties of the notion of reducibility in query size f and introduce the notions of unidirectionality and bidirectionality between two sets based on it. We then show in Section 4 the unidirectionality between Ω and $\mathrm{dom}\, U$ in a general setting. In Section 5 we present theorems which play a crucial role in establishing the bidirectionality in Section 6. Based on them, we show in Section 6 the bidirectionality between $Z(T)$ and $\mathrm{dom}\, U$ with a computable real $T \in (0,1)$ in a general setting. We conclude this paper with the remarks on the origin of the phase transition of the behavior of $Z(T)$ between $T = 1$ and $T < 1$ in Section 7.

2 Preliminaries

We start with some notation about numbers and strings which will be used in this paper. $\mathbb{N} = \{0, 1, 2, 3, \dots\}$ is the set of natural numbers, and $\mathbb{N}^+$ is the set of positive integers. $\mathbb{Z}$ is the set of integers, and $\mathbb{Q}$ is the set of rationals. $\mathbb{R}$ is the set of reals. A sequence $\{a_n\}_{n\in\mathbb{N}}$ of numbers (rationals or reals) is called *increasing* if $a_{n+1} > a_n$ for all $n \in \mathbb{N}$.

Normally, $o(n)$ denotes any function $f\colon \mathbb{N}^+ \to \mathbb{R}$ such that $\lim_{n\to\infty} f(n)/n = 0$. On the other hand, $O(1)$ denotes any function $g\colon \mathbb{N}^+ \to \mathbb{R}$ such that there is $C \in \mathbb{R}$ with the property that $|g(n)| \leq C$ for all $n \in \mathbb{N}^+$.

$\{0,1\}^* = \{\lambda, 0, 1, 00, 01, 10, 11, 000, 001, 010, \dots\}$ is the set of finite binary strings where λ denotes the *empty string*, and $\{0,1\}^*$ is ordered as indicated. We identify any string in $\{0,1\}^*$ with a natural number in this order, i.e., we consider $\varphi\colon \{0,1\}^* \to \mathbb{N}$ such that $\varphi(s) = 1s - 1$ where the concatenation $1s$ of strings 1 and s is regarded as a dyadic integer, and then we identify s with $\varphi(s)$. For any $s \in \{0,1\}^*$, $|s|$ is the *length* of s. For any $n \in \mathbb{N}$, we denote by $\{0,1\}^n$ the set $\{\, s \mid s \in \{0,1\}^* \ \&\ |s| = n\}$. A subset S of $\{0,1\}^*$ is called *prefix-free* if no string in S is a prefix of another string in S. For any subset S of $\{0,1\}^*$ and any $n \in \mathbb{N}$, we denote by $S{\restriction}_n$ the set $\{s \in S \mid |s| \leq n\}$. For any function f, the domain of definition of f is denoted by $\mathrm{dom}\, f$. We write "r.e." instead of "recursively enumerable."

Let α be an arbitrary real. $\lfloor \alpha \rfloor$ is the greatest integer less than or equal to α, and $\lceil \alpha \rceil$ is the smallest integer greater than or equal to α. For any $n \in \mathbb{N}$,

we denote by $\alpha\restriction_n \in \{0,1\}^*$ the first n bits of the base-two expansion of $\alpha - \lfloor\alpha\rfloor$ with infinitely many zeros. For example, in the case of $\alpha = 5/8$, $\alpha\restriction_6 = 101000$. On the other hand, for any non-positive integer $n \in \mathbb{Z}$, we set $\alpha\restriction_n = \lambda$.

A real α is called *r.e.* if there exists a computable, increasing sequence of rationals which converges to α. An r.e. real is also called a *left-computable* real. We say that a real α is *computable* if there exists a computable sequence $\{a_n\}_{n\in\mathbb{N}}$ of rationals such that $|\alpha - a_n| < 2^{-n}$ for all $n \in \mathbb{N}$. It is then easy to see that, for every real α, the following four conditions are equivalent: (i) α is computable. (ii) α is r.e. and $-\alpha$ is r.e. (iii) If $f\colon \mathbb{N} \to \mathbb{Z}$ with $f(n) = \lceil \alpha n \rceil$ then f is a total recursive function. (iv) If $g\colon \mathbb{N} \to \mathbb{Z}$ with $g(n) = \lfloor \alpha n \rfloor$ then g is a total recursive function.

2.1 Algorithmic Information Theory

In the following we concisely review some definitions and results of AIT [7,8,1,12,9]. A *prefix-free machine* is a partial recursive function $F\colon \{0,1\}^* \to \{0,1\}^*$ such that $\mathrm{dom}\, F$ is a prefix-free set. For each prefix-free machine F and each $s \in \{0,1\}^*$, $H_F(s)$ is defined by $H_F(s) = \min\left\{\, |p| \;\middle|\; p \in \{0,1\}^* \,\&\, F(p) = s \,\right\}$ (may be ∞). A prefix-free machine U is said to be *optimal* if for each prefix-free machine F there exists $d \in \mathbb{N}$ with the following property; if $p \in \mathrm{dom}\, F$, then there is $q \in \mathrm{dom}\, U$ for which $U(q) = F(p)$ and $|q| \leq |p| + d$. It is then easy to see that there exists an optimal prefix-free machine. We choose a particular optimal prefix-free machine U as the standard one for use, and define $H(s)$ as $H_U(s)$, which is referred to as the *program-size complexity* of s or the *Kolmogorov complexity* of s. For any $s, t \in \{0,1\}^*$, we define $H(s,t)$ as $H(b(s,t))$, where $b\colon \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ is a particular bijective total recursive function.

Chaitin [7] introduced Ω number as follows. For each optimal prefix-free machine V, the halting probability Ω_V of V is defined by

$$\Omega_V = \sum_{p \in \mathrm{dom}\, V} 2^{-|p|}.$$

For every optimal prefix-free machine V, since $\mathrm{dom}\, V$ is prefix-free, Ω_V converges and $0 < \Omega_V \leq 1$. For any $\alpha \in \mathbb{R}$, we say that α is *weakly Chaitin random* if there exists $c \in \mathbb{N}$ such that $n - c \leq H(\alpha\restriction_n)$ for all $n \in \mathbb{N}^+$ [7,8]. Chaitin [7] showed that Ω_V is weakly Chaitin random for every optimal prefix-free machine V. Therefore $0 < \Omega_V < 1$ for every optimal prefix-free machine V.

Let M be a deterministic Turing machine with the input and output alphabet $\{0,1\}$, and let F be a prefix-free machine. We say that M *computes* F if the following holds: for every $p \in \{0,1\}^*$, when M starts with the input p, (i) M halts and outputs $F(p)$ if $p \in \mathrm{dom}\, F$; (ii) M does not halt forever otherwise. We use this convention on the computation of a prefix-free machine by a deterministic Turing machine throughout the rest of this paper. Thus, we exclude the possibility that there is $p \in \{0,1\}^*$ such that, when M starts with the input p, M halts but $p \notin \mathrm{dom}\, F$. For any $p \in \{0,1\}^*$, we denote the running time of M on the input p by $T_M(p)$ (may be ∞). Thus, $T_M(p) \in \mathbb{N}$ for every $p \in \mathrm{dom}\, F$ if M computes F.

We define $L_M = \min\{\, |p| \mid p \in \{0,1\}^* \ \&\ M \text{ halts on input } p\}$ (may be ∞). For any $n \geq L_M$, we define I_M^n as the set of all halting inputs p for M with $|p| \leq n$ which take longest to halt in the computation of M, i.e., as the set $\{\, p \in \{0,1\}^* \mid |p| \leq n \ \&\ T_M(p) = T_M^n \,\}$ where T_M^n is the maximum running time of M on all halting inputs of length at most n. In the work [17], we slightly strengthened the result presented in Chaitin [8] to obtain Theorem 3 below (see Note in Section 8.1 of Chaitin [8]).

Theorem 3 (Chaitin [8] and Tadaki [17]). *Let V be an optimal prefix-free machine, and let M be a deterministic Turing machine which computes V. Then $n = H(n,p) + O(1) = H(p) + O(1)$ for all (n,p) with $n \geq L_M$ and $p \in I_M^n$.* □

Note that the proof of Theorem 3 is omitted in the work [17]. See Appendix A of Tadaki [19] for the proof.

2.2 Partial Randomness

In the work [15], we generalized the notion of the randomness of a real so that *the degree of the randomness*, which is often referred to as *the partial randomness* recently [5,13,6,9], can be characterized by a real T with $0 \leq T \leq 1$ as follows.

Definition 3. *Let $T \in [0,1]$ and let $\alpha \in \mathbb{R}$. We say that α is weakly Chaitin T-random if there exists $c \in \mathbb{N}$ such that, for all $n \in \mathbb{N}^+$, $Tn - c \leq H(\alpha\restriction_n)$.* □

In the case of $T = 1$, the weak Chaitin T-randomness results in the weak Chaitin randomness.

Definition 4. *Let $T \in [0,1]$ and let $\alpha \in \mathbb{R}$. We say that α is T-compressible if $H(\alpha\restriction_n) \leq Tn + o(n)$, i.e., if $\limsup_{n\to\infty} H(\alpha\restriction_n)/n \leq T$. We say that α is strictly T-compressible if there exists $d \in \mathbb{N}$ such that, for all $n \in \mathbb{N}^+$, $H(\alpha\restriction_n) \leq Tn + d$.* □

For every $T \in [0,1]$ and every $\alpha \in \mathbb{R}$, if α is weakly Chaitin T-random and T-compressible, then $\lim_{n\to\infty} H(\alpha\restriction_n)/n = T$, i.e., the *compression rate* of α equals to T.

In the work [15], we generalized Chaitin Ω number to $Z(T)$ as follows. For each optimal prefix-free machine V and each real $T > 0$, the *partition function* $Z_V(T)$ of V at temperature T is defined by

$$Z_V(T) = \sum_{p \in \mathrm{dom}\, V} 2^{-\frac{|p|}{T}}.$$

Thus, $Z_V(1) = \Omega_V$. If $0 < T \leq 1$, then $Z_V(T)$ converges and $0 < Z_V(T) < 1$, since $Z_V(T) \leq \Omega_V < 1$. The following theorem holds for $Z_V(T)$.

Theorem 4 (Tadaki [15]). *Let V be an optimal prefix-free machine.*

(i) If $0 < T \leq 1$ and T is computable, then $Z_V(T)$ is an r.e. real which is weakly Chaitin T-random and T-compressible.
(ii) If $1 < T$, then $Z_V(T)$ diverges to ∞. □

An r.e. real has a special property on partial randomness, as shown in Theorem 5 below. For any r.e. reals α and β, we say that α *dominates* β if there are computable, increasing sequences $\{a_n\}$ and $\{b_n\}$ of rationals and $c \in \mathbb{N}^+$ such that $\lim_{n\to\infty} a_n = \alpha$, $\lim_{n\to\infty} b_n = \beta$, and $c(\alpha - a_n) \geq \beta - b_n$ for all $n \in \mathbb{N}$ [14].

Definition 5 (Tadaki [18]). *Let $T \in (0,1]$. An increasing sequence $\{a_n\}$ of reals is called T-convergent if $\sum_{n=0}^{\infty}(a_{n+1} - a_n)^T < \infty$. An r.e. real α is called T-convergent if there exists a T-convergent computable, increasing sequence of rationals which converges to α. An r.e. real α is called $\Omega(T)$-like if it dominates all T-convergent r.e. reals.* □

Theorem 5 (Equivalent Characterizations of Partial Randomness for an R.E. Real, Tadaki [18]). *Let T be a computable real in $(0,1]$, and let α be an r.e. real. Then the following three conditions are equivalent: (i) α is weakly Chaitin T-random. (ii) α is $\Omega(T)$-like. (iii) For every T-convergent r.e. real β there exists $d \in \mathbb{N}$ such that, for all $n \in \mathbb{N}^+$, $H(\beta\restriction_n) \leq H(\alpha\restriction_n) + d$.* □

3 Reducibility in Query Size f

In this section we investigate some properties of the notion of reducibility in query size f and introduce the notions of unidirectionality and bidirectionality between two sets.

First note that, for every set $A \subset \{0,1\}^*$, A is reducible to A in query size n, where "n" denotes the identity function $I\colon \mathbb{N} \to \mathbb{N}$ with $I(n) = n$. We follow the notation in computational complexity theory. The following are simple observations on the notion of reducibility in query size f.

Proposition 1. *Let $f\colon \mathbb{N} \to \mathbb{N}$ and $g\colon \mathbb{N} \to \mathbb{N}$, and let $A, B, C \subset \{0,1\}^*$.*

(i) If A is reducible to B in query size f and B is reducible to C in query size g, then A is reducible to C in query size $g \circ f$.

(ii) Suppose that $f(n) \leq g(n)$ for every $n \in \mathbb{N}$. If A is reducible to B in query size f then A is reducible to B in query size g.

(iii) Suppose that A is reducible to B in query size f. If A is not recursive then f is unbounded. □

The following proposition is a restatement of the well-known fact that, for every optimal prefix-free machine V, the first n bits of the base-two expansion of Ω_V solve the halting problem of V for inputs of length at most n.

Proposition 2. *Let V be an optimal prefix-free machine. Then $\operatorname{dom} V$ is reducible to $\mathrm{Pf}(\Omega_V)$ in query size n.* □

Definition 6. *An order function is a non-decreasing total recursive function $f\colon \mathbb{N} \to \mathbb{N}$ such that $\lim_{n\to\infty} f(n) = \infty$.* □

Let f be an order function. Intuitively, the notion of the reduction of A to B in query size f is equivalent to that, for every $n \in \mathbb{N}$, if n and $B\restriction_{f(n)}$ are given,

then $A\restriction_n$ can be calculated. With this view in mind, we introduce the notions of unidirectionality and bidirectionality between two sets as follows.

Definition 7. *Let $A, B \subset \{0,1\}^*$. We say that the computation from A to B is unidirectional if the following holds: For every order functions f and g, if B is reducible to A in query size f and A is reducible to B in query size g then the function $g(f(n)) - n$ of $n \in \mathbb{N}$ is unbounded. We say that the computations between A and B are bidirectional if the computation from A to B is not unidirectional and the computation from B to A is not unidirectional.* □

The notion of unidirectionality of the computation from A to B in the above definition is, in essence, interpreted as follows: No matter how a order function f is chosen, if f satisfies that $B\restriction_n$ can be calculated from n and $A\restriction_{f(n)}$, then $A\restriction_{f(n)}$ cannot be calculated from n and $B\restriction_{n+O(1)}$.

4 Unidirectionality

In this section we show the unidirectionality between Ω_U and $\mathrm{dom}\, U$ in a general setting. Theorems 6 and 7 below are two of the main results of this paper.

Theorem 6 (Elaboration of $\Omega_U \leq_{wtt} \mathrm{dom}\, U$). *Let V and W be optimal prefix-free machines, and let f be an order function. Then the following two conditions are equivalent:*

(i) $\mathrm{Pf}(\Omega_V)$ *is reducible to* $\mathrm{dom}\, W$ *in query size* $f(n) + O(1)$.
(ii) $\sum_{n=0}^{\infty} 2^{n-f(n)} < \infty$. □

Theorem 6 is proved in Subsection 4.1 below. Theorem 6 corresponds to Theorem 4 of Tadaki [17], and is proved by modifying the proof of Theorem 4 of [17]. Let V and W be optimal prefix-free machines. The implication (ii) $\Rightarrow$ (i) of Theorem 6 results in, for example, that $\mathrm{Pf}(\Omega_V)$ is reducible to $\mathrm{dom}\, W$ in query size $n + \lfloor (1+\varepsilon)\log_2 n \rfloor + O(1)$ for every real $\varepsilon > 0$. On the other hand, the implication (i) $\Rightarrow$ (ii) of Theorem 6 results in, for example, that $\mathrm{Pf}(\Omega_V)$ is not reducible to $\mathrm{dom}\, W$ in query size $n + \lfloor \log_2 n \rfloor + O(1)$ and therefore, in particular, $\mathrm{Pf}(\Omega_V)$ is not reducible to $\mathrm{dom}\, W$ in query size $n + O(1)$.

Theorem 7 (Elaboration of $\mathrm{dom}\, U \leq_{wtt} \Omega_U$). *Let V and W be optimal prefix-free machines, and let f be an order function. Then the following two conditions are equivalent:*

(i) $\mathrm{dom}\, W$ *is reducible to* $\mathrm{Pf}(\Omega_V)$ *in query size* $f(n) + O(1)$.
(ii) $n \leq f(n) + O(1)$. □

Theorem 7 is proved in Subsection 4.2 below. Theorem 7 corresponds to Theorem 11 of Tadaki [17], and is proved by modifying the proof of Theorem 11 of [17]. The implication (ii) $\Rightarrow$ (i) of Theorem 7 results in that, for every optimal prefix-free machines V and W, $\mathrm{dom}\, W$ is reducible to $\mathrm{Pf}(\Omega_V)$ in query size $n + O(1)$. On the other hand, the implication (i) $\Rightarrow$ (ii) of Theorem 7 says that this upper bound "$n + O(1)$" of the query size is, in essence, tight.

Theorem 8. *Let V and W be optimal prefix-free machines. Then the computation from* $\mathrm{Pf}(\Omega_V)$ *to* $\mathrm{dom}\, W$ *is unidirectional and the computation from* $\mathrm{dom}\, W$ *to* $\mathrm{Pf}(\Omega_V)$ *is also unidirectional.*

Proof. Let V and W be optimal prefix-free machines. For arbitrary order functions f and g, assume that $\mathrm{dom}\, W$ is reducible to $\mathrm{Pf}(\Omega_V)$ in query size f and $\mathrm{Pf}(\Omega_V)$ is reducible to $\mathrm{dom}\, W$ in query size g. It follows from the implication (i) $\Rightarrow$ (ii) of Theorem 7 that there exists $c \in \mathbb{N}$ for which $n \leq f(n) + c$ for all $n \in \mathbb{N}$. On the other hand, it follows from the implication (i) $\Rightarrow$ (ii) of Theorem 6 that $\sum_{n=0}^{\infty} 2^{n-g(n)} < \infty$ and therefore $\lim_{n\to\infty} g(n) - n = \infty$. Since g is an order function, we have $g(f(n)) - n \geq g(n-c) - (n-c) - c$ for all $n \geq c$. Thus, the computation from $\mathrm{Pf}(\Omega_V)$ to $\mathrm{dom}\, W$ is unidirectional. On the other hand, we have $f(g(n)) - n \geq g(n) - n - c$ for all $n \in \mathbb{N}$. Thus, the computation from $\mathrm{dom}\, W$ to $\mathrm{Pf}(\Omega_V)$ is unidirectional. □

4.1 The Proof of Theorem 6

Theorem 6 follows from Theorems 12 and 14 below, and the fact that Ω_V is a weakly Chaitin random r.e. real for every optimal prefix-free machine V. We first prove Theorem 12 using Theorems 9 and 11 below.

Theorem 9 (Kraft-Chaitin Theorem, Chaitin [7]). *Let $f\colon \mathbb{N} \to \mathbb{N}$ be a total recursive function such that $\sum_{n=0}^{\infty} 2^{-f(n)} \leq 1$. Then there exists a total recursive function $g\colon \mathbb{N} \to \{0,1\}^*$ such that (i) g is an injection, (ii) the set $\{\, g(n) \mid n \in \mathbb{N}\}$ is prefix-free, and (iii) $|g(n)| = f(n)$ for all $n \in \mathbb{N}$.* □

We refer to Theorem 10 below from Tadaki [17]. Theorem 11 is a restatement of it using the notion of reducibility in query size f.

Theorem 10 (Tadaki [17]). *Let V be an optimal prefix-free machine. Then, for every prefix-free machine F there exists $d \in \mathbb{N}$ such that, for every $p \in \{0,1\}^*$, if p and the list of all halting inputs for V of length at most $|p| + d$ are given, then the halting problem of the input p for F can be solved.* □

Theorem 11. *Let V be an optimal prefix-free machine. Then, for every prefix-free machine F there exists $d \in \mathbb{N}$ such that* $\mathrm{dom}\, F$ *is reducible to* $\mathrm{dom}\, V$ *in query size $n + d$.* □

Theorem 12. *Let α be an r.e. real, and let V be an optimal prefix-free machine. For every total recursive function $f\colon \mathbb{N} \to \mathbb{N}$, if $\sum_{n=0}^{\infty} 2^{n-f(n)} < \infty$, then there exists $c \in \mathbb{N}$ such that* $\mathrm{Pf}(\alpha)$ *is reducible to* $\mathrm{dom}\, V$ *in query size $f(n) + c$.*

Proof. Let α be an r.e. real, and let V be an optimal prefix-free machine. For an arbitrary total recursive function $f\colon \mathbb{N} \to \mathbb{N}$, assume that $\sum_{n=0}^{\infty} 2^{n-f(n)} < \infty$. In the case of $\alpha \in \mathbb{Q}$, the result is obvious. Thus, in what follows, we assume that $\alpha \notin \mathbb{Q}$ and therefore the base-two expansion of $\alpha - \lfloor \alpha \rfloor$ is unique and contains infinitely many ones.

Since $\sum_{n=0}^{\infty} 2^{n-f(n)} < \infty$, there exists $d_0 \in \mathbb{N}$ such that $\sum_{n=0}^{\infty} 2^{n-f(n)-d_0} \leq 1$. Hence, by the Kraft-Chaitin Theorem, i.e., Theorem 9, there exists a total recursive function $g\colon \mathbb{N} \to \{0,1\}^*$ such that (i) the function g is an injection, (ii) the set $\{\, g(n) \mid n \in \mathbb{N}\}$ is prefix-free, and (iii) $|g(n)| = f(n) - n + d_0$ for all $n \in \mathbb{N}$. On the other hand, since α is r.e., there exists a total recursive function $h\colon \mathbb{N} \to \mathbb{Q}$ such that $h(k) \leq \alpha$ for all $k \in \mathbb{N}$ and $\lim_{k\to\infty} h(k) = \alpha$.

Now, let us consider a prefix-free machine F such that, for every $n \in \mathbb{N}$ and $s \in \{0,1\}^*$, $g(n)s \in \mathrm{dom}\, F$ if and only if (i) $|s| = n$ and (ii) $0.s < h(k) - \lfloor \alpha \rfloor$ for some $k \in \mathbb{N}$. It is easy to see that such a prefix-free machine F exists. We then see that, for every $n \in \mathbb{N}$ and $s \in \{0,1\}^n$,

$$g(n)s \in \mathrm{dom}\, F \text{ if and only if } s \leq \alpha\restriction_n, \tag{1}$$

where s and $\alpha\restriction_n$ are regarded as a dyadic integer. Then, by the following procedure, we see that $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, F$ in query size $f(n) + d_0$.

Given $t \in \{0,1\}^*$, based on the equivalence (1), one determines $\alpha\restriction_n$ by putting the queries $g(n)s$ to the oracle $\mathrm{dom}\, F$ for all $s \in \{0,1\}^n$, where $n = |t|$. Note here that all the queries are of length $f(n) + d_0$, since $|g(n)| = f(n) - n + d_0$. One then accepts if $t = \alpha\restriction_n$ and rejects otherwise.

On the other hand, by Theorem 11, there exists $d \in \mathbb{N}$ such that $\mathrm{dom}\, F$ is reducible to $\mathrm{dom}\, V$ in query size $n + d$. Thus, by Proposition 1 (i), $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size $f(n) + d_0 + d$, as desired. □

We next prove Theorem 14 using Theorem 3 and the Ample Excess Lemma below.

Theorem 13 (Ample Excess Lemma, Miller and Yu [11]). *For every $\alpha \in \mathbb{R}$, α is weakly Chaitin random if and only if $\sum_{n=1}^{\infty} 2^{n-H(\alpha\restriction_n)} < \infty$.* □

Theorem 14. *Let α be a real which is weakly Chaitin random, and let V be an optimal prefix-free machine. For every order function f, if $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size f then $\sum_{n=0}^{\infty} 2^{n-f(n)} < \infty$.*

Proof. Let α be a real which is weakly Chaitin random, and let V be an optimal prefix-free machine. For an arbitrary order function f, assume that $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size f. Since f is an order function, $S_f = \{n \in \mathbb{N} \mid f(n) < f(n+1)\}$ is an infinite recursive set. Therefore there exists an increasing total recursive function $h\colon \mathbb{N} \to \mathbb{N}$ such that $h(\mathbb{N}) = S_f$. It is then easy to see that $f(n) = f(h(k+1))$ for every k and n with $h(k) < n \leq h(k+1)$. Thus, for each $k \geq 1$, we see that

$$\begin{aligned} \sum_{n=h(0)+1}^{h(k)} 2^{n-f(n)} &= \sum_{j=0}^{k-1} \sum_{n=h(j)+1}^{h(j+1)} 2^{n-f(n)} = \sum_{j=0}^{k-1} 2^{-f(h(j+1))} \sum_{n=h(j)+1}^{h(j+1)} 2^{n} \\ &= \sum_{j=0}^{k-1} 2^{-f(h(j+1))} \left(2^{h(j+1)+1} - 2^{h(j)+1}\right) < 2 \sum_{j=1}^{k} 2^{h(j)-f(h(j))}. \end{aligned} \tag{2}$$

On the other hand, let M be a deterministic Turing machine which computes V. For each $n \geq L_M$, we choose a particular p_n from I_M^n. Note that, given $(n, p_{f(n)})$ with $f(n) \geq L_M$, one can calculate the finite set $\mathrm{dom}\, V\restriction_{f(n)}$ by simulating the computation of M with the input q until at most the time step $T_M(p_{f(n)})$, for each $q \in \{0,1\}^*$ with $|q| \leq f(n)$. This can be possible because $T_M(p_{f(n)}) = T_M^{f(n)}$ for every $n \in \mathbb{N}$ with $f(n) \geq L_M$. Thus, since $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size f by the assumption, we see that there exists a partial recursive function $\Psi\colon \mathbb{N} \times \{0,1\}^* \to \{0,1\}^*$ such that, for all $n \in \mathbb{N}$ with $f(n) \geq L_M$, $\Psi(n, p_{f(n)}) = \alpha\restriction_n$. It follows from the optimality of U that $H(\alpha\restriction_n) \leq H(n, p_{f(n)}) + O(1)$ for all $n \in \mathbb{N}$ with $f(n) \geq L_M$. On the other hand, since the mapping $\mathbb{N} \ni k \mapsto f(h(k))$ is an increasing total recursive function, it follows also from the optimality of U that $H(h(k), s) \leq H(f(h(k)), s) + O(1)$ for all $k \in \mathbb{N}$ and $s \in \{0,1\}^*$. Therefore, using Theorem 3 we see that

$$H(\alpha\restriction_{h(k)}) \leq f(h(k)) + O(1) \tag{3}$$

for all $k \in \mathbb{N}$. Since α is weakly Chaitin random, using the Ample Excess Lemma, i.e., Theorem 13, we have $\sum_{n=1}^{\infty} 2^{n-H(\alpha\restriction_n)} < \infty$. Note that the function h is injective. Thus, using (3) we have

$$\sum_{j=1}^{\infty} 2^{h(j)-f(h(j))} \leq \sum_{j=1}^{\infty} 2^{h(j)-H(\alpha\restriction_{h(j)})+O(1)} \leq \sum_{n=1}^{\infty} 2^{n-H(\alpha\restriction_n)+O(1)} < \infty.$$

It follows from (2) that $\lim_{k\to\infty} \sum_{n=h(0)+1}^{h(k)} 2^{n-f(n)} < \infty$. Thus, since $2^{n-f(n)} > 0$ for all $n \in \mathbb{N}$ and $\lim_{k\to\infty} h(k) = \infty$, we have $\sum_{n=0}^{\infty} 2^{n-f(n)} < \infty$, as desired. □

4.2 The Proof of Theorem 7

The implication (ii) ⇒ (i) of Theorem 7 follows immediately from Proposition 2 and Proposition 1 (ii). On the other hand, the implication (i) ⇒ (ii) of Theorem 7 is proved as follows.

Proof (of (i) ⇒ (ii) of Theorem 7). Let V and W be optimal prefix-free machines, and let f be an order function. Suppose that there exists $c \in \mathbb{N}$ such that $\mathrm{dom}\, W$ is reducible to $\mathrm{Pf}(\Omega_V)$ in query size $f(n) + c$. Then, by considering the following procedure, we first see that $n < H(n, \Omega_V\restriction_{f(n)+c}) + O(1)$ for all $n \in \mathbb{N}$.

Given n and $\Omega_V\restriction_{f(n)+c}$, one first calculates the finite set $\mathrm{dom}\, W\restriction_n$. This is possible since $\mathrm{dom}\, W$ is reducible to $\mathrm{Pf}(\Omega_V)$ in query size $f(n) + c$ and $f(k) \leq f(n)$ for all $k \leq n$. Then, by calculating the set $\{\, W(p) \mid p \in \mathrm{dom}\, W\restriction_n \,\}$ and picking any one finite binary string s which is not in this set, one can obtain $s \in \{0,1\}^*$ such that $n < H_W(s)$.

Thus, there exists a partial recursive function $\Psi\colon \mathbb{N} \times \{0,1\}^* \to \{0,1\}^*$ such that, for all $n \in \mathbb{N}$, $n < H_W(\Psi(n, \Omega_V\restriction_{f(n)+c}))$. It follows from the optimality of W that

$$n < H(n, \Omega_V\restriction_{f(n)+c}) + O(1) \tag{4}$$

for all $n \in \mathbb{N}$.

Now, let us assume contrarily that the function $n - f(n)$ of $n \in \mathbb{N}$ is unbounded. Recall that f is an order function. Hence it is easy to show that there exists a total recursive function $g\colon \mathbb{N} \to \mathbb{N}$ such that the function $f(g(k))$ of k is increasing and the function $g(k) - f(g(k))$ of k is also increasing. For clarity, we define a total recursive function $m\colon \mathbb{N} \to \mathbb{N}$ by $m(k) = f(g(k)) + c$. Since m is injective, it is then easy to see that there exists a partial recursive function $\Phi\colon \mathbb{N} \to \mathbb{N}$ such that $\Phi(m(k)) = g(k)$ for all $k \in \mathbb{N}$. Therefore, based on the optimality of U, it is shown that $H(g(k), \Omega_V\restriction_{m(k)}) \leq H(\Omega_V\restriction_{m(k)}) + O(1)$ for all $k \in \mathbb{N}$. It follows from (4) that $g(k) < H(\Omega_V\restriction_{m(k)}) + O(1)$ for all $k \in \mathbb{N}$. On the other hand, we can show that $H(s) \leq |s| + H(|s|) + O(1)$ for all $s \in \{0,1\}^*$. Therefore we have $g(k) - f(g(k)) < H(m(k)) + O(1)$ for all $k \in \mathbb{N}$. Then, since the function $g(k) - f(g(k))$ of k is unbounded, it is easy to see that there exists a total recursive function $\Theta\colon \mathbb{N}^+ \to \mathbb{N}$ such that, for every $l \in \mathbb{N}^+$, $l \leq H(\Theta(l))$. It follows from the optimality of U that $l \leq H(l) + O(1)$ for all $l \in \mathbb{N}^+$. On the other hand, we can show that $H(l) \leq 2\log_2 l + O(1)$ for all $l \in \mathbb{N}^+$. Thus we have $l \leq 2\log_2 l + O(1)$ for all $l \in \mathbb{N}^+$. However, we have a contradiction on letting $l \to \infty$ in this inequality. This completes the proof. □

5 *T*-Convergent R.E. Reals

Let T be an arbitrary computable real with $0 < T \leq 1$. The parameter T plays a crucial role in the present paper.[2] In this section, we investigate the relation of T-convergent r.e. reals to the halting problems. In particular, Theorem 20 below is used to show Theorem 21 in the next section, and plays a major role in establishing the bidirectionality in the next section.

Recently, Calude, Hay, and Stephan [3] showed the existence of an r.e. real which is weakly Chaitin T-random and strictly T-compressible, in the case where T is a computable real with $0 < T < 1$, as follows.

Theorem 15 (Calude, Hay, and Stephan [3]). *Suppose that T is a computable real with $0 < T < 1$. Then there exist an r.e. real $\alpha \in (0,1)$ and $d \in \mathbb{N}$ such that, for all $n \in \mathbb{N}^+$, $|H(\alpha\restriction_n) - Tn| \leq d$.* □

We first show that the same r.e. real α as in Theorem 15 has the following property.

Theorem 16. *Suppose that T is a computable real with $0 < T < 1$. Let V be an optimal prefix-free machine. Then there exists an r.e. real $\alpha \in (0,1)$ such that α is weakly Chaitin T-random and $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size $\lfloor Tn \rfloor + O(1)$.* □

Calude, et al. [3] use Lemma 1 below to show Theorem 15. We also use it to show Theorem 16.

[2] The parameter T corresponds to the notion of "temperature" in the statistical mechanical interpretation of AIT introduced by Tadaki [16].

Lemma 1 (Reimann and Stephan [13] and Calude, Hay, and Stephan [3]). *Let T be a real with $T > 0$, and let V be an optimal prefix-free machine.*

(i) Suppose that $T < 1$. Then there exists $c \in \mathbb{N}^+$ such that, for every $s \in \{0,1\}^*$, there exists $t \in \{0,1\}^c$ for which $H_V(st) \geq H_V(s) + Tc$.

(ii) There exists $c \in \mathbb{N}^+$ such that, for every $s \in \{0,1\}^*$, $H_V(s0^c) \leq H_V(s) + Tc - 1$ and $H_V(s1^c) \leq H_V(s) + Tc - 1$. □

Proof (of Theorem 16). Suppose that T is a computable real with $0 < T < 1$. Let V be an optimal prefix-free machine. Then it follows from Lemma 1 that there exists $c \in \mathbb{N}^+$ such that, for every $s \in \{0,1\}^*$, there exists $t \in \{0,1\}^c$ for which

$$H_V(st) \geq H_V(s) + Tc. \tag{5}$$

For each prefix-free machine G and each $s \in \{0,1\}^*$, we denote by $S(G;s)$ the set $\{\, u \in \{0,1\}^{|s|+c} \mid s \text{ is a prefix of } u \;\&\; H_G(u) > T|u| \,\}$.

Now, we define a sequence $\{a_k\}_{k\in\mathbb{N}}$ of finite binary strings recursively on $k \in \mathbb{N}$ by $a_k := \lambda$ if $k = 0$ and $a_k := \min S(V; a_{k-1})$ otherwise. First note that a_0 is properly defined as λ and therefore satisfies $H_V(a_0) > T|a_0|$. For each $k \geq 1$, assume that $a_0, a_1, a_2, \ldots, a_{k-1}$ are properly defined. Then $H_V(a_{k-1}) > T|a_{k-1}|$ holds. It follow from (5) that there exists $t \in \{0,1\}^c$ for which $H_V(a_{k-1}t) \geq H_V(a_{k-1}) + Tc$, and therefore $a_{k-1}t \in \{0,1\}^{|a_{k-1}|+c}$ and $H_V(a_{k-1}t) \geq T|a_{k-1}t|$. Thus $S(V; a_{k-1}) \neq \emptyset$, and therefore a_k is properly defined. Hence, a_k is properly defined for every $k \in \mathbb{N}$. We thus see that, for every $k \in \mathbb{N}$, $a_k \in \{0,1\}^{ck}$, $H_V(a_k) > T|a_k|$, and a_k is a prefix of a_{k+1}. Therefore, it is easy to see that, for every $m \in \mathbb{N}^+$, there exists $k \in \mathbb{N}$ such that a_k contains m zeros. Thus, we can uniquely define a real $\alpha \in [0,1)$ by the condition that $\alpha\restriction_{ck} = a_k$ for all $k \in \mathbb{N}^+$. It follows that $H_V(\alpha\restriction_{ck}) > T|\alpha\restriction_{ck}|$ for all $k \in \mathbb{N}^+$. Note that there exists $d_0 \in \mathbb{N}$ such that, for every $s, t \in \{0,1\}^*$, if $|t| \leq c$ then $|H_V(st) - H_V(s)| \leq d_0$. Therefore, there exists $d_1 \in \mathbb{N}$ such that, for every $n \in \mathbb{N}^+$, $H_V(\alpha\restriction_n) > Tn - d_1$, which implies that α is weakly Chaitin T-random and therefore $\alpha \in (0,1)$.

Next, we show that $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size $\lceil Tn \rceil + O(1)$. For each $k \in \mathbb{N}$, we denote by F_k the set $\{s \in \{0,1\}^* \mid H_V(s) \leq \lfloor Tck \rfloor\}$. It follows that

$$a_k = \min\{\, u \in \{0,1\}^{ck} \mid a_{k-1} \text{ is a prefix of } u \;\&\; u \notin F_k \,\} \tag{6}$$

for every $k \in \mathbb{N}^+$. By the following procedure, we see that $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size $\lfloor Tn \rfloor + O(1)$.

Given $s \in \{0,1\}^*$ with $s \neq \lambda$, one first calculates the k_0 finite sets $F_1, F_2, \ldots, F_{k_0}$, where $k_0 = \lceil |s|/c \rceil$, by putting queries to the oracle $\mathrm{dom}\, V$. Note here that all the queries can be of length at most $\lfloor T(|s| + c) \rfloor$. One then calculates $a_1, a_2, \ldots, a_{k_0}$ in this order one by one from $a_0 = \lambda$ based on the relation (6) and $F_1, F_2, \ldots, F_{k_0}$. Finally, one accepts s if s is a prefix of a_{k_0} and rejects otherwise. This is possible since $\alpha\restriction_{ck_0} = a_{k_0}$ and $|s| \leq ck_0$.

Finally, we show that α is an r.e. real. Let $p_1, p_2, p_3, \ldots$ be a particular recursive enumeration of the infinite r.e. set $\mathrm{dom}\, V$. For each $l \in \mathbb{N}^+$, we define a prefix-free machine $V^{(l)}$ by the following two conditions (i) and (ii): (i)

$\mathrm{dom}\, V^{(l)} = \{p_1, p_2, \ldots, p_l\}$. (ii)$V^{(l)}(p) = V(p)$ for every $p \in \mathrm{dom}\, V^{(l)}$. It is easy to see that such prefix-free machines $V^{(1)}, V^{(2)}, V^{(3)}, \ldots$ exist. For each $l \in \mathbb{N}^+$ and each $s \in \{0,1\}^*$, note that $H_{V^{(l)}}(s) \geq H_V(s)$ holds, where $H_{V^{(l)}}(s)$ may be ∞. For each $l \in \mathbb{N}$, we define a sequence $\{a_k^{(l)}\}_{k\in\mathbb{N}}$ of finite binary strings recursively on $k \in \mathbb{N}$ by $a_k^{(l)} := \lambda$ if $k = 0$ and $a_k^{(l)} := \min(S(V^{(l)}; a_{k-1}^{(l)}) \cup \{a_{k-1}^{(l)} 1^c\})$ otherwise. It follows that $a_k^{(l)}$ is properly defined for every $k \in \mathbb{N}$. Note, in particular, that $a_k^{(l)} \in \{0,1\}^{ck}$ and $a_k^{(l)}$ is a prefix of $a_{k+1}^{(l)}$ for every $k \in \mathbb{N}$.

Let $l \in \mathbb{N}^+$. We show that $a_k^{(l)} \leq a_k$ for every $k \in \mathbb{N}^+$. To see this, assume that $a_{k-1}^{(l)} = a_{k-1}$. Then, since $H_{V^{(l)}}(s) \geq H_V(s)$ holds for every $s \in \{0,1\}^*$, based on the constructions of $a_k^{(l)}$ and a_k from $a_{k-1}^{(l)}$ and a_{k-1}, respectively, we see that $a_k^{(l)} \leq a_k$. Thus, based on the constructions of $\{a_k^{(l)}\}_{k\in\mathbb{N}}$ and $\{a_k\}_{k\in\mathbb{N}}$ we see that $a_k^{(l)} \leq a_k$ for every $k \in \mathbb{N}^+$.

We define a sequence $\{r_k\}_{k\in\mathbb{N}}$ of rationals by $r_k = 0.a_k^{(k)}$. Obviously, $\{r_k\}_{k\in\mathbb{N}}$ is a computable sequence of rationals. Based on the result in the previous paragraph, we see that $r_k \leq \alpha$ for every $k \in \mathbb{N}^+$. Based on the constructions of prefix-free machines $V^{(1)}, V^{(2)}, V^{(3)}, \ldots$ from V, it is also easy to see that $\lim_{k\to\infty} r_k = \alpha$. Thus we see that α is an r.e. real. □

Using Theorem 15 and Theorem 5 we can prove the following theorem.

Theorem 17. *Suppose that T is a computable real with $0 < T < 1$. For every r.e. real β, if β is T-convergent then β is strictly T-compressible.*

Proof. Suppose that T is a computable real with $0 < T < 1$. It follows from Theorem 15 that there exists an r.e. real α such that α is weakly Chaitin T-random and

$$H(\alpha\restriction_n) \leq Tn + O(1) \tag{7}$$

for all $n \in \mathbb{N}^+$. Since α is weakly Chaitin T-random, using the implication (i) $\Rightarrow$ (iii) of Theorem 5 we see that, for every T-convergent r.e. real β, there exists $d \in \mathbb{N}$ such that, for all $n \in \mathbb{N}^+$, $H(\beta\restriction_n) \leq H(\alpha\restriction_n) + d$. Thus, for each T-convergent r.e. real β, using (7) we see that $H(\beta\restriction_n) \leq Tn + O(1)$ for all $n \in \mathbb{N}^+$, which implies that β is strictly T-compressible. □

Using Theorem 7 of Tadaki [18], Theorem 17, and Theorem 4 (i), we can prove the following theorem.

Theorem 18. *Suppose that T is a computable real with $0 < T < 1$. Let V be an optimal prefix-free machine. Then there exists $d \in \mathbb{N}$ such that, for all $n \in \mathbb{N}^+$, $|H(Z_V(T)\restriction_n) - Tn| \leq d$.*

Proof. Suppose that T is a computable real with $0 < T < 1$. Let V be an optimal prefix-free machine. By Theorem 7 of Tadaki [18], $Z_V(T)$ is a T-convergent r.e. real. It follows from Theorem 17 that $Z_V(T)$ is strictly T-compressible. On the other hand, by Theorem 4 (i), $Z_V(T)$ is weakly Chaitin T-random. This completes the proof. □

Calude, et al. [3], in essence, showed the following result. For completeness, we include its proof.

Theorem 19 (Calude, Hay, and Stephan [3]). *If a real β is weakly Chaitin T-random and strictly T-compressible, then there exists $d \geq 2$ such that a base-two expansion of β has neither a run of d consecutive zeros nor a run of d consecutive ones.*

Proof. Let β be a real which is weakly Chaitin T-random and strictly T-compressible. Then there exists $d_0 \in \mathbb{N}$ such that, for every $n \in \mathbb{N}$,

$$|H(\beta\restriction_n) - Tn| \leq d_0. \tag{8}$$

On the other hand, by Lemma 1 (ii) we see that there exists $c \in \mathbb{N}^+$ such that, for every $s \in \{0,1\}^*$, $H(s0^c) \leq H(s) + Tc - 1$ and $H(s1^c) \leq H(s) + Tc - 1$. We choose a particular $k_0 \in \mathbb{N}^+$ with $k_0 > 2d$.

Assume first that a base-two expansion of β has a run of ck_0 consecutive zeros. Then $\beta\restriction_{n_0} 0^{ck_0} = \beta\restriction_{n_0+ck_0}$ for some $n_0 \in \mathbb{N}$. Thus we have $H(\beta\restriction_{n_0+ck_0}) - T(n_0 + ck_0) + k_0 \leq H(\beta\restriction_{n_0}) - Tn_0$, and therefore $-|H(\beta\restriction_{n_0+ck_0}) - T(n_0 + ck_0)| + k_0 \leq |H(\beta\restriction_{n_0}) - Tn_0|$ where we used the triangle inequality. It follows from (8) that $-d_0 + k_0 \leq d_0$ and therefore $k_0 \leq 2d_0$. This contradicts the fact that $k_0 > 2d$. Hence, a base-two expansion of β does not have a run of ck_0 consecutive zeros. In a similar manner we can show that a base-two expansion of β does not have a run of ck_0 consecutive ones, as well. □

Theorem 20. *Suppose that T is a computable real with $0 < T < 1$. Let V be an optimal prefix-free machine. For every r.e. real β, if β is T-convergent and weakly Chaitin T-random, then* $\mathrm{Pf}(\beta)$ *is reducible to* $\mathrm{dom}\, V$ *in query size* $\lfloor Tn \rfloor + O(1)$.

Proof. Suppose that T is a computable real with $0 < T < 1$. Let V be an optimal prefix-free machine. Then, by Theorem 16, there exist an r.e. real $\alpha \in (0,1)$ and $d_0 \in \mathbb{N}$ such that α is weakly Chaitin T-random and $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size $\lfloor Tn \rfloor + d_0$. Since α is an r.e. real which is weakly Chaitin T-random, it follow from the implication (i) $\Rightarrow$ (ii) of Theorem 5 that α is $\Omega(T)$-like.

Now, for an arbitrary r.e. real β, assume that β is T-convergent and weakly Chaitin T-random. Then, by Theorem 17, β is strictly T-compressible. It follows from Theorem 19 that there exists $c \geq 2$ such that the base-two expansion of β has neither a run of c consecutive zeros nor a run of c consecutive ones. On the other hand, since the r.e. real α is weakly Chaitin T-random, from the definition of $\Omega(T)$-likeness we see that α dominates β. Therefore, there are computable, increasing sequences $\{a_k\}_{k\in\mathbb{N}}$ and $\{b_k\}_{k\in\mathbb{N}}$ of rationals and $d_1 \in \mathbb{N}$ such that $\lim_{k\to\infty} a_k = \alpha$ and $\lim_{k\to\infty} b_k = \beta$ and, for all $k \in \mathbb{N}$, $\alpha - a_k \geq 2^{-d_1}(\beta - b_k)$ and $\lfloor \beta \rfloor = \lfloor b_k \rfloor$. Let $d_2 = d_1 + c + 2$. Then, by the following procedure, we see that $\mathrm{Pf}(\beta)$ is reducible to $\mathrm{dom}\, V$ in query size $\lfloor T(n + d_2) \rfloor + d_0$.

Given $s \in \{0,1\}^*$, one first calculates $\alpha\restriction_{n+d_2}$ by putting the queries t to the oracle $\mathrm{dom}\, V$, where $n = |s|$. This is possible since $\mathrm{Pf}(\alpha)$ is reducible to $\mathrm{dom}\, V$ in query size $\lfloor Tn \rfloor + d_0$. Note here that all the queries can be of length at most $\lfloor T(n + d_2) \rfloor + d_0$. One then find $k_0 \in \mathbb{N}$ such that $0.(\alpha\restriction_{n+d_2}) < a_{k_0}$. This is

possible since $0.(\alpha\restriction_{n+d_2}) < \alpha$ and $\lim_{k\to\infty} a_k = \alpha$. It follows that $2^{-(n+d_2)} > \alpha - 0.(\alpha\restriction_{n+d_2}) > \alpha - a_{k_0} \geq 2^{-d_1}(\beta - b_{k_0})$. Thus, $0 < \beta - b_{k_0} < 2^{-(n+c+2)}$. Let t be the first $n+c+2$ bits of the base-two expansion of the rational number $b_{k_0} - \lfloor b_{k_0} \rfloor$ with infinitely many zeros. Then, $|\, b_{k_0} - \lfloor b_{k_0} \rfloor - 0.t \,| \leq 2^{-(n+c+2)}$. It follows from $|\, \beta - \lfloor \beta \rfloor - 0.(\beta\restriction_{n+c+2}) \,| < 2^{-(n+c+2)}$ that $|\, 0.(\beta\restriction_{n+c+2}) - 0.t_n \,| < 3 \cdot 2^{-(n+c+2)} < 2^{-(n+c)}$. Hence, $|\, \beta\restriction_{n+c+2} - t \,| < 2^2$, where $\beta\restriction_{n+c+2}$ and t in $\{0,1\}^{n+c+2}$ are regarded as a dyadic integer. Thus, t is obtained by adding to $\beta\restriction_{n+c+2}$ or subtracting from $\beta\restriction_{n+c+2}$ a 2 bits dyadic integer. Since the base-two expansion of β has neither a run of c consecutive zeros nor a run of c consecutive ones, it can be checked that the first n bits of t equals to $\beta\restriction_n$. Thus, one accepts s if s is a prefix of t and rejects otherwise. Recall here that $|s| = n$. □

6 Bidirectionality

In this section we show the bidirectionality between $Z_U(T)$ and $\operatorname{dom} U$ with a computable real $T \in (0,1)$ in a general setting. Theorems 21 and 22 below are two of the main results of this paper.

Theorem 21 (Elaboration of $Z_U(T) \leq_{wtt} \operatorname{dom} U$). *Suppose that T is a computable real with $0 < T < 1$. Let V and W be optimal prefix-free machines, and let f be an order function. Then the following two conditions are equivalent:*

(i) $\mathrm{Pf}(Z_V(T))$ *is reducible to* $\operatorname{dom} W$ *in query size* $f(n) + O(1)$.
(ii) $Tn \leq f(n) + O(1)$. □

Theorem 22 (Elaboration of $\operatorname{dom} U \leq_{wtt} Z_U(T)$). *Suppose that T is a computable real with $0 < T \leq 1$. Let V and W be optimal prefix-free machines, and let f be an order function. Then the following two conditions are equivalent:*

(i) $\operatorname{dom} W$ *is reducible to* $\mathrm{Pf}(Z_V(T))$ *in query size* $f(n) + O(1)$.
(ii) $n/T \leq f(n) + O(1)$. □

Theorem 21 and Theorem 22 are proved in Subsection 6.1 and Subsection 6.2 below, respectively. Note that the function Tn in the condition (ii) of Theorem 21 and the function n/T in the condition (ii) of Theorem 22 are the inverse functions of each other. This implies that the computations between $\mathrm{Pf}(Z_V(T))$ and $\operatorname{dom} W$ are bidirectional in the case where T is a computable real with $0 < T < 1$. The formal proof is as follows.

Theorem 23. *Suppose that T is a computable real with $0 < T < 1$. Let V and W be optimal prefix-free machines. Then the computations between* $\mathrm{Pf}(Z_V(T))$ *and* $\operatorname{dom} W$ *are bidirectional.*

Proof. Let V and W be optimal prefix-free machines. It follows from the implication (ii) $\Rightarrow$ (i) of Theorem 22 that there exists $c \in \mathbb{N}$ for which $\operatorname{dom} W$ is reducible to $\mathrm{Pf}(Z_V(T))$ in query size f with $f(n) = \lfloor n/T \rfloor + c$. On the other hand, it follows from the implication (ii) $\Rightarrow$ (i) of Theorem 21 that there exists $d \in \mathbb{N}$ for which $\mathrm{Pf}(Z_V(T))$ is reducible to $\operatorname{dom} W$ in query size g with

$g(n) = \lfloor Tn \rfloor + d$. Since T is computable, f and g are order functions. For each $n \in \mathbb{N}$, we see that $g(f(n)) \leq Tf(n) + d \leq n + Tc + d$. Thus, the computation from $\mathrm{Pf}(\Omega_V)$ to $\mathrm{dom}\, W$ is not unidirectional. In a similar manner, we see that the computation from $\mathrm{dom}\, W$ to $\mathrm{Pf}(\Omega_V)$ is not unidirectional. This completes the proof. □

6.1 The Proof of Theorem 21

Let T be a computable real with $0 < T < 1$, and let V be an optimal prefix-free machine. Then, by Theorem 7 of Tadaki [18], $Z_V(T)$ is a T-convergent r.e. real. Moreover, by Theorem 4 (i), $Z_V(T)$ is weakly Chaitin T-random. Thus, the implication (ii) ⇒ (i) of Theorem 21 follows immediately from Theorem 20 and Proposition 1 (ii).

On the other hand, the implication (i) ⇒ (ii) of Theorem 21 follows immediately from Theorem 4 (i) and Theorem 24 below. In order to prove Theorem 24, we use Theorem 3.

Theorem 24. *Suppose that T is a computable real with $0 < T \leq 1$. Let β be a real which is weakly Chaitin T-random, and let V be an optimal prefix-free machine. For every order function f, if $\mathrm{Pf}(\beta)$ is reducible to $\mathrm{dom}\, V$ in query size f then $Tn \leq f(n) + O(1)$.*

Proof. Suppose that T is a computable real with $0 < T \leq 1$. Let β be a real which is weakly Chaitin T-random, and let V be an optimal prefix-free machine. For an arbitrary order function f, assume that $\mathrm{Pf}(\beta)$ is reducible to $\mathrm{dom}\, V$ in query size f. Let M be a deterministic Turing machine which computes V. For each n with $f(n) \geq L_M$, we choose a particular p_n from $I_M^{f(n)}$. Then, by the following procedure, we see that there exists a partial recursive function $\Psi\colon \mathbb{N} \times \{0,1\}^* \to \{0,1\}^*$ such that, for all n with $f(n) \geq L_M$,

$$\Psi(n, p_n) = \beta\restriction_n . \tag{9}$$

Given (n, p_n) with $f(n) \geq L_M$, one first calculates the finite set $\mathrm{dom}\, V\restriction_{f(n)}$ by simulating the computation of M with the input q until at most the time step $T_M(p_n)$, for each $q \in \{0,1\}^*$ with $|q| \leq f(n)$. This can be possible because $T_M(p_n) = T_M^{f(n)}$ for every n with $f(n) \geq L_M$. One then calculates $\beta\restriction_n$ using $\mathrm{dom}\, V\restriction_{f(n)}$ and outputs it. This is possible since $\mathrm{Pf}(\beta)$ is reducible to $\mathrm{dom}\, V$ in query size f.

It follows from (9) that

$$H(\beta\restriction_n) \leq H(n, p_n) + O(1) \tag{10}$$

for all n with $f(n) \geq L_M$.

Now, let us assume contrarily that the function $Tn - f(n)$ of $n \in \mathbb{N}$ is unbounded. Recall that f is an order function and T is computable. Hence it is easy to show that there exists a total recursive function $g\colon \mathbb{N} \to \mathbb{N}$ such that the function $f(g(k))$ of k is increasing and the function $Tg(k) - f(g(k))$ of k is also

increasing. Since the function $f(g(k))$ of k is injective, it is then easy to see that there exists a partial recursive function $\Phi\colon \mathbb{N} \to \mathbb{N}$ such that $\Phi(f(g(k))) = g(k)$ for all $k \in \mathbb{N}$. Thus, based on the optimality of U, it is shown that $H(g(k), s) \leq H(f(g(k)), s) + O(1)$ for all $k \in \mathbb{N}$ and $s \in \{0,1\}^*$. Hence, using (10) and Theorem 3 we have $H(\beta\restriction_{g(k)}) \leq H(f(g(k)), p_{g(k)}) + O(1) \leq f(g(k)) + O(1)$ for all k with $f(g(k)) \geq L_M$. Since β is weakly Chaitin T-random, we have $Tg(k) \leq H(\beta\restriction_{g(k)}) + O(1) \leq f(g(k)) + O(1)$ for all k with $f(g(k)) \geq L_M$. However, this contradicts the fact that the function $Tg(k) - f(g(k))$ of k is unbounded, and the proof is completed. □

6.2 The Proof of Theorem 22

The implication (i) ⇒ (ii) of Theorem 22 can be proved based on Theorem 18 as follows.

Proof (of (i) ⇒ (ii) of Theorem 22). In the case of $T = 1$, the implication (i) ⇒ (ii) of Theorem 22 results in the implication (i) ⇒ (ii) of Theorem 7. Thus, we assume that T is a computable real with $0 < T < 1$ in what follows. Let V and W be optimal prefix-free machines, and let f is an order function. Suppose that there exists $c \in \mathbb{N}$ such that $\mathrm{dom}\, W$ is reducible to $\mathrm{Pf}(Z_V(T))$ in query size $f(n) + c$. Then, by considering the following procedure, we first see that $n < H(n, Z_V(T)\restriction_{f(n)+c}) + O(1)$ for all $n \in \mathbb{N}$.

Given n and $Z_V(T)\restriction_{f(n)+c}$, one first calculates the finite set $\mathrm{dom}\, W\restriction_n$. This is possible since $\mathrm{dom}\, W$ is reducible to $\mathrm{Pf}(Z_V(T))$ in query size $f(n) + c$ and $f(k) \leq f(n)$ for all $k \leq n$. Then, by calculating the set $\{\, W(p) \mid p \in \mathrm{dom}\, W\restriction_n \}$ and picking any one finite binary string s which is not in this set, one can obtain $s \in \{0,1\}^*$ such that $n < H_W(s)$.

Thus, there exists a partial recursive function $\Psi\colon \mathbb{N} \times \{0,1\}^* \to \{0,1\}^*$ such that, for all $n \in \mathbb{N}$, $n < H_W(\Psi(n, Z_V(T)\restriction_{f(n)+c}))$. It follows from the optimality of W that

$$n < H(n, Z_V(T)\restriction_{f(n)+c}) + O(1) \tag{11}$$

for all $n \in \mathbb{N}$.

Now, let us assume contrarily that the function $n/T - f(n)$ of $n \in \mathbb{N}$ is unbounded. Recall that f is an order function and T is computable. Hence it is easy to show that there exists a total recursive function $g\colon \mathbb{N} \to \mathbb{N}$ such that the function $f(g(k))$ of k is increasing and the function $g(k)/T - f(g(k))$ of k is also increasing. For clarity, we define a total recursive function $m\colon \mathbb{N} \to \mathbb{N}$ by $m(k) = f(g(k)) + c$. Since m is injective, it is then easy to see that there exists a partial recursive function $\Phi\colon \mathbb{N} \to \mathbb{N}$ such that $\Phi(m(k)) = g(k)$ for all $k \in \mathbb{N}$. Therefore, based on the optimality of U, it is shown that $H(g(k), Z_V(T)\restriction_{m(k)}) \leq H(Z_V(T)\restriction_{m(k)}) + O(1)$ for all $k \in \mathbb{N}$. It follows from (11) that $g(k) < H(Z_V(T)\restriction_{m(k)}) + O(1)$ for all $k \in \mathbb{N}$. On the other hand, since T is a computable real with $0 < T < 1$, it follows from Theorem 18 that $H(Z_V(T)\restriction_n) \leq Tn + O(1)$ for all $n \in \mathbb{N}$. Therefore we have $g(k) < Tf(g(k)) + O(1)$ for all $k \in \mathbb{N}$. However, this contradicts the fact that the function $g(k)/T - f(g(k))$ of k is unbounded, and the proof is completed. □

On the other hand, the implication (ii) ⇒ (i) of Theorem 22 follows immediately from Theorem 25 below and Proposition 1 (ii).

Theorem 25. *Suppose that T is a computable real with $0 < T \leq 1$. Let V be an optimal prefix-free machine, and let F be a prefix-free machine. Then* $\mathrm{dom}\, F$ *is reducible to* $\mathrm{Pf}(Z_V(T))$ *in query size* $\lceil n/T \rceil + O(1)$.

Proof. In the case where $\mathrm{dom}\, F$ is a finite set, the result is obvious. Thus, in what follows, we assume that $\mathrm{dom}\, F$ is an infinite set.

Let $p_0, p_1, p_2, p_3, \ldots$ be a particular recursive enumeration of $\mathrm{dom}\, F$, and let G be a prefix-free machine such that $\mathrm{dom}\, G = \mathrm{dom}\, F$ and $G(p_i) = i$ for all $i \in \mathbb{N}$. Recall here that we identify $\{0,1\}^*$ with $\mathbb{N}$. It is also easy to see that such a prefix-free machine G exists. Since V is an optimal prefix-free machine, from the definition of optimality of a prefix-free machine there exists $d \in \mathbb{N}$ such that, for every $i \in \mathbb{N}$, there exists $q \in \{0,1\}^*$ for which $V(q) = i$ and $|q| \leq |p_i| + Td$. Thus, $H_V(i) \leq |p_i| + Td$ for every $i \in \mathbb{N}$. For each $s \in \{0,1\}^*$, we define $Z_V(T;s)$ as $\sum_{V(p)=s} 2^{-|p|/T}$. Then, for each $i \in \mathbb{N}$,

$$Z_V(T;i) \geq 2^{-H_V(i)/T} \geq 2^{-|p_i|/T-d}. \tag{12}$$

Then, by the following procedure, we see that $\mathrm{dom}\, F$ is reducible to $\mathrm{Pf}(Z_V(T))$ in query size $\lceil n/T \rceil + d$.

Given $s \in \{0,1\}^*$, one first calculates $Z_V(T)\restriction_{\lceil n/T \rceil + d}$ by putting the queries t to the oracle $\mathrm{Pf}(Z_V(T))$ for all $t \in \{0,1\}^{\lceil n/T \rceil + d}$, where $n = |s|$. Note here that all the queries are of length $\lceil n/T \rceil + d$. One then find $k_e \in \mathbb{N}$ such that $\sum_{i=0}^{k_e} Z_V(T;i) > 0.(Z_V(T)\restriction_{\lceil n/T \rceil + d})$. This is possible because $0.(Z_V(T)\restriction_{\lceil n/T \rceil + d}) < Z_V(T)$, $\lim_{k\to\infty} \sum_{i=0}^{k} Z_V(T;i) = Z_V(T)$, and T is a computable real. It follows that

$$\begin{aligned}\sum_{i=k_e+1}^{\infty} Z_V(T;i) &= Z_V(T) - \sum_{i=0}^{k_e} Z_V(T;i) < Z_V(T) - 0.(Z_V(T)\restriction_{\lceil n/T \rceil + d}) \\ &< 2^{-\lceil n/T \rceil - d} \leq 2^{-n/T-d}.\end{aligned}$$

Therefore, by (12), $\sum_{i=k_e+1}^{\infty} 2^{-|p_i|/T} \leq 2^d \sum_{i=k_e+1}^{\infty} Z_V(T;i) < 2^{-n/T}$. It follows that, for every $i > k_e$, $2^{-|p_i|/T} < 2^{-n/T}$ and therefore $n < |p_i|$. Hence, $\mathrm{dom}\, F\restriction_n = \{\, p_i \mid i \leq k_e \;\&\; |p_i| \leq n \,\}$. Thus, one can calculate the finite set $\mathrm{dom}\, F\restriction_n$. Finally, one accepts if $s \in \mathrm{dom}\, F\restriction_n$ and rejects otherwise. □

7 Concluding Remarks

Suppose that T is a computable real with $0 < T \leq 1$. Let V and W be optimal prefix-free machines. It is worthwhile to clarify the origin of the difference of the behavior of $Z_V(T)$ between $T = 1$ and $T < 1$ with respect to the notion of reducibility in query size f. In the case of $T = 1$, the Ample Excess Lemma [11] (i.e., Theorem 13) plays a major role in establishing the unidirectionality of the computation from Ω_V to $\mathrm{dom}\, W$. However, in the case of $T < 1$, this is not

true because the weak Chaitin T-randomness of a real α does not necessarily imply that $\sum_{n=1}^{\infty} 2^{Tn-H(\alpha\restriction_n)} < \infty$ [13]. On the other hand, in the case of $T < 1$, Lemma 1 (i) plays a major role in establishing the bidirectionality of the computations between $Z_V(T)$ and $\operatorname{dom} W$. However, this does not hold for the case of $T = 1$.

Acknowledgments. This work was supported by KAKENHI (23340020) and KAKENHI (23650001), by CREST from Japan Science and Technology Agency, and by the Ministry of Economy, Trade and Industry of Japan.

References

1. Calude, C.S.: Information and Randomness, 2nd edn. Revised and Extended. Springer, Heidelberg (2002)
2. Calude, C.S., Hertling, P.H., Khoussainov, B., Wang, Y.: Recursively enumerable reals and Chaitin Ω numbers. Theoret. Comput. Sci. 255, 125–149 (2001)
3. Calude, C.S., Hay, N.J., Stephan, F.C.: Representation of left-computable ε-random reals. J. Comput. Syst. Sci. 77, 812–819 (2011)
4. Calude, C.S., Nies, A.: Chaitin Ω numbers and strong reducibilities. Journal of Universal Computer Science 3(11), 1162–1166 (1997)
5. Calude, C.S., Staiger, L., Terwijn, S.A.: On partial randomness. Annals of Pure and Applied Logic 138, 20–30 (2006)
6. Calude, C.S., Stay, M.A.: Natural halting probabilities, partial randomness, and zeta functions. Inform. and Comput. 204, 1718–1739 (2006)
7. Chaitin, G.J.: A theory of program size formally identical to information theory. J. Assoc. Comput. Mach. 22, 329–340 (1975)
8. Chaitin, G.J.: Algorithmic Information Theory. Cambridge University Press, Cambridge (1987)
9. Downey, R.G., Hirschfeldt, D.R.: Algorithmic Randomness and Complexity. Springer, New York (2010)
10. Kučera, A., Slaman, T.A.: Randomness and recursive enumerability. SIAM J. Comput. 31(1), 199–211 (2001)
11. Miller, J., Yu, L.: On initial segment complexity and degrees of randomness. Trans. Amer. Math. Soc. 360, 3193–3210 (2008)
12. Nies, A.: Computability and Randomness. Oxford University Press, Inc., New York (2009)
13. Reimann, J., Stephan, F.: On hierarchies of randomness tests. In: Proceedings of the 9th Asian Logic Conference, August 16-19, World Scientific Publishing, Novosibirsk (2005)
14. Solovay, R.M.: Draft of a paper (or series of papers) on Chaitin's work... done for the most part during the period of September–December (1974); unpublished manuscript. IBM Thomas J. Watson Research Center, p. 215. Yorktown Heights, New York (May 1975)
15. Tadaki, K.: A generalization of Chaitin's halting probability Ω and halting self-similar sets. Hokkaido Math. J. 31, 219–253 (2002)

16. Tadaki, K.: A statistical mechanical interpretation of algorithmic information theory. In: Local Proceedings of Computability in Europe 2008 (CiE 2008), June 15-20, pp. 425–434. University of Athens, Greece (2008); An Extended Version Available at arXiv:0801.4194v1
17. Tadaki, K.: Chaitin Ω Numbers and Halting Problems. In: Ambos-Spies, K., Löwe, B., Merkle, W. (eds.) CiE 2009. LNCS, vol. 5635, pp. 447–456. Springer, Heidelberg (2009)
18. Tadaki, K.: Partial Randomness and Dimension of Recursively Enumerable Reals. In: Královič, R., Niwiński, D. (eds.) MFCS 2009. LNCS, vol. 5734, pp. 687–699. Springer, Heidelberg (2009)
19. Tadaki, K.: A computational complexity-theoretic elaboration of weak truth-table reducibility. Research Report of CDMTCS 406 (July 2011)

Computer Runtimes and the Length of Proofs

With an Algorithmic Probabilistic Application to Waiting Times in Automatic Theorem Proving

Hector Zenil

Dept. of Computer Science, University of Sheffield, UK
and Special Projects, Wolfram Research, Inc., USA
h.zenil@sheffield.ac.uk

Abstract. This paper is an experimental exploration of the relationship between the runtimes of Turing machines and the length of proofs in formal axiomatic systems. We compare the number of halting Turing machines of a given size to the number of provable theorems of first-order logic of a given size, and the runtime of the longest-running Turing machine of a given size to the proof length of the most-difficult-to-prove theorem of a given size. It is suggested that theorem provers are subject to the same non-linear tradeoff between time and size as computer programs are, affording the possibility of determining optimal timeouts and waiting times in automatic theorem proving. I provide the statistics for some small choices of parameters for both of these systems.

Keywords: halting problem, halting probability, proof length, automatic theorem proving, Busy Beaver problem, program-size complexity, small Turing machines.

1 Introduction

While profound connections between computer programs and mathematical proofs have been studied and are known (e.g. the Curry-Howard correspondence), little has been done to connect the two fields at the level of empirical practice. We present an experimental approach to the question of optimal proving times for automatic theorem provers, which bears out Calude and Stay's theoretical findings that programs either stop quickly or never halt [4].

Working with self-delimiting programs, that is, programs that are not the beginning of any other valid programs, Chaitin defined the complexity of the runtime of a program which eventually halts that we cannot effectively compute [5], and Calude and Stay have recently proven [4] that even though short programs can run for a very long time, long programs are the scarcest because most of them will stop rather quickly—if they ever do—depending on their length. Thus, the probability of a machine halting decreases the longer it takes to halt, if it ever does.

Just as Calude and Stay suggest that most Turing machines are fully determined qua termination by a small number of computational steps, and that the

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 224–240, 2012.

error margin drops drastically, in [8] we have also shown that Turing machines are fully determined qua extensionality by a small number of initial input values (a theoretical value for the error margin has yet to be determined but the very few data points that we could generate suggest to follow at least a polynomial distribution).

We undertake an experimental approach to the runtimes of deterministic Turing machines up to three states and two symbols in connection, and empirical evidence, to Calude and Stay's theoretical results. Then we undertake the same experimental approach to formulas of predicate calculus, in order to find some (if any) evidence in favour of a possible similar non-linear phenomenon in the distribution of proof lengths of (dis)proven theorems in random axiom systems and Turing machines.

Traditional intuition might make one think this an ill-fated approach. On the one hand because undecidability would interfere in any such experimental attempt, and on the other hand, because small systems may say more about design choices than about important results. Even though possible limiting effects may appear right away one can limitedly circumvent these limits (as the Busy Beaver problem does) in an effort tantamount to other interesting experiments including some of Calude's own interest [3] or of my own [7], this latter providing useful applications for the evaluation of the algorithmic complexity of short strings difficult to calculate with the other alternative (lossless compression algorithms). With the intuition one gets from studying small systems (see [13]), it seems worth it and insightful to undertake these kind of experiments.

1.1 The Halting Problem

The Halting Problem for Turing machines involves deciding whether an arbitrary Turing machine M eventually halts on an arbitrary input x. One can ask whether there is a Turing machine halt M which, given code (M) and the input x, eventually stops and produces 1 if $M(x)$ stops, and 0 if $M(x)$ does not stop. Turing's seminal result states that this problem cannot be solved by any Turing machine, i.e. there is no such halt M. Halting can be recognized by simply running the machine in question; the main difficulty is to detect non-halting machines.

Since many real-world problems arising in the fields of compiler optimization, automatized software engineering, formal proof systems, and so forth are deeply connected to the halting problem, there is an interest in understanding the problem in order to translate theoretical results into practical applications.

In [4], it was observed that for any computable probability distribution, most long times are effectively rare, so that at the limit they all had the same behavior regardless of the choice of distribution. They proved that the exact time at which a program stops is not too complicated algorithmically. It is (algorithmically) non-random because most programs either stop 'quickly' or never halt. Since non-random times are (effectively) rare, according to Calude and Stay, the density of times at which an N-bit program can stop decreases quickly.

2 The Busy Beaver Problem

There are $(4n+2)^{2n}$ possible $(n,2)$ deterministic Turing machines with n states and 2 symbols. We denote by (n,m) the class (or space) of all n-state m-symbol Turing machines having a bidirectional tape and remaining on the same cell when entering the (additional to n) halting state. Among the machines that halt, there are some that print more 1s on their output tapes than any other Turing machines of the same size, and some that reach a maximum number of steps upon halting.

If σ_T is the number of 1s on the tape of a Turing machine T upon halting, then: $\sum(n) = \max\{\sigma_T : T \in (n,2)\ T(n)\ halts\}$ with n the number of states of the Turing machine.

If t_T is the number of steps that a machine T takes upon halting, then $S(n) = \max\{t_T : T \in (n,2)\ T(n)\ halts\}$ with n the number of states of the Turing machine.

$\sum(n)$ and $S(n)$ are noncomputable functions [9] by reduction to the halting problem. Yet values are known for $(n,2)$ with $n \leq 4$. The solution for $(n,2)$ with $n < 3$ is trivial; the process leading to the solution in $(3,2)$ is discussed by Lin and Rado [11]; and the process leading to the solution in $(4,2)$ is discussed in [1].

Solving the halting problem for small machines. It is easy to see that $\sum(1) = 1$ and $\sum(2) = 4$. Lin and Rado [9] proved $\sum(3) = 6$ and Brady [1] that $\sum(4) = 13$. The exact known values for S are $S(1) = 1$, $S(2) = 6$, $S(3) = 21$, $S(4) = 107$. These Busy Beaver values are for 2-symbol Turing machines.

These numerical values of the Busy Beaver functions have been calculated by a combination of techniques, notably the exhaustive simulation of a reduced number of non-equivalent Turing machines, as it turns out that many can be decided (e.g. evident loops, etc) and because the number of cases is small enough one can either analyse case by case or actually run the machines and analyse their behaviour until deciding whether it halts or not. This is evidently possible because of the relatively small number of Turing machines with up to the number of states for for which the values of the Busy Beaver functions are known.

A program showing the evolution of all known Busy Beaver machines developed by this paper's authors is available online [15]. The formalism followed in this paper is the same as the one originally described and followed for the Busy Beaver problem as introduced by Rado [9].

It is worth noting that the Busy Beaver problem is defined for Turing machines with initial empty tapes, and Turing machines studied in this paper are all provided with an initially empty tapes too. Turing universality tells us, however, that for every Turing machine with an arbitrary input there is a Turing machine with empty input computing the same function, hence Turing machines with empty tapes cover all possible cases (the translation may only result in some extra states).

3 Halting and Runtime Distributions

Calude and Stay showed that "long-running" Turing machines can only halt at non-random times; the density of non-random times near n is about $1/n$. "Long-running" means that if we have a universal Turing machine U and machine M is implemented by a program m for U of length n, then $U(m)$ runs for more than $c \times 2^n$ steps, where c is some uncomputable constant depending on U.

3.1 Halting History of (2, 2) Turing Machines

We know that a machine halts if it enters the halting state before reaching the known Busy Beaver value $S(n)$. If it does not, then it never halts. The halting problem and the halting probability problem are closely related to the Busy Beaver problem in that a solution to any one of them would yield a solution to each of the others.

Consider the halting space of all $(2, 2)$ Turing machines (with an extra halting state) provided with an empty tape. The table in Fig. 1 shows the runtime distribution at which all machines in $(2, 2)$ halt (or do not).

t	k_t	$p(k_t)$
—	6544	0.65
1	2000	0.20
2	800	0.080
3	160	0.016
4	56	0.0056
5	362	0.036
6	78	0.0078

Fig. 1. Runtime distribution at which all machines halt (those that don't are indicated by "—"). Where t is the number of steps, k_t the number of machines that halted at t (out of a total of 3456 that halt), and $p(k_t)$ is the halting probability of a machine to halt (or not) in time t.

There are 10 000 2-state, 2-symbol Turing machines (the 10 000 figure comes simply from the formula giving the number of Turing machines with $n = 2$ states $(4n + 2)^{2n}$). No other Turing machine halts after 6 steps (see Fig. 1) in $(2, 2)$. Machines that never halt are 6544 in number, representing around .65 of the total.

What we term a runtime space is the product of a class of (n, m) Turing machines for fixed n and m, where programs are uniformly distributed, and the time space, which is discrete, has a halting time mapped to a greyscale color (the lighter the color, the sooner it halted; white means the program never halted and red means it reached the Busy Beaver value $S(n)$).

Each point in Fig. 3 represents a Turing machine and as defined by the corresponding spectrum in Fig. 2, the lighter the square the sooner it halted. White

Fig. 2. Halting color mapping spectrum for Turing machines in $(2, 2)$ (the last color is red, visible in the online and printed versions only)

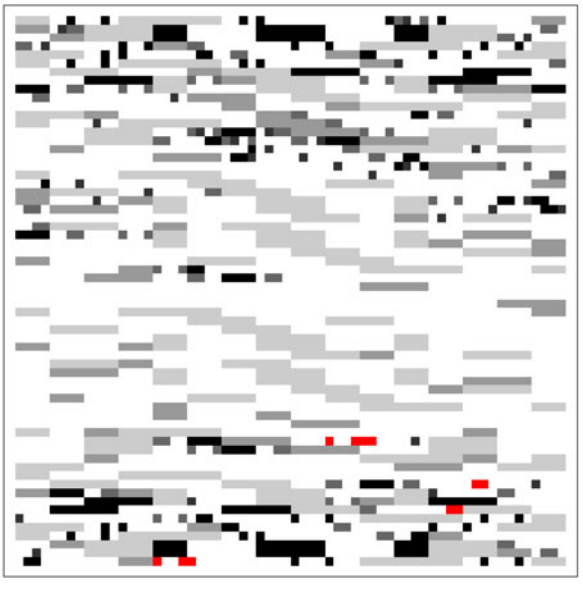


Fig. 3. Runtime distribution plot showing all the 10 000 Turing machines in $(2, 2)$ compressed in a Peano curve packing array (preserving the enumeration distance between machines). Some clusters may emerge due to the enumeration (e.g. terms involving transition rule parameters grouping Turing machines). The plot may look as if it had less than the necessary rows and columns to represent all the 10 000 Turing machines, but that is a consequence of the Peano packing, each apparent pixel is in fact a small cluster of several machines.

cells represent machines that don't halt. Red cells (only visible in the online and color printed versions) show the Busy Beaver machines (for this space, with runtime $S(2) = 6$ steps).

Among all the 3456 Turing machines in $(2, 2)$ that halt, .65 of them do so after the first step, .2 do so after the second, .05 after the third, and so on. In other words, .57 out of the 3456 $(2, 2)$ Turing machines that halted did so at the first step, .81 halted before or by the second step at the latest, .84 before or by the third step at the latest, and so on (see Fig. 6).

t	k_t	$100 \times 2^{14-t}$	$p(k_t)$
–	5382624		0.71
1	1075648	819200	0.14
2	614656	409600	0.082
3	263424	204800	0.035
4	97216	102400	0.013
5	53760	51200	0.0071
6	20800	25600	0.0028
7	12512	12800	0.0017
8	4264	6400	0.00057
9	2424	3200	0.00032
10	1064	1600	0.00014
11	536	800	0.000071
12	304	400	0.000040
13	176	200	0.000023
14	128	100	0.000017

Fig. 4. Where t is the number of steps, k_t the number of machines that halted at t, and $p(k_t)$ is the halting probability calculated from t and k_t. $100 \times 2^{14-t}$ is a good fit to the limit behavior as a function relating runtimes and the number of Turing machines halting at a certain runtime for the 14 runtimes at which Turing machines halt.

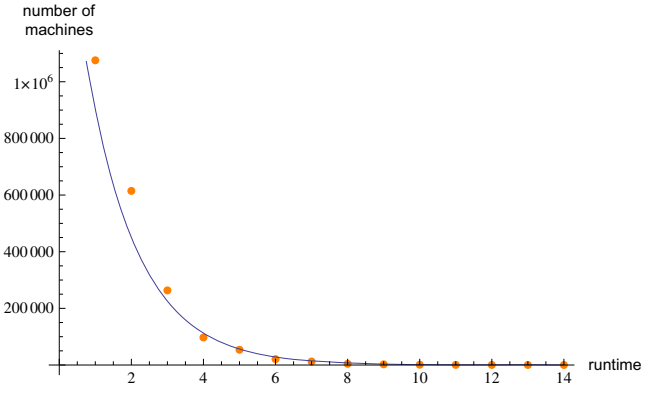


Fig. 5. Number of machines in $(3, 2)$ that halt step by step versus $100 \times 2^{14-t}$ (dark line (blue in color version))

3.2 Halting History of (3, 2) Turing Machines

Interesting output distribution facts:

- Out of 7 529 536 machines only 2 146 912 halt.
- There are 5 382 624 machines that do not halt.
- Those machines that halt only produce 126 different output strings, with the largest being 6 digits in length (the Busy Beavers).
- Exactly .2 of the Turing machines produce a 0 or a 1 as output.

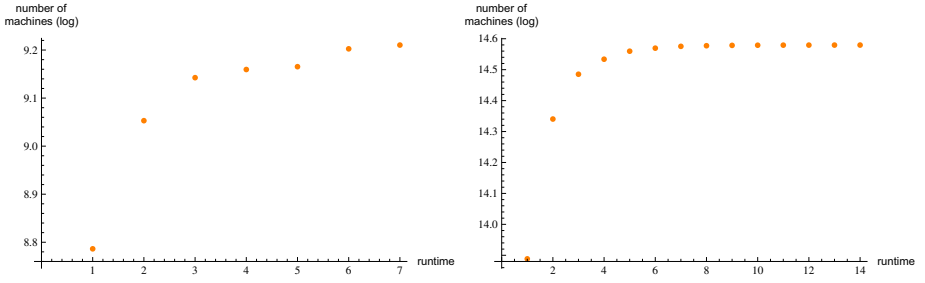


Fig. 6. Accumulated number of machines in $(2, 2)$ (left) and $(3, 2)$ (right) that halt step by step

The fact that the figures are mostly white and lightly colored is an indicator of the sparsity of non-halting or quickly-halting machines.

Fig. 7. Halting spectrum for $(3, 2)$. Last color in the spectrum is red (only visible in the online and color printed versions).

4 Gödel Meets Turing in the Computational Universe

Inspired by [13] where Wolfram undertakes an exhaustive investigation of the space of propositional logic formulas, I extended his ideas to investigate the space of first order logic. The extension wasn't trivial, among other reasons because unlike propositional calculus, predicate calculus is undecidable, meaning that one may come across cases where formulas (or their negations) are not proven or disproven in an axiom system of first order logic.

Proof lengths are, of course, not bounded, or one would be able to decide whether a formula in an axiom system can be proven or not if it has reached a limit. Frequency of proof lengths for randomly generated formulas, however, can be studied and analyzed. Frequency distributions of (dis)proven formulas turn out to follow a similar distribution to those of randomly generated computer programs, in which most programs, just as we found for formulas, halt (or are (dis)proven) quickly, with their number diminishing fast over time. When I met Cris Calude and became acquainted with his fascinating work, including a recent collaboration with Michael Stay on the distribution of halting times of random computer programs [4], it prompted me to seek connections with these other findings—persuaded as I was of the strong connections known to exist between computation and proof theory—and to undertake an empirical

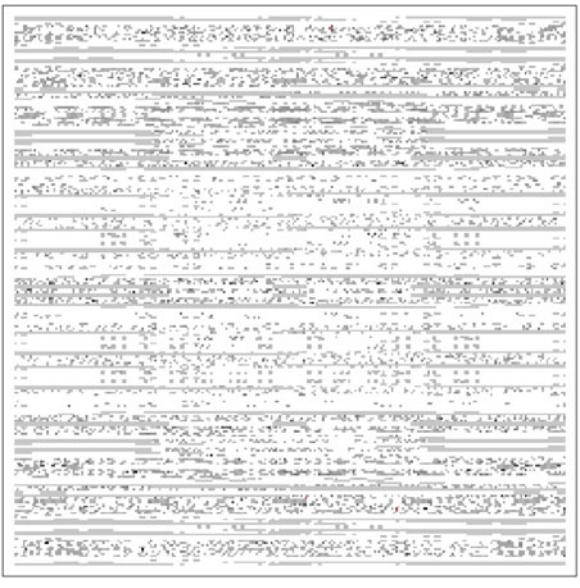


Fig. 8. Runtime deep field of a segment of runtimes from the 7 529 536 Turing machines in $(3, 2)$. The $(3, 2)$ Busy Beavers are barely visible as isolated red points (online and color printed versions only).

Fig. 9. This is what a typical random part of the runtime deep field looks like after a $10\times$ zoom from a 10th. square area of the original (Fig. 8) image.

investigation of both the halting runtimes of Turing machines that Calude and Stay had calculated theoretically, and the lengths of proofs found by automatic theorem provers.

It follows from Chaitin [5] and Calude and Stay [4] that to (dis)prove a formula in an axiom system one only needs to check up to the runtime for which the Turing machine encoding the proof no longer halts. Busy Beavers, as used in the previous section, are therefore relevant to automatic theorem proving because they provide an upper bound on the length of proofs. One only needs to run the computer to (dis)prove the formula up to the Busy Beaver value of the size of the Turing machine, and if it cannot be proven by then then it is undecidable for that axiom system. Moreover, Calude and Stay's work may then suggest that chances of proving a formula should decrease over time, or that if a formula can be (dis)proven it will likely do so early in time rather than later meaning that one can set an optimal time for a given provability certainty goal.

4.1 Computer Runtimes and Lengths of Proofs

Optimal proving times are relevant because, on the one hand, they may allow one to set a maximum waiting time, given that proofs may never arrive if a theorem is undecidable in an axiom system, but also because one would know how long to wait before giving up with a certain degree of certainty of provability. If one had a goal (say to prove a fraction of .90 of a set of formulas) one could calculate an optimal timeout and a maximum waiting time, taking advantage of the fact that in the case of theorem provers running on digital computers, there is a correspondence between runtime and proof length. The numbers involved are so large and grow so fast because of the combinatoric explosion (in the number of formulas as well as the number of Turing machines). We were only able to explore the tip of the iceberg of the space of all possible first-order formulas, but with interesting and encouraging results nonetheless.

4.2 Enumerating and Generating Predicate Calculus Axiom Systems with Equality

A number of sound and complete calculi have been developed enabling fully automated theorem provers for first-order logic. Equational logic is quite simple, and yet powerful [2]. Its atomic formulas are equations, making it very easy to encode and deal with. In our formalism, terms are first-order formulas built from variables and constants using function symbols. Equalities of the form $lhs = rhs$ are the atomic formulas in our language, where lhs and rhs are terms. One can represent most mathematical axiom systems and theorems in equational form, so it is expressively very rich. A logical system which possesses an explicitly stated set of axioms from which theorems can be derived is an axiomatic system.

In predicate calculus, a formula is in *prenex* normal form if it can be written as a string of quantifiers followed by a quantifier-free part. All first-order well-formed formulas (hereafter simply 'formulas') are logically equivalent to some formula in *prenex* normal form. *Skolemization* is a way of removing existential

quantifiers from a formula. Variables bound by existential quantifiers which are not within the scope of universal quantifiers can simply be replaced by the appropriate constants. Both will be used in order to enumerate all possible quantified axioms and formulas of first order logic.

All equational formulas can be represented with two binary operators f and p, where p is a pairing function and f is an indexing operator (any possible binary function). The first parameter of f will be a constant determining its index, while the second is any other term (variable, constant, f itself or p).

When the existential quantifier is inside a universal quantifier, the bound variable must be replaced by a Skolem function of the variables bound by universal quantifiers. We can then specify any constant using a formula of the form: $\forall_a \forall_b f(a,a) = f(b,b)$. And the $i-$th constant can be defined in terms of f and p recursively as follows:

```
c(0)=p(f(a, a), f(a, a))
c(n+1)=p(f(a, a), c(n))
```

Or in a single *Mathematica* expression:

```
Nest[p[f[a,a],#]&,p[a,a],i]
```

To represent all possible functions one can combine both f and p. For instance, $f(c(i), p(c(i), x))$ is the expression representing the i-th function (the function with index i) of x. This assumes that there are an infinite number of individuals in the most general case. Notice that x may be a list built from pairs.

Formulas were enumerated and generated by the number of variables and constants on both sides of the equality. There are no formulas of length 1, simply because an equality requires at least 2 terms on each side. Finally, all single axioms were arranged by length. The length of an equational formula is the sum of the bound variables on both sides of the equality. Axiom systems are simply all the possible subsets over the formulas of fixed length. Applying this operation makes the number of axiom systems to grow exponentially, so we were able to proceed exhaustively only up to 3 bound variables formulas and to generate a sample of 1000 axiom systems only (an initial segment) for 4 bound variables formulas. An automatic theorem prover was fed with all 4 bound variable single formulas as its proving goal for each of the generated axiom system, producing almost 10×10^3 proofs. Among the initial 1000 axiom systems, 607 were used only, as they were proven to be consistent (no axiom was the negation of any other) and independent (no axiom could be derived from the others).

An example of a formula with 3 bound variables is: $\forall_{x_1} \forall_{x_2} \forall_{x_3}, x_1 = f(f(x_2, x_3), x_1)$ and with four: $\forall_{x_1} \forall_{x_2} \forall_{x_3} \forall_{x_4}, x_1 = p(f(x_2, x_3), x_4)$. An example of an axiom system consisting of 2 axioms each with 2 bound variables is: $\forall_{x_1} \forall_{x_2}, x_1 = f(x_2, x_1) \wedge \forall_{x_1} \forall_{x_2}, x_1 = p(x_1, x_2)$. Notice that one does not need to further compose f with p or p with f in order to produce other possible formulas, because f is a general function with an index as first parameter and any term as second parameter which can be p or f itself, without the need of infinitely nesting each into the other in order to reach other possible constructions.

4.3 Experimental Setting

The project was undertaken using *Mathematica*'s built-in implementation of the well known and award-winning theorem prover *Waldmeister*[1]. *Waldmeister* returns True after evaluating an expression in *Mathematica* if it can prove the conclusions from the given axioms, and False if it can prove that the conclusions do not follow from the axioms. If it cannot prove either, it returns Unevaluated.

The axiom systems generated—as described in section 4.2—were first checked for logical consistency and internal axiom independence, these being two of the most important qualities of conventional mathematical axiom systems. A is said to be consistent if no theorem and its negation can be derived from A. On the other hand, if A is an axiom system and $a \in A$, then a is considered independent in A, or an independent axiom of A if a cannot be derived from $A - \{a\}$. As with any axiomatic system, we want this axiomatic system to be minimal, i.e. to contain no superfluous axiom. From this point on, only consistent axiom systems were taken into account.

Miscellaneous interesting first results:

- It was found that only .01 out of a total of 490 axiomatic systems with 1 or 2 axioms of length up to 3 bound variables were non-independent, i.e. one of its members could be derived from a combination of the others.
- All the 29 axiomatic systems of length 3 with 2 or more axioms were independent. This could be explained by the way in which the axiomatic systems were enumerated, because axioms closer to each other in the enumeration seem to have a better chance of being derived from each other. The condition of being a theorem or an axiom is evidently an arbitrary convention.
- The number of consistent axiom systems of length 3 was only .0342 percent of a total of 1024 initial axiomatic systems.
- In the case of axiom systems of length 4 (composed by formulas of that size), .607 of them were found to be consistent. This may be interpreted in two different ways: that even when the complexity of the axiom systems grows, the overall inconsistency does not increase, or else that the process only unveils the tip of the iceberg, where they are consistent chiefly due to their simplicity (both in terms of number of axioms per axiom system and the length of the axioms themselves, thereby reducing the possible number of clashes).

4.4 Distribution of Proof Lengths

The relation between the length of the formulas and the optimal runtime limit is of particular utility when no upper bound is known (or possible), when, for example, there are non-provable formulas for which longer runtimes will not make any difference—which, as verified herein, would cover a negligible number of cases.

[1] http://www.mpi-inf.mpg.de/~hillen/waldmeister/ (August, 2011).

A total of 89 145 formulas out of the 97 727 with at most 4 variables were proven to be theorems (or their negations) after a single step. One can call such a theorem *trivial* simply because its proof, requiring only 1 step, can be accomplished with an axiom, therefore itself being an axiom. The proof length (t) distribution (in percentage) of formulas with up to 4 variables is as shown in Fig. 10.

t	k_t	$p(k_t)$
1	89145	91.2184
2	2311	2.36475
3	473	0.484001
4	931	0.952654
5	928	0.949584
6	426	0.435908
7	577	0.59042
8	834	0.853398
9	1344	1.37526
10	294	0.300838
11	186	0.190326
12	206	0.210791
13	44	0.0450234
14	15	0.0153489
15	7	0.00716281
16	2	0.00204652
17	4	0.00409303

Fig. 10. Proof length (t) distribution (in percentage) of formulas with up to 4 variables

Proof length distribution of (dis)proven theorems. Where t is the number of steps the theorem prover has taken to produce the proof, k_t the number of machines that halted at t, and $p(k_t)$ is the halting probability of having (dis)proven k theorems in time t from which one can build a probability distribution $p(k_t)$.

It is worth noting that the behavior of 10 graph resembles the first case of $(2, 2)$ Turing machines, where the number of machines that halted was not strictly decreasing (unlike $(3, 2)$ that was monotonically decreasing).

Already 0.912 out of the total number of theorems are proven by the very first step, with that number dropping as the total is approached. From the distribution it follows that going beyond the 7th. step to the 17 steps that require the longest proofs only adds .012 new (dis)proven formulas to the total. Summary of proving times:

- A total of 89 145 formulas out of 97 727 were immediately proved (or disproved) after the first step (i.e. 91.21%).
- 95.96 were proven after 5 steps, and 96 969 formulas were proven after 9 steps (which is almost half of the 17 maximum number of steps reached by the formulas with 4 bound variables). That is, 99.22% of the total.
- Letting the theorem prover run up to 17 steps only generates 758 new proofs, that is only 0.77% of the total.

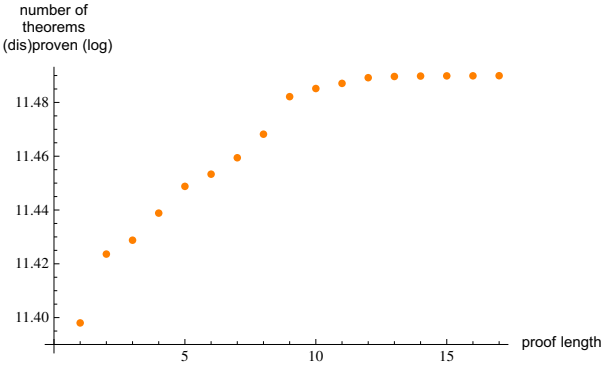


Fig. 11. Accumulated number of theorems (dis)proven step by step

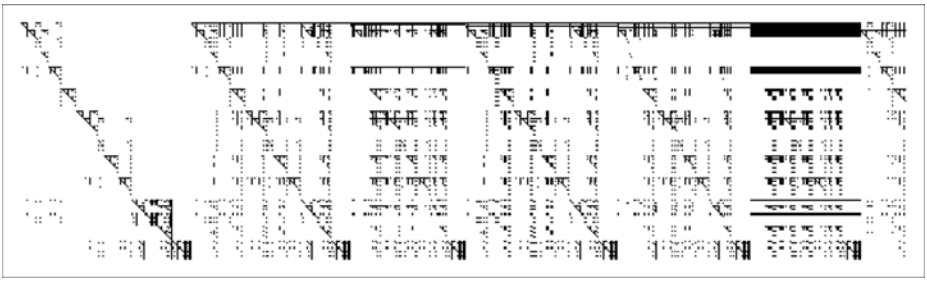


Fig. 12. Truth space of 97 727 proofs from the 607 consistent and independent axiom systems (x axis) against 161 formulas (y axis) from formulas with 4 bound variables. Every dot is a proof, a black square indicates that a particular theorem holds in a particular axiom system (which explains the diagonal, among other patterns) and white means the formula was proven to be false in the corresponding axiom system (i.e. the negation is a theorem). No undecidable candidate was found.

Fig. 13. Color mapping spectrum for proofs of length 4

As for Turing machines (see Fig. 8), the space of proof lengths (Fig. 14) is mostly white and lightly colored as an indicator of the sparsity of long proof lengths given that most formulas are (dis)proven very quickly, suggesting that the distribution of proof lengths follows the distribution of program runtimes.

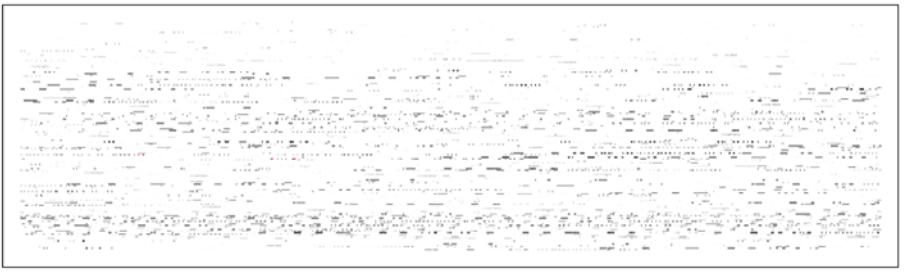


Fig. 14. Proof length deep field plot from the 97 727 formulas of up to 4 variables. *formula Busy Beavers* are barely visible as isolated red points (online and color printed versions only). Points are arranged as in 12.

5 Timeouts and Optimal Waiting Times

As for Busy Beaver Turing machines, the values of which depend on the size of the Turing machines (states and symbols), proof lengths depend on the length of the formulas. One can define *Busy Beaver formulas* (the values of which will be denoted by $fBB(n)$) as the formulas for which an automatic theorem prover takes more time to (dis)prove whether a theorem is decidable, or to produce the longest proof, among all the formulas of a fixed length. Unlike Turing machines, however, the size of a formula can take many forms, and may depend on the number of bound variables (as was the case in the experiments undertaken here), the number of logical operators or the number of symbols in general. It also depends on the formalism, just as Busy Beavers depend on the formalism used by Rado [9]. Following the analogy, the values of $fBB(n)$ would therefore work in a similar way and may be used just as Busy Beaver Turing machine values are currently used—for defining maximum runtimes and maximum output lengths for (small) Turing machines, saving time once an upper limit is known. The exact relation would also save considerable computational resources in automatic theorem proving.

As explained before, the theoretical algorithmic analysis in [4] indicates that a program that has not stopped after running for a long time has smaller and smaller chances of eventually stopping, so the longer the time t the more unlikely the program is to halt. Calude and Stay's results can be interpreted as follows: most Turing machines are fully determined qua termination by a small number of computational steps, and the error margin upon betting that a Turing machine will halt drops exponentially. Because proofs are programs for automatic theorem prover and one can connect this interpretation to the probability of a formula to be (dis)proven in an axiom system with a confidence error margin to be proven dropping fast.

Let the "optimal timeout" be the number of steps for which a fraction of formulas from a set of fixed length is (dis)proven. Evidently, proving time is asymptotically optimal, in the sense that the closest to the maximum runtime (the Busy

runtime t	(dis)proven fraction of theorems $p(t)$	$f(t) = 1/2^t$ (first significant digit)
1	0.9	0.5
2	0.02	0.2
3	0.005	0.1
4	0.01	0.06
5	0.009	0.03
6	0.004	0.02
7	0.006	0.008
8	0.009	0.004
9	0.01	0.002
10	0.003	0.001
11	0.002	0.0005
12	0.002	0.0002
13	0.0005	0.0001
14	0.0002	0.00006
15	0.00007	0.00003
16	0.00002	0.00002
17	0.00001	0.00001

Fig. 15. Runtime distribution at which all machines halt (those that don't are indicated by "—"). Where t is the number of steps, k_t the number of machines that halted at t (out of a total of 3456 that halt), and $p(k_t)$ is the halting probability calculated from t and k_t.

Beaver formula values), the greatest the fraction of (dis)proven formulas. An optimal time $OPTime$ for a given goal γ implies that upon t one has reached a fraction γ of (dis)proved formulas. Thus $OPTime(n, \gamma) = \min\{t(n) : |\alpha_{t(n)}| = \gamma\}$, where n is the length of the set of formulas, γ the desired fraction of (dis)proved formulas and $|\alpha|$ the number of formulas proven at time $t(n) \geq 0$. Obviously $0 < OPTime(n) \leq fBB(n)$ for each time $t > 0$, and $OPTime(n) = fBB(n)$ if $\gamma = 1$, that is, if the fraction of formulas to be (dis)proved is 1 (i.e. if the goal is to (dis)prove all the formulas of a fixed length).

Just as with Busy Beavers, the exact value of $OPTime(n)$ is uncomputable and unpredictable in general, but one can approach it. For example, in our formalism, for 4 bound variables it can be calculated from the probability distribution in 15. One can ascertain, for example, that from a uniform distribution of randomly generated formulas, nearly .90 of the formulas will be proven after the first step. And that the number of new proofs from then on will rapidly drop as a function of the number of steps. The value of $OPTime(n)$ can also determine a timeout for single formulas, given a confidence expectation. Which is to say that a single formula has, for example, a .90 chance of being (dis)proven in the first step, and that it has diminishing possibilities, if any, of being (dis)proven thereafter. We think that the results are robust enough to model specifications of theorem provers, despite not being completely independent. We were able to verify the results using another very different theorem prover, the Automatic

Proof Search or AProS [10] for propositional logic and predicate calculus (the theorem prover deals, however, with all sorts of other classical and non-classical calculus). AProS uses the intercalation method to search for normal natural deduction proofs not requiring a language in which the atomic formulas are identities, unlike *Waldmeister*. Notice that for this new case, the definition of the length of formulas was adjusted to the new framework, given that since the prover calculus does not require equality, no sense can be given to left or right hand sides. The set of randomly chosen operators used to generate formulas were the classic *and*, *or*, *implies* and *double implies*. AProS found proofs for .12 of the assertions (and for .353 of a set of assertions with no-double conditionals), out of a random choice of 1000 automatically generated predicate calculus assertions with up to 4 quantifiers, 3 general functions, 3 logical operators and 3 variables. The longest proof length (runtime) was 42 with an average proof length of 13, and a distribution very close to the one described by *Waldmeister* using *Mathematica*.

6 Concluding Remarks and Further Work

A logically significant question concerns the structure of the theorems established. If significant structural features are uncovered, then one could generate randomly formulas of that structure and repeat the proof length and runtime distribution experiments. It would be quite interesting, if one could find, for example, systematic biases for different theorem provers and theorem proving techniques when deviating in distribution from each other.

One can continue the process of generalizing theoretical results from computer programs to proof lengths and seek the equivalent of Busy Beavers in sets of well defined proofs and theorem provers. Just as for larger Busy Beaver Turing machine values, the computer time and resources to explore much larger sets of proofs are out of reach. The experiments suggest that the statistics for theorem proving times from randomly generated formulas may follow a similar trend to the distribution of runtimes of random computer programs. And that when searching for proofs, appropriate timeouts can be set and optimal waiting times defined depending on the size of the formulas as it has been determined that runtimes depend on the size of machines. It is too soon, however, to declare any true resemblance and there are always dangers of extrapolating from the behavior of small systems.

Acknowledgments. I am grateful to Cris Calude who encouraged me to publish these results in connection with his own work [4]. I am also indebted to Stephen Wolfram, Todd Rowland and Matthew Szudzik for their support and guidance during and after the 2005 NKS Summer School at Brown University, when I started this project as part of a 3-week Summer project and inspired by Stephen Wolfram's own work in [13], intending to extend his results from propositional logic to predicate calculus. I am also grateful to J.-P. Delahaye

with whom I've undertaken related research [7], studying the output distribution of abstract computing machines. To Wilfried Sieg for his guidance and for introducing me to AProS, which I used to strengthen the experimental results in this paper while a visiting scholar at Carnegie Mellon, and to Jeremy Avigad who brought me to Carnegie Mellon. And to the anonymous referee. Any error or omission remains, of course, the sole responsibility of this author.

References

1. Brady, A.H.: The Determination of the Value of Rado's Noncomputable Function for Four-State Turing Machines. Math. Comput. 40, 647–665 (1983)
2. Baumgartner, P., Zhang, H.: On Using Ground Joinable Equations in Equational Theorem Proving. In: Proceedings of the 3rd International Workshop on First Order Theorem Proving (St Andrews, Scotland), Fachberichte Informatik 5/2000, pp. 33–43. Universität Koblenz-Landau (2000)
3. Calude, C.S., Dinneen, M.J., Shu, C.-K.: Computing a glimpse of randomness. Experimental Mathematics 11(2), 369–378 (2002)
4. Calude, C.S., Stay, M.A.: Most programs stop quickly or never halt. Advances in Applied Mathematics 40, 295–308 (2005)
5. Chaitin, G.J.: Computing the Busy Beaver function. Information, Randomness & Incompleteness, 74–76 (1984)
6. Chaitin, G.J.: A theory of program size formally identical to information theory. J. ACM 22, 329–340 (1975)
7. Delahaye, J.-P., Zenil, H.: Numerical Evaluation of Algorithmic Complexity for Short Strings: A Glance Into the Innermost Structure of Randomness. Appl. Math. Comput. (in press, 2011)
8. Joosten, J., Soler-Toscano, F., Zenil, H.: Program-size Versus Time Complexity, Speed-up and Slowdown Phenomena in Small Turing Machines. International Journal of Unconventional Computing (2011)
9. Rado, T.: On Non-Computable Functions. Bell System Technical J. 41, 877–884 (1962)
10. Sieg, W.: The AProS Project: Strategic Thinking & Computational Logic. Logic Journal of the IGPL 15(4), 359–368 (2007)
11. Lin, S., Rado, T.: Computer Studies of Turing Machine Problems. J. ACM 12, 196–212 (1965)
12. Hillenbrand, T., Löchner, B.: The Next WALDMEISTER Loop. In: Voronkov, A. (ed.) CADE 2002. LNCS (LNAI), vol. 2392, pp. 486–500. Springer, Heidelberg (2002)
13. Wolfram, S.: A New Kind of Science. Wolfram Media (2002)
14. Zvonkin, A.K., Levin, L.A.: The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. Russian Math. Surveys 25(6), 83–124 (1970)
15. Zenil, H.: Busy Beaver, from the Wolfram Demonstrations Project (2009), http://demonstrations.wolfram.com/BusyBeaver/

Symmetry of Information: A Closer Look

Marius Zimand*

Department of Computer and Information Sciences,
Towson University, Baltimore, MD, USA
http://triton.towson.edu/~mzimand

Abstract. Symmetry of information establishes a relation between the information that x has about y (denoted $I(x : y)$) and the information that y has about x (denoted $I(y : x)$). In classical information theory, the two are exactly equal, but in algorithmical information theory, there is a small excess quantity of information that differentiates the two terms, caused by the necessity of packaging information in a way that makes it accessible to algorithms. It was shown in [Zim11] that in the case of strings with simple complexity (that is the Kolmogorov complexity of their Kolmogorov complexity is small), the relevant information can be packed in a very economical way, which leads to a tighter relation between $I(x : y)$ and $I(y : x)$ than the one provided in the classical symmetry-of-information theorem of Kolmogorov and Levin. We give here a simpler proof of this result.

1 Introduction

In classical information theory, the information that X has about Y is equal to the information that Y has about X, i.e.,

$$I(X : Y) = I(Y : X).^{[1]}$$

In algorithmical information theory, there exists a similar relation, but it is less perfect:

$$I(x : y) \approx I(y : x).$$

In this paper we take a closer look at the level of approximation indicated by "$\approx$." In the line above, x and y are binary strings of lengths n_x and, respectively, n_y, and $I(x : y)$, the information in x about y, is defined by $I(x : y) = C(y \mid n_y) - C(y \mid x, n_y)$, where $C(\cdot)$ is the plain Kolmogorov complexity. The precise form of the equation is given by the classical Kolmogorov-Levin theorem [ZL70], which states that

$$|I(x : y) - I(y : x)| = O(\log n_x + \log n_y). \tag{1}$$

This is certainly an important result, but in some situations, it is not saying much. Suppose, for example, that x has very little information about y, say,

* The author is supported in part by NSF grant CCF 1016158.

[1] X and Y are random variables and $I(X : Y) = H(Y) - H(Y \mid X)$, where H is Shannon entropy.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 241–246, 2012.

$I(x : y)$ is a constant (not depending on x and y). What does this tell about $I(y : x)$? Just from the definition and ignoring $I(x : y)$, $I(y : x)$ can be as large as $n_x - O(1)$, and, somewhat surprisingly, given that $I(x : y) = O(1)$, $I(y : x)$ can still have the same order of magnitude. As an example, consider y a c-random string (meaning $C(y \mid n_y) \geq n_y - c$), whose length n_y is also c-random. Let x be the binary encoding of n_y. Then $I(x : y) = C(y \mid n_y) - C(y \mid x, n_y) = O(1)$, but $I(y : x) = C(x \mid n_x) - C(x \mid y, n_x) = n_x - O(1)$. This example shows that even though, for some strings, the relation (1) is trivial, it is also tight.

In many applications, the excess term $O(\log n_x + \log n_y)$ does not hurt too much. But sometimes it does. For example, it is the reason why the Kolmogorov-Levin Theorem does not answer the following basic "direct product" question: If x and y are c-random n-bit strings and $I(x : y) \leq c$, does it follow that xy is $O(c)$-random?

The answer is positive (providing the finite analog of the van Lambalgen theorem). It follows from a recent result of the author [Zim11], which establishes a more precise symmetry-of-information relation for strings with *simple complexity*, i.e., for strings x such that $C^{(2)}(x \mid n_x)$ is small, say, bounded by a constant.[2] Note that all random strings have simple complexity, and that there are other types of strings with simple complexity as well. The main result in [Zim11] implies that if x and y are strings of equal length that have simple complexity bounded by c and $I(x : y) \leq c$, then $I(y : x) \leq O(c)$.

The proof method in [Zim11] is based on randomness extraction. Alexander Shen (personal communication) has observed that in fact a proof of the above result in [Zim11] can be obtained via the standard method used in the proof of Kolmogorov-Levin Theorem (see [DH10, LV08]). We present here such a proof.

The paper [Zim10a] provides several examples of counting arguments based on extractors.

We slightly extend the symmetry-of-information relation from [Zim11] to strings x and y that may have different lengths. We prove the following (with the convention that $\log 0 = 0$):

Theorem 1 (Main Theorem). *For all strings x and y satisfying the above complexity constraint,*

$$I(y : x) \leq I(x : y) + O(\log I(x : y)) + O(\log |n_x - n_y|) + \delta(x, y),$$

where $\delta(x, y) = O(C^{(2)}(x \mid n_x) + C^{(2)}(y \mid n_y))$.

Thus, for strings x and y with simple complexity, $I(y : x) \leq I(x : y) + O(\log I(x : y)) + O(\log |n_x - n_y|)$.

We next describe the main ideas in the proof.

1.1 Proof Techniques

The proofs of Symmetry of Information Theorems have a combinatorial nature. To fix some ideas, let us first sketch the proof of the classical Kolmogorov-Levin Theorem which establishes relation (1). The setting of the theorem, as

[2] The notation $C^{(2)}(x \mid n_x)$ is a shorthand for $C(C(x \mid n_x) \mid n_x)$.

well as for the rest of our discussion, is as follows: x and y are binary strings of lengths n_x and, respectively, n_y. An easy observation (see Section 2.2), shows that a relation between $I(y : x)$ and $I(x : y)$ can be deduced from a "chain rule:" $C(xy \mid n_x, n_y) \geq C(x \mid n_x) + C(y \mid x, n_y) -$ (small term). In our case, to obtain (1), it is enough to show that

$$C(xy \mid n_x, n_y) \geq C(x \mid n_x) + C(y \mid x, n_y) - O(\log n_x + \log n_y). \tag{2}$$

Let us consider a $2^{n_x} \times 2^{n_y}$ table with rows indexed by $u \in \{0,1\}^{n_x}$ and columns indexed by $v \in \{0,1\}^{n_y}$. Let $t = C(xy \mid n_x, n_y)$. We assign boolean values to the cells of the table as follows. The (u, v)-cell in the table is 1 if $C(uv \mid n_x, n_y) \leq t$ and it is 0 otherwise. The number of cells equal to 1 is less than 2^{t+1}, because there are only 2^{t+1} programs of length $\leq t$. Let m be such that the number of 1's in the x row is in the interval $(2^{m-1}, 2^m]$. Note that, given x, n_y and t, we can enumerate the 1-cells in the x-row and one of them is the (x, y) cell. It follows that

$$C(y \mid x, n_y) \leq m + O(\log t). \tag{3}$$

Now consider the set of rows that have at least 2^{m-1} 1s. The number of such rows is at most $2^{t+1}/2^{m-1} = 2^{t-m+2}$. We can effectively enumerate these rows if we are given n_x, n_y, m and t. Since the x row is one of these rows, it follows that

$$C(x \mid n_x) \leq t - m + O(\log n_y + \log m + \log t). \tag{4}$$

Adding equations (3) and (4) and keeping into account that $\log m \leq \log n_y$ and $\log t \leq O(\log n_x + \log n_y)$, the relation (2) follows.

A careful inspection of the proof reveals that the excess term $O(\log n_x + \log n_y)$ is caused by the fact that the enumerations involved in describing y and x need to know m (which is bounded by $C(y \mid n_y)$) and $t = C(xy \mid n_x, n_y)$. In case x and y are c-random, it is more economical to use randomness deficiencies. In particular instead of using $t = C(xy \mid n_x, n_y)$, we can do a similar argument based on the randomness deficiency of xy, i.e., $w = (n_x + n_y) - C(xy \mid n_x, n_y)$. This is an observation of Chang, Lyuu, Ti and Shen [CLTS10], who attribute it to folklore. For strings with simple complexity, it is advantageous to express t as $t = C(x \mid n_x) + C(y \mid x) - w$ because the first two terms have a short description. This yields a proof of the *Main Theorem* which we present in Section 3.

2 Preliminaries

2.1 Notation and Background on Kolmogorov Complexity

The Kolmogorov complexity of a string x is the length of the shortest effective description of x. There are several versions of this notion. We use here the *plain complexity*, denoted $C(x)$, and also the *conditional plain complexity* of a string x given a string y, denoted $C(x \mid y)$, which is the length of the shortest effective description of x given y. The formal definitions are as follows. We work over the binary alphabet $\{0,1\}$. A string is an element of $\{0,1\}^*$. If x is a string, n_x

denotes its length. Let M be a Turing machine that takes two input strings and outputs one string. For any strings x and y, define the *Kolmogorov complexity* of x conditioned by y with respect to M, as $C_M(x \mid y) = \min\{|p| \mid M(p, y) = x\}$. There is a universal Turing machine U with the following property: For every machine M there is a constant c_M such that for all x, $C_U(x \mid y) \leq C_M(x \mid y) + c_M$. We fix such a universal machine U and dropping the subscript, we write $C(x \mid y)$ instead of $C_U(x \mid y)$. We also write $C(x)$ instead of $C(x \mid \lambda)$ (where λ is the empty string). If n is a natural number, $C(n)$ denotes the Kolmogorov complexity of the binary representation of n. We use $C^{(2)}(x|n_x)$ as a shorthand for $C(C(x \mid n_x) \mid n_x)$. For two strings x and y, the information in x about y is denoted $I(x : y)$ and is defined as $I(x : y) = C(y \mid n_y) - C(y \mid x, n_y)$.

In this paper, the constant hidden in the $O(\cdot)$ notation only depends on the universal Turing machine. Also, by convention, $\log 0 = 0$.

2.2 Symmetry of Information and the Chain Rule

All the forms of the Symmetry of Information that we are aware of have been derived from the *chain rule* and this paper is no exception. The chain rule states that $C(xy \mid n_x, n_y) \approx C(x \mid n_x) + C(y \mid x, n_y)$. For us it is of interest to have an accurate estimation of the "$\approx$" relation. It is immediate to see that $C(xy \mid n_x, n_y) \leq C(y \mid n_y) + C(x \mid y, n_x) + 2C^{(2)}(y \mid n_y) + O(1)$. If we show a form of the converse inequality

$$C(xy \mid n_x, n_y) \geq C(x \mid n_x) + C(y \mid x, n_y) - (\text{small term}), \tag{5}$$

then we deduce that $C(x \mid n_x) - C(x \mid y, n_x) \leq C(y \mid n_y) - C(y \mid x, n_y) + 2C^{(2)}(y \mid n_y) + (\text{small term})$, i.e.,

$$I(y : x) \leq I(x : y) + 2C^{(2)}(y \mid n_y) + (\text{small term}).$$

Thus our efforts will be directed to establishing forms of the relation (5) in which (small term) is indeed small.

3 Proof of the Main Theorem

We demonstrate the *Main Theorem*, using the standard proof method of the Kolmogorov-Levin Theorem.

Let $t_x = C(x \mid n_x), t_y = C(y \mid x, n_y)$ and $t = C(xy \mid n_x, n_y)$. We take $w = t_x + t_y - t$, which is the term called *(small term)* in equation (5), and plays the role of randomness deficiency. Our goal is to show that

$$w = O(\log I(x : y)) + O(\log |n_x - n_y|) + \delta(x, y),$$

from which the Main Theorem follows (using the discussion in Section 2.2).

We assume that $w > 0$, otherwise there is nothing to prove. We will need to handle information $(|n_x - n_y|, t_x, t_y, w, b))$, where b is a bit that indicates

if $n_x > n_y$ or not. Let Λ be a string encoding in a self-delimiting way this information, and let λ be the length of Λ. Note that

$$\lambda \leq 2 \log w + O(C^{(2)}(x \mid n_x) + C^{(2)}(y \mid n_y) + \log I(x:y) + \log |n_x - n_y|).$$

We build a $2^{n_x} \times 2^{n_y}$ boolean table, with rows and columns indexed by the strings in $\{0,1\}^{n_x}$ and, respectively, $\{0,1\}^{n_y}$, and we set the value of cell (u,v) to be 1 if $C(uv \mid n_x, n_y) \leq t$, and to be 0 otherwise. Let S be the set of 1-cells, and S_u be the set of 1-cells in row u. Note that

$$|S| \leq 2^{t+1}.$$

Let m be defined as $2^{m-1} < |S_x| \leq 2^m$. We take F to be the set of "fat" rows, i.e., the set of rows having more than 2^{m-1} 1-cells. We have $|F| < |S|/2^{m-1} \leq 2^{t-m+2}$. Note that x is in F, and that the elements of F can be effectively enumerated given the information Λ and m. It follows that, given Λ and m, the string x can be described by its index in the enumeration of F. This index can be written on *exactly* $t - m + 2$ bits, so that knowing t which can be deduced from Λ, we can reconstruct m. It follows that

$$C(x \mid n_x, \Lambda) \leq t - m + 2 + O(1). \tag{6}$$

Next we note that y is in S_x and the elements of S_x can be enumerated given x and Λ. It follows that y can be described by its index in the enumeration of S_x. We obtain

$$C(y \mid x, \Lambda) \leq m + O(1). \tag{7}$$

From Equations (6) and (7), we conclude that

$$C(x|n_x, \Lambda) + C(y|x, \Lambda) \leq t + O(1) = t_x + t_y - w + O(1).$$

Note that $t_x = C(x \mid n_x) \leq C(x \mid n_x, \Lambda) + \lambda$ and $t_y = C(y \mid x, n_y) \leq C(y \mid x, \Lambda) + \lambda$. We obtain $t_x + t_y \leq t_x + t_y - w + 2\lambda$. Taking into account the estimation for λ, we have $w - 4 \log w = O(C^{(2)}(x \mid n_x) + C^{(2)}(y \mid n_y) + \log I(x:y) + \log |n_x - n_y|)$, from which, $w = O(C^{(2)}(x \mid n_x) + C^{(2)}(y \mid n_y) + \log I(x:y) + \log |n_x - n_y|)$, as desired.

Acknowledgements. I am grateful to Alexander Shen who has noticed that the main theorem can be obtained with the method used in the standard proof of the Kolmogorov-Levin Theorem.

References

[CLTS10] Chang, C., Lyuu, Y., Ti, Y., Shen, A.: Sets of k-independent sets. International Journal of Foundations of Computer Science 21(3), 321–327 (2010)

[DH10] Downey, R., Hirschfeldt, D.: Algorithmic randomness and complexity. Springer, Heidelberg (2010)

[FHP^+06] Fortnow, L., Hitchcock, J.M., Pavan, A., Vinodchandran, N.V., Wang, F.: Extracting Kolmogorov Complexity with Applications to Dimension Zero-One Laws. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 335–345. Springer, Heidelberg (2006)

[HPV09] Hitchcock, J., Pavan, A., Vinodchandran, N.V.: Kolmogorov complexity in randomness extraction. In: Electronic Colloquium on Computational Complexity (ECCC), pp. 09–071 (2009)

[LV08] Li, M., Vitanyi, P.: An introduction to Kolmogorov complexity and its applications, 3rd edn. Springer, Heidelberg (2008); 1st edn. (1993)

[Zim10a] Zimand, M.: Counting Dependent and Independent Strings. In: Hliněný, P., Kučera, A. (eds.) MFCS 2010. LNCS, vol. 6281, pp. 689–700. Springer, Heidelberg (2010)

[Zim10b] Zimand, M.: Possibilities and impossibilities in Kolmogorov complexity extraction. SIGACT News 41(4) (December 2010)

[Zim11] Zimand, M.: Symmetry of information and bounds on nonuniform randomness extraction via Kolmogorov complexity. In: Proceedings 26th IEEE Conference on Computational Complexity, pp. 148–156 (June 2011)

[ZL70] Zvonkin, A., Levin, L.: The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms. Russian Mathematical Surveys 25(6), 83–124 (1970)

How Much Information Can There Be in a Real Number?*

Gregory Chaitin

IBM T.J. Watson Research Center
P.O. Box 218, Yorktown Heights, NY 10598, USA
gjchaitin@gmail.com

Abstract. This note gives some information about the magical number Ω and why it is of interest. Our purpose is to explain the significance of recent work by Calude and Dinneen attempting to compute Ω. Furthermore, we propose measuring human intellectual progress (not scientific progress) via the number of bits of Ω that can be determined at any given moment in time using the current mathematical theories.

Keywords: halting problem, halting probability.

1 Introduction

A real number corresponds to the length of a line segment that is measured with infinite precision. A rational number has a periodic decimal expansion. For example,

$$\frac{1}{3} = 0.3333333...$$

The decimal expansion of an irrational real number is not periodic. Here are three well-known irrational reals that everyone encounters in high school and college mathematics: $\sqrt{2}$, π, and e.

Each of these numbers would seem to contain an infinite amount of information, because they have an infinite decimal expansion that never repeats. For example,

$$\pi = 3.1415926...$$

However, π actually only contains a finite amount of information, because there is a small computer program for computing π. Instead of sending someone the digits of π, we can just explain to them how to compute as many digits as they want.

Are there any real numbers that contain an infinite amount of information? Well, clearly, if the successive decimal digits are chosen at random, the resulting

* This paper has originally appeared in *International Journal of Bifurcation and Chaos* 17(6), 1933–1935 (©2007, WSPC). It is reprinted with kind permission of World Scientific Publishing Company.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 247–251, 2012.

stream of digits has no structure, each digit is a complete surprise, and there cannot be an algorithm for computing the number digit by digit.

However, this random sequence of digits is not **useful** information, not at all. It's an infinite amount of completely useless information.

2 Borel's Know-It-All Real Number

In 1927, the French mathematician Emile Borel pointed out that there are real numbers which contain an infinite amount of extremely **useful** information. The particular example that he gave is defined like this: Its Nth digit answers the Nth yes/no question in an infinite list of all possible yes/no questions, questions about the weather, the stock market, history, the future, physics, mathematics... Here I am talking about the Nth digit after the decimal point. Borel's number is between zero and one; there is nothing before the decimal point, only stuff after the decimal point. And we can assemble this list of questions because the set of all possible questions is what mathematicians call a countable or a denumerable set.

3 Using a Real Number as an Oracle for the Halting Problem

Borel's real number may seem rather unreal, rather fantastic, even though it exists in some Platonic, ideal, conceptual sense. How about a more realistic example, and now let's use base two, not base ten. Well, there is a real Θ whose Nth bit tells us whether or not the Nth computer program ever halts. This time we imagine an infinite list of all possible self-contained computer programs—not yes/no questions—and ask which programs will eventually finish running. This is Alan Turing's famous 1936 halting problem.

Θ doesn't tell us anything about the stock market or history, but it does tell us a great deal about mathematics. Why? Because knowing this number Θ would automatically enable us to resolve famous mathematical problems like Fermat's so-called last theorem, which asserts that there are no positive integer solutions for

$$x^N + y^N = z^N$$

with the power N greater than two.

How can Θ enable us to decide if Fermat was right and this equation has no solutions? There is a simple computer program for systematically searching for a solution of Fermat's equation. This program will fail to halt precisely if Fermat's conjecture that there are no solutions is correct.

However, in the case of Fermat's conjecture there is no need to wait for the number Θ; Andrew Wiles now has a proof that there are no solutions. But Θ would enable us to answer an infinite number of such conjectures, more precisely, all conjectures that can be refuted by a single counter example that we can search for using a computer.

4 N Cases of the Halting Problem Is Only $\log_2 N$ Bits of Information

So knowing the answers to individual cases of the halting problem can be valuable information, and Θ enables us to answer **all** such problems, but unfortunately not in an optimal way. Θ isn't optimal, it is highly redundant, we're wasting lots of bits. Individual answers to the halting problem aren't independent, they're highly correlated.

Why? Because if we are given N programs, we can determine which ones halt and which ones don't if we merely know **how many** of these N programs halt, and to know that is only about $\log_2 N$ bits of information. (Run all N programs in parallel until precisely the correct number have stopped; the remaining programs will never stop.)

Furthermore, $\log_2 N$ is much smaller than N for all sufficiently large values of N.

So what is the best we can do? Is there an oracle for the halting problem that isn't redundant, that doesn't waste any bits?

5 The Halting Probability Ω Is the Most Compact Oracle for the Halting Problem

The best way to pack information about the halting problem into a real number is to know a great many bits of the numerical value of the probability that a program chosen at random will eventually halt. Precisely how do I define this halting probability? Well, the exact definition is a little complicated, and in fact the numerical value of Ω depends on the particular computer and the programming language that you pick.

The general idea is that the computer that we are using flips a fair coin to generate each bit of the program, a heads yields a 1, a tails yields a 0, successive coin tosses are independent, and the computer starts running the program right away as it generates these bits. Ω is the probability that this process will eventually halt.

More precisely, each K-bit program p that halts contributes precisely $1/2^K$ to the halting probability Ω:

$$\Omega = \Sigma_{p \text{ halts}}\, 2^{-(\text{the size of } p \text{ in bits})}.$$

Furthermore, to avoid having this sum diverge to infinity, the set of meaningful programs must be a prefix-free set, in other words, no extension of a valid program is a valid program. Then what information theorists call the Kraft inequality applies to the set of all programs and Ω is necessarily less than one.

Ω is a very valuable oracle, because knowing the first N bits of Ω would enable us to resolve the halting problem for all programs up to N bits in size. No oracle for the halting problem can do better than this. Ω is so valuable

precisely because it is the most compact way to represent this information. It's the best possible oracle for the halting problem. You get the biggest bang for your buck with each bit!

And because this information is so valuable, Ω is maximally unknowable, maximally uncomputable: An N-bit computer program can compute at most N bits of Ω, and a mathematical theory with N bits of axioms can enable us to determine at most N bits of Ω. In other words, the bits of Ω are incompressible, irreducible information, both logically irreducible and computationally irreducible.

Paradoxically, however, even though Ω is packed full of useful information, its successive bits **appear** to be totally unstructured and random, totally chaotic, because otherwise Ω would not be the most compact oracle for the halting problem. If one could predict future bits from past bits, then Ω would not be the best possible compression of all the answers to individual cases of Turing's halting problem.

6 Measuring Mathematical or Human Intellectual Progress in Terms of Bits of Ω

Counting how many bits of Ω our current mathematical theories permit us to know, gives us a way to measure the complexity of our mathematical knowledge as a function of time. Ω is infinitely complex, and at any given moment our theories capture at most a finite amount of this complexity. Our minds are finite, not infinitely complex like Ω.

But what if we bravely try to compute Ω anyway?

7 Storming the Heavens: Attempting to Compute the Uncomputable Bits of Ω

This amounts to a systematic attempt to increase the complexity of our mathematical knowledge, and it is precisely what Calude and Dinneen try to do in [1]. As they show, you can start off well enough and indeed determine a few of the initial bits of Ω. But as I have tried to explain, the further you go, the more creativity, the more ingenuity is required. To continue making progress, you will eventually need to come up with more and more complicated mathematical principles, novel principles that are not consequences of our current mathematical knowledge.

Will mathematics always be able to advance in this way, or will we eventually hit an insurmountable obstacle? Who knows! What is clear is that Ω can never be known in its entirety, but if the growth of our mathematical knowledge continues unabated, each individual bit of Ω can eventually be known.

I hope that this note gives some idea why [1] is of interest. (See also [2].[1]) For more on Ω, please see my article in *Scientific American* [3] or my book [4]. A more recent paper is my Enriques lecture at the University of Milan in 2006 [5].

References

1. Calude, C.S., Dinneen, M.J.: Exact approximations of omega numbers. Int. Journal of Bifurcation & Chaos 17(6), 1937–1954 (2007)
2. Calude, C.S., Calude, E., Dinneen, M.J.: A new measure of the difficulty of problems. Journal of Multiple-Valued Logic and Soft Computing 12, 285–307 (2006)
3. Chaitin, G.: The limits of reason. Scientific American 294(3), 74–81 (2006)
4. Chaitin, G.: Meta Math! Pantheon, New York (2005); Meta Maths. Atlantic Books, London (2006)
5. Chaitin, G.: The halting probability Ω: Irreducible complexity in pure mathematics. Milan Journal of Mathematics 75, 291–304 (2007)

[1] Editors note: The complexity of mathematical problems was developed in a series of papers including C. S. Calude, E. Calude. Evaluating the complexity of mathematical problems. Part 1, *Complex Systems* 18 (2009), 267–285, Part 2 *Complex Systems* 18 (2010), 387–401, and C. S. Calude, E. Calude and K. Svozil. The complexity of proving chaoticity and the Church-Turing Thesis, *Chaos* 20 037103 (2010), 1–5.

Mathematics, Metaphysics and the Multiverse

S. Barry Cooper*

School of Mathematics, University of Leeds, Leeds LS2 9JT, U.K.
pmt6sbc@leeds.ac.uk
http://www.amsta.leeds.ac.uk/~pmt6sbc/

Abstract. It would be nice if science answered all questions about our universe. In the past, mathematics has not just provided the language in which to frame suitable scientific answers, but was also able to give us clear indications of its own limitations. The former was able to deliver results via an ad hoc interface between theory and experiment. But to characterise the power of the scientific approach, one needs a parallel higher-order understanding of how the working scientist uses mathematics, and the development of an informative body of theory to clarify and expand this understanding. We argue that this depends on us selecting mathematical models which take account of the 'thingness' of reality, and puts the mathematics in a correspondingly rich information-theoretic context. The task is to restore the role of embodied computation and its hierarchically arising attributes. The reward is an extension of our understanding of the power and limitations of mathematics, in the mathematical context, to that of the real world. Out of this viewpoint emerges a widely applicable framework, with not only epistemological, but also ontological consequences – one which uses Turing invariance and its putative breakdowns to confirm what we observe in the universe, to give a theoretical status to the dichotomy between quantum and relativistic domains, and which removes the need for many-worlds and related ideas. In particular, it is a view which confirms that of many quantum theorists – that it is the quantum world that is 'normal', and our classical level of reality which is strange and harder to explain. And which complements fascinating work of Cristian Calude and his collaborators on the mathematical characteristics of quantum randomness, and the relationship of 'strong determinism' to computability in nature.

Academics have ever been able to build successful (and very useful) careers within the bounds of 'normal science', and to revel in a near-Laputian unworldliness. But science would not progress half so well without a leavening of the less conventional, and an occasional engagement with the real-world. Particularly appropriate to the Turing Centenary, this short piece is dedicated to Cristian Calude on his sixtieth birthday—a researcher not just of formidable expertise, but one whose innovative work on real problems related to physics and randomness has much enriched our understanding of the world around us. This relevance

* Preparation of this article supported by E.P.S.R.C. Research Grant No. EP/G000212.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 252–267, 2012.

is unusual. The risk-taking, and willingness to work with uncertainty, of 'post-normal science' (as defined by Silvio Funtowicz and Jerome Ravetz [22]) is not for everyone.

1 The World of Science: Two Fundamental Issues

Max Born starts his 1935 book [6] on "The Restless Universe" with the words:

> It is odd to think that there is a word for something which, strictly speaking, does not exist, namely, "rest". ... What seems dead, a stone or the proverbial "door-nail", say, is actually for ever in motion. We have merely become accustomed to judge by outward appearances; by the deceptive impressions we get through our senses.

Here from Nassim Taleb's 2007 book [37] on "The Black Swan" is a more disturbing take on the underlying complexity of our world:

> I have spent my entire life studying randomness, practicing randomness, hating randomness. The more that time passes, the worse things seem to me, the more scared I get, the more disgusted I am with Mother Nature. The more I think about my subject, the more I see evidence that the world we have in our minds is different from the one playing outside. Every morning the world appears to me more random than it did the day before, and humans seem to be even more fooled by it than they were the previous day. It is becoming unbearable. I find writing these lines painful; I find the world revolting.

Back in 1935, Max Born ended his book:

> The scientist's urge to investigate, like the faith of the devout or the inspiration of the artist, is an expression of mankind's longing for something fixed, something at rest in the universal whirl: God, Beauty, Truth.
>
> Truth is what the scientist aims at. He finds nothing at rest, nothing enduring, in the universe. Not everything is knowable, still less is predictable. But the mind of man is capable of grasping and understanding at least a part of Creation; amid the flight of phenomena stands the immutable pole of law.

These at first sight rather different world-views are actually quite consistent. A careful reading (helped by a familiarity with the books) reveals that the authors are talking about *different things*. And it is not just the difference between economics and quantum mechanics.

Nassim Nicholas Taleb was a successful Wall Street trader who went on to prophetically warn of the uncertainties underlying unregulated markets. Taleb is talking about computability of prediction, a problem experienced in any sufficiently complex field, from quantum mechanics to financial markets to human creativity.

Max Born (besides being grandfather to Olivia Newton-John) was awarded the Nobel Prize in Physics in 1954 for his earlier statistical interpretation of the wavefunction. He once said ('Concluding remarks' to *Natural Philosophy of Cause and Chance* [7, p.209]):

> There are two objectionable types of believers: those who believe the incredible and those who believe that 'belief' must be discarded and replaced by 'the scientific method.'

Born is concerned with *descriptions* of the universe. Like all scientists, he is aiming at predictive content to his predictions. But he has a grasp of the fact (familiar in a formal sense to the logician) that computability has an uneasy relationship with definability, both in mathematics, and in the real world. And that just as in the post-normal science of Funtowicz and Ravetz, there is a very real value, even necessity, of descriptions based on incomplete information.

What is quite staggering in its enormity is the lack of *explicit* attention given to definability as a physical determinant. We are all familiar with the usefulness of a view of the universe as information, with natural laws computationally modelled. We have no problem with extracting from computational descriptions of real phenomena a latter-day expression of Leibniz's *Principle of Sufficient Reason*. We are used to the epistemological role of descriptions governing the world we live in. What is not admitted is a physical existence to higher-order descriptions – or, for that matter, of higher-order notions of computability, such as is convincingly described in Stephen Kleene's late sequence of papers (see [24,25]) on *Recursive Functionals and Quantifiers of Finite Types*.

This discussion of prediction versus description provides the background to two important and closely related questions:

I) How do scientists *represent and establish* control over information about the Universe.
II) How does the Universe itself *exercise* control over its own development ... or, more feasibly:
IIa) How can we *reflect that control* via our scientific and mathematical representations.

Of course, science very effectively comes up with specific theories for particular aspects of the universe, theories which say "it's like this ... this is how *we* do it, and this is how the *universe* does it". But when the narrative falters, the ad hoc approach allows—actually demands—that one resorts to guess-work. Examples of this in physics are not rare. There is plenty of scope for some more basic mathematical value added. As David Gross was quoted as saying (*New Scientist*, Dec. 10 2005, *Nobel Laureate Admits String Theory Is In Trouble*):

> The state of physics today is like it was when we were mystified by radioactivity ... They were missing something absolutely fundamental. We are missing perhaps something as profound as they were back then.

Rising to the challenge, we start by reducing the gap between computation and description.

2 The Power of a Paradigm

There are conditions under which the validity of definability as a determinant of physical reality is recognised, and is actually part of everyday mathematical practice. If we go back to the time of Isaac Newton, we have the observed reality of the orbits of the planets, though in need of description and of clarification, and of *prediction*. The inverse square law as it applied to gravitation was already conjectured by Robert Hooke and others by 1679; but it was the moving planets that were the reality, the law that was the conjecture, and the connection between the two that had as yet no description.

It was the description that Newton's development of the calculus provided, giving the essential link between law and observed data. And, as any student of pure mathematics learns, the description does involve those annoying quantifiers, hidden away in the limits underling the differential calculus, and made explicit again by the analyst. And the magic worked by the 'last sorcerer' Isaac Newton (see Michael White [42]) was accepted because it was *the result* which fitted with observation, and the *predictions* of which the outcomes were constituted. And it was the description of reality so attained which consolidated the inverse square law into a generally accepted natural law—one which would eventually even lose its suspect 'action at a distance' nature with the much later development of sub-atomic particle physics.

So was founded a powerful paradigm under which the determining role of definability would function unrecognised for hundreds of years, familiar in its usefulness, strange in its unrealised ramifications. Differential equations became the common currency of science, as theory ran on rails to descriptions of reality, connecting foundational conjectures to higher-order descriptions of what we see.

Of course, differential equations do not always present us with nicely formulated solutions, enabling computable predictions and unique descriptions. This presents no foundational uneasiness for the working scientist. We know what sort of reality we are looking for. We can extract numerical approximations to solutions from the more complex descriptions given in terms of, say, non-linear differential equations. And if there are multiple solutions—well, we know reality is uniquely determined—we just need the right boundary conditions to get everything in order. Normal science works, crank the handle, make definability work for us. Restrict one's research to problems related to the technicalities contained within the trusted paradigm.

One could always rely on Alan Turing to stretch a paradigm—if not to breaking point—at least to the point where it connects up with something else, something which makes us aware of hidden potentialities underlying our everyday assumptions. Who would have thought that differential equations would tell us about the incidence of Fibonacci sequences in nature, in the structure of sunflower heads; about the shapes of patterns on a cow; or predict moving stripes on a fish. What was special about Turing's seminal work [40] on morphogenesis was its *revelation* of mathematical patterns where not previously suspected. Revolutionised how we saw the well-established area of phylotaxis and played a

seminal role in later developmental biology. It was a visionary leap in the dark, a latter-day and unexpected application of Leibniz's 'principle of sufficient reason'.

From today's vantage point, the surprise of finding cow hide markings constrained according to a mathematically defined description is not so great. Turning the pages of Hans Meinhardt's beautifully illustrated book [29] *The Algorithmic Beauty of Sea Shells*, based on Alan Turing's work, the mathematical nature of the patterns seems to leap out at us. But the one-time adventure of discovery is clear from Bernard Richards' later account [33] of his work on Radiolaria structure as an MSc student with Turing in the early 1950s:

> When I had solved the algebraic equations, I then used the computer to plot the shape of the resulting organisms. Turing told me that there were real organisms corresponding to what I had produced. He said that they were described and depicted in the records of the voyages of HMS Challenger in the 19th Century.
>
> I solved the equations and produced a set of solutions which corresponded to the actual species of Radiolaria discovered by HMS Challenger in the 19th century. That expedition to the Pacific Ocean found eight variations in the growth patterns. ...
>
> My work seemed to vindicate Turing's Theory of Morphogenesis. These results were obtained in late June of 1954: alas Alan Turing died on 7th June that year (still in his prime- aged 41), just a few days short of his birthday on the 23rd. Sadly, although he knew of the Radiolaria drawings from the Challenger voyage, he never saw the full outcome of my work nor indeed the accurate match between the computer results ... and Radiolaria.

Of course, Turing's differential equations, connecting simple reaction-diffusion systems to morphogenesis, were not very complicated. But they pointed the way to the application of mathematical descriptions in other more complicated contexts. And, to the *existence* of descriptions associated with natural phenomena which might be hard to capture, but which themselves were subject to a level of mathematical constraint only explored by the logician and the metamathematician.

Stretch the paradigm too far, though, and it loses its power. It becomes unrecognisable, and no longer applies. In any case, we have not travelled far beyond Question I) of Section 1. Newton told us to seek mathematical definitions of what we observed to be real. To that extent one does not fully embrace definability as more than an epistemological tool.

3 Definability and the Collapse of the Wave Function

Familiar to us now is the 20th century building of a remarkably successful description of the observed ambiguities of quantum phenomena—with an associated probabilistic analysis enabling the retention of a reassuring level of determinism. Also familiar is the filling of deterministic shortcomings with

a (mathematically naive) randomness underlying the predicted probabilities, which Calude and his collaborators have examined more closely in various writings (see, e.g. [10,11]) over the years. For those working with clarified notions, randomness loses its fundamentality and scientific dignity when calibrated, and made to rub shoulders with mere incomputability.

From a practical point of view, quantum mechanics is an astonishing success. The uncertain attributes of an individual particle is not important in our world. But from an interpretive point of view, it is something of a car crash. Let us look at some familiar aspects, and describe how the paradigm established by Isaac Newton and stretched by Turing can be extended further to provide an appropriate home for the apparent weirdness of the quantum world.

What is problematic is what happens in the collapse of the wave function—concerning the reasons for which, according to Richard Feynman, "we have no idea" (Feynman, Leighton and Sands [21]). The fact that the collapse appears to bring with it nonlocal communication of a seemingly causal nature, indicating a need for some conceptually new analysis of physical causality. The discovery of the reality of the nonlocality originated with the puzzle presented by the EPR thought experiment of Albert Einstein, Boris Podolsky and Nathan Rosen [20]. By 1964, Bell's [3] inequality had provided an empirically testable version, later confirmed via real experiment by Aspect and his collaborators [1,2]. Even though QED provides a mathematical formulation which seems to successfully transcend some of the conceptual and descriptive difficulties inherited from the classical context (e.g., wave/particle ambiguity), it does not remove the essential dichotomy between theoretical descriptions at the quantum and classical levels or explain the unpredictability and inherent inconsistencies associated with the transition between the two.

The details of the original thought experiment, its recasting by Bell [3], and its subsequent empirical scrutiny, can be found in various places—see, for example, Omnès [30, chap.9]. Einstein and his two colleagues considered (in the more famous of their examples) the behaviour of two particles whose initial interaction means that their subsequent descriptions are derived from a single Schrödinger wave equation, although subsequently physical communication between the two, according to special relativity, may involve a significant delay. If one imagines the simultaneous measurement of the momentum of particle one and of the position of particle two, one can arrive at a complete description of the system. But according to the uncertainty principle, one cannot simultaneously quantify the position and momentum observables of a particle. The standard Copenhagen interpretation of quantum theory requires that the measurement relative to particle one should instantaneously make the measurement relative to particle two ill-defined—described in terms of a collapse of the wave function.

One could try to explain this in various ways. Clearly it constituted a major challenge to any existing causal explanation, for various reasons. Kreisel [26, p.177], reminds us, in a mathematical context, that the appearance of a process is not necessarily a good guide to the computability or otherwise of its results. The immediate question was whether there existed undiscovered, but qualitatively

familiar, causal aspects of the material universe in terms of which this apparent inconsistency could be explained. EPR commented:

> While we have thus shown that the wave function does not provide a complete description of the physical reality, we left open the question of whether or not such a description exists. We believe, however, that such a theory is possible.

Despite Einstein's later negative comments concerning the particular proposal of Bohm [5], this has been taken (see Bell [4]) as a tacit endorsement of a 'hidden variables' approach. And the experimental violation of Bell's inequality as a rejection of the possibility of a hidden variables solution.

But the problem goes beyond the discussion of hidden variables. There are aspects of the nonlocal 'causal' connection revealed during the collapse of the wave function that are not at all like analogous 'nonlocality' in the classical context. As Maudlin [28, pp.22–23] notes, unlike in the familiar classical context, the power of the causal connection is not attenuated by increasing separation of the particles; whilst the connection is peculiar to the two particles involved, drawing no others into the causal dynamics; and (most importantly) the timing of the connection entails a connection travelling faster than light. In fact, as Maudlin (p.141) puts it:

> Reliable violations of Bell's inequality need not allow superluminal signalling but they do require superluminal causation. One way or another the occurrence of some events depends on the occurrence of space-like separated events, and the dependence cannot be understood as a result of the operation of non-superluminal common causes.

And (p.144): 'We must begin to speculate about exactly how the superluminal connection might be made.' Otherwise, we are threatened with an inconsistency between quantum theory and relativity, and a quagmire of metaphysical debate about tensing and foliations of space-time (see *The Oxford Handbook of Philosophy of Time* [8]). All this points to the need for a comprehensive interpretation of the apparent 'collapse' of the wave function. Existing ones from within physics tend to fall into those which deny the physical reality of the collapse in terms of hidden variables or decoherent realities, and those which attempt a 'realistic' collapse interpretation. The confusion of proposals emerging present all the symptoms of a classic Kuhnian paradigm-shift in progress.

One is left with the unanswered question of what is the appropriate mathematical framework, radical enough to clarify questions raised about the logical status of contemporary physics—in particular, its *completeness* (in regard to explanations of nonlocality and the measurement problem), and *consistency* (in particular concerning the contrasting descriptions of the classical and quantum worlds). Of course, a genuine Kuhnian shift of paradigm should deliver *more* than solutions to puzzles. What it should deliver is a compulsively persuasive world-view within which people may construct their own versions of the reality in question.

4 Definition as Embodied Computation

We have argued elsewhere (e.g., [15], [17]) that emergence in the real world is the avatar of mathematical definability. That if one extracts mathematical structure from the material universe, then the definable relations on that structure have a special status which is reflected in our observation of higher-order pattern-formation in the physical or mental context. As we have seen, there is no problem with this at the epistemological level—the language involved is a welcome facilitator there. But what does this have to deliver in relation to the collapse of the wave function and our question II) above? In what sense can we regard our relativistic quasi-classical world we live in as *emergent* from the quantum world underpinning it, and what does such a view deliver, in terms of what we know from physical examples of emergence and the mathematical characterisation in terms of definability?

There is of course a general acceptance of the practical and historical separation of different areas of scientific and social investigation. Despite the growing awareness of the non-exclusiveness of the boundaries between, say, quantum theory and biology (see, for example, Riepers, Anders and Vedral [34]), there is no doubt that biology has largely established an autonomous domain of objects of study, concepts, causal relationships, etc, which do not benefit in any practical way from reduction to the underlying quantum world, with its quite different scientific concerns. And this fragmentation of human epistemology is an everyday feature of modern life.

The claim is that this is emergence on a macro-scale, characterised by a hierarchical descriptive structure, which makes the reductive possibilities between levels of science not just inconvenient, but impossible in any computationally accessible sense. So we observe the quantum world 'computing' objects and relations at a higher level, but not within the standard framework of the computer science built on Turing's universal machines. We are now approaching the 'strong determinism' of Roger Penrose whose computational aspects were discussed in the 1995 paper of Calude et al [9]. And it is not just an epistemological divide: the science reflects a level of physical reality.

One is not accustomed to this hierarchical division between the quantum and the relativistic worlds—it is all physics. If one were to demand a consistent theory of biology and quantum mechanics, one would recognise the inappropriateness. It is not immediately obvious to the physicists that they are trying to do something which does not make sense.

In what sense can emergence be regarded as computation? Part of the answer depends on the way we regard *information*, the raw material of computational processes. An interesting pointer to the non-uniqueness of quantum phenomena, suggesting we treat them in a more general context, is Rijsbergen's analysis [41] of information retrieval in terms of quantum mechanics. Of course, the standard computational paradigm based on the universal Turing machine simplifies the physical context of a computation to the point where the only interest in the actual representation of the information underlying the computation is a semantic one. In emergence, there is a context composed of basic objects obeying basic

computational rules not obviously taking us beyond the standard computational model. But what one observes is high level of interactivity which gives the system a globally effective functionality whereby it 'computes' relations on itself. The 'information' becomes a possibly rich structure, presenting a computation whose character owes much to its *embodiment*. At the macro-level, it is this embodied computation one would look to to deliver the global relations on the universe which we observe as natural laws. In fact, if one has experience of mathematical structures and their definable relations, one might look for the definability of *everything* in the universe which we observe, from the atomic particles to the laws governing them.

As mentioned earlier, one can view this physical computability as a mathematically characterisable computability. One just needs to apply the familiar (to computability theorists) notions of higher-type computability, complete with much of the character of the type 1 theory. However, the higher type data being computed over cannot be handled in the same way as the type 0 or type 1 inputs (via approximations) can. This is where *embodiment* plays a key role. In the case of the human mind, the *brain* is actually part of the input. So however one captures the logical aspects of the computation, this leaves a key part of the computational apparatus necessarily embodied. In practice, this means there is no universal computing machine at this level. The functionalist view of computation realisable on different platforms loses its power in this context.

Before looking more closely at the mathematics, let us review *The Five Great Problems in Theoretical Physics* listed by Lee Smolin in his book [36] on *The Trouble With Physics*:

- *Problem 1: Combine relativity and quantum theory into a single theory that can claim to be the complete theory of nature.*
- *Problem 2: Resolve the problems in the foundations of quantum mechanics, either by making sense of the theory as it stands or by inventing a new theory that does make sense.*
- *Problem 3: Determine whether or not the various particles and forces can be unified in a theory that explains them all as manifestations of a single, fundamental entity.*
- *Problem 4: Explain how the values of the free constants in the standard model of particle physics are chosen in nature.*
- *Problem 5: Explain dark matter and dark energy. Or, if they don't exist, determine how and why gravity is modified on large scales. More generally, explain why the constants of the standard model of cosmology, including the dark energy, have the values they do.*

All of these questions, in a broad sense, can be framed as questions concerning definability. Unfortunately, like computation, definability needs an input. Leaving aside the most fundamental question of what underlies what we *can* talk about, we move on to examine what underlies much of what we *do* talk about. it is important to frame the mathematics in sufficiently general

information-theoretic terms to be applicable to a wide range of scientific contexts. We will return to Smolin's problems at the end.

5 Turing Invariance and the Structure of Physics

Turing's oracle machines were first described in his 1939 paper [39] (for details, see [14]). Intuitively, the computations performable are no different to those of a standard Turing machine, except that auxiliary information can be asked for, the queries modelled on those of everyday scientific practice. This is seen most clearly in today's digital data gathering, whereby one is limited to receiving data which can be expressed, and transmitted to others, as information essentially finite in form. But with the model comes the capacity to collate data in such a way as enable us to deal with arbitrarily close approximations to infinitary inputs and hence outputs, giving us an exact counterpart to the computing scientist working with real-world observations. If the different number inputs to the oracle machine result in 0-1 outputs from the corresponding Turing computations, one can collate the outputs to get a binary real computed from the oracle real, the latter now viewed as an input. This gives a partial computable functional Φ, say, from reals to reals. One can obtain a standard list of all such functionals.

Put $\mathbb{R}$ together with this list, and we get the Turing Universe. Emil Post [32] gathered together binary reals which are computationally indistinguishable from each other, in the sense that they are mutually Turing computable from each other. Mathematically, this delivered a more standard mathematical structure to investigate—the familiar upper semi-lattice of the *degrees of unsolvability*, or *Turing degrees*.

There are obvious parallels between the Turing universe and the material world. Most basic, science describes the world in terms of real numbers. This is not always immediately apparent. Nevertheless, scientific theories consist, in their essentials, of postulated relations upon reals. These reals are abstractions, and do not come necessarily with any recognisable metric. They are used because they are the most advanced presentational device we can practically work with, although there is no faith that reality itself consists of information presented in terms of reals.

Some scientists would take us in the other direction, and claim that the universe is actually finite, or at least countably discrete. We have argued elsewhere (see for example [18]) that to most of us a universe without algorithmic content is inconceivable. And that once one has swallowed that bitter pill, infinitary objects are not just a mathematical convenience (or inconvenience, depending on ones viewpoint), but become part of the mathematical mould on which the world depends for its shape. As it is, we well know how essential algorithmic content is to our understanding of the world. The universe comes with recipes for doing things. It is these recipes which generate the rich information content we observe, and it is reals which are the most capacious receptacles we can humanly carry our information in, and practically unpack.

Globally, there are still many questions concerning the extent to which one can extend the scientific perspective to a comprehensive presentation of the universe in terms of reals—the latter being just what we need to do in order to model the immanent emergence of constants and natural laws from an entire universe. Of course, there are many examples of presentations entailed by scientific models of particular aspects of the real world. But given the fragmentation of science, it is fairly clear that less natural presentations may well have an explanatory role, despite their lack of a role in practical computation.

The natural laws we observe are largely based on algorithmic relations between reals. Newtonian laws of motion will computably predict, under reasonable assumptions, the state of two particles moving under gravity over different moments in time. And, as previously noted, the character of the computation involved can be represented as a Turing functional over the reals representing different time-related two-particle states. One can point to physical transitions which are not obviously algorithmic, but these will usually be composite processes, in which the underlying physical principles are understood, but the mathematics of their workings outstrip available analytical techniques.

What is important about the Turing universe is that it has a rich structure of an apparent complexity to parallel that of the real universe. At one time it was conjectured that the definable relations on it had a mathematically simple characterisation closely related to second-order arithmetic. Nowadays, it appears much more likely that the Turing universe supports an interesting automorphism group (see [13]), echoing the fundamental symmetries emerging from theoretical descriptions of the physical universe. On the other hand, there are rigid substructures of the Turing universe (see [35]) reminiscent of the classical reality apparent in everyday life. And the definable relations hosted by substructures provide the basis for a hierarchical development.

Anyway, it is time to return to Lee Smolin's 'Great Problems'.

The first question asked for a combining of relativity and quantum theory into a single theory that can claim to be the complete theory of nature. Smolin comments [36, p.5] that "This is called the *problem of quantum gravity.*"

Given that quantum gravity has turned out to be such a difficult problem, with most proposals for a solution having a Frankensteinian aspect, the suspicion is that the quest is as hopeless as that for the Holy Grail. And that a suitable hierarchical model of the fragmentation of the scientific enterprise may just be what is needed to give the picture we already have some philosophical respectability. What is encouraging about the Turing model is that it currently supports the dichotomy between a 'low level' or 'local' structure with a much sparser level of uniquely definable relations than one encounters at higher levels of the structure. Of course, the reason for this situation is that the higher one ascends the structure from computationally simpler to more informative information, the more data one possesses with which to describe structure.

We are happy to accept that despite the causal connections between particle physics and the study of living organisms, the corresponding disciplines are based on quite different basic entities and natural laws, and there is no feasible and

informative reduction of the higher level to the more basic one. The entities in one field may emerge through phase transitions characterised in terms of definable relations in the other, along with their distinct causal structures. In this context, it may be that the answer to Smolin's first Great Problem consists of an explanation of why there is no single theory (of the kind that makes useful predictions) combining general relativity and quantum theory.

It is worth noting that even the question of consistency of quantum theory with special relativity and interaction presents serious problems. The mathematician Arthur Jaffe, with J. Glimm and other collaborators, succeeded in solving this problem in space-time of less than four dimensions in a series of papers (see [23]), but it is still open for higher dimensions.

The second question asks us to 'resolve the problems in the foundations of quantum mechanics, either by making sense of the theory as it stands or by inventing a new theory that does make sense'.

This is all about 'realism', its scope, how to establish it. Of course, the absence of a coherent explanation of what we observe makes it difficult to pin down what are the obstacle. But it is in the nature of the establishment of a new paradigm that it does not just tidy up some things we were worried about, but gives us a picture that tells us more than we even knew we were lacking. A feel for mathematical definability; a plausible fundamental structure which supports the science, and is known to embody great complexity which tests researchers beyond human limit; and the demystifying of observed emergence via the acceptance of the intimate relationship between definition and ontology. That has the power to do it in this case. And it all becomes clear. The material universe has to do everything for itself. It has nothing it does not define. Or more to the point, its character is established by the nature of its automorphisms. Applying the sense underlying Leibniz's principle of sufficient reason, objects and interactions which exist materialise according to what the mathematics permits—that is, what the structure itself pins down. If the whole structure can be mapped onto itself with a particle in two different locations in space-time , then the particle *exists* in two different locations. If entanglement changes the role of an aspect of the universe in relation to the automorphisms permitted, so does the ontology and the associated epistemological status. Observation can do this to one of our particles. Of course, the lack of a comprehensive description of the particle will have to be shared by whatever form the particle takes. There is nothing odd about the two slit experiment. Or about the dual existence of a photon as a wave or as a particle. And the interference pattern is just evidence that a particle as two entities can define more than it can solo. And as for decoherence, many-worlds and the multiverse? It is not just that the whole scenario becomes redundant, it is that the acceptance of such multifarious permutations of reality is mathematically naive. A small change in a complex structure with a high degree of interactivity can have a massive global effect. And the mathematics does have to encompass such modifications in reality.

So—there is a qualitatively different apparent breakdown in computability of natural laws at the quantum level—the *measurement problem* challenges us

to explain how certain quantum mechanical probabilities are converted into a well-defined outcome following a measurement. In the absence of a plausible explanation, one is denied a computable prediction. The physical significance of the Turing model depends upon its capacity for explaining what is happening here. If the phenomenon is not composite, it does need to be related in a clear way to a Turing universe designed to model computable causal structure. We look more closely at definability and invariance.

Let us first look at the relationship between automorphisms and many-worlds. When one says "I tossed a coin and it came down heads, maybe that means there is a parallel universe where I tossed the coin and it came down tails", one is actually predicating a large degree of correspondence between the two parallel universes. The assumption that *you* exist in the two universes puts a huge degree of constraint on the possible differences—but nevertheless, some relatively minor aspect of our universe has been rearranged in the parallel one. There are then different ways of relating this to the mathematical concept of an automorphism.

One could say that the two parallel worlds are actually isomorphic, but that the structure was not able to *define* the outcome of the coin toss. So it and its consequences appear differently in the two worlds. Or one could say that what has happened is that the worlds are *not* isomorphic, that actually we were able to change quite a lot, without the parallel universe looking very different, and that it was these fundamental but hidden differences which forces the worlds to be separate and not superimposed, quantum fashion. The second view is more consistent with the view of quantum ambiguity displaying a failure of definability. The suggestion here being that the observed existence of a particle (or cat!) in two different states at the same time merely exhibits an automorphism of our universe under which the classical level is rigid (just as the Turing universe displays rigidity above $\mathbf{0}''$) but under which the sparseness of defining structure at the more basic quantum level enables the automorphism to re-represent our universe, with everything at our level intact, but with the particle in simultaneously different states down at the quantum level. And since our classical world has no need to decohere these different possibilities into parallel universes, we live in a world with the automorphic versions superimposed. But when we make an observation, we establish a link between the undefined state of the particle and the classical level of reality, which destroys the relevance of the automorphism.

To believe that we now get parallel universes in which the alternative states are preserved, one now needs to decide how much else one is going to change about our universe to enable the state of the particle destroyed as a possibility to survive in the parallel universe—and what weird and wonderful things one must accommodate in order to make that feasible. It is hard at this point to discard the benefits brought by a little mathematical sophistication. Quantum ambiguity as a failure of definability is a far more palatable alternative than the invention of new worlds of which we have no evidence or scientific understanding.

Let's take question 4 next: Explain how the values of the free constants in the standard model of particle physics are chosen in nature.

Things are starting to run on rails. Conceptually, the drawing together of a global picture of our universe with a basic mathematical model is the correspondence between emergent phenomena and definable relations. This gives us a framework within which to explain the particular forms of the physical constants and natural laws familiar to us from the standard model science currently provides. It goes some way towards substantiating Penrose's [31, pp.106-107] 'strong determinism', according to which "all the complication, variety and apparent randomness that we see all about us, as well as the precise physical laws, are all exact and unambiguous consequences of one single coherent mathematical structure". Of course, this is all schematic in the extreme and takes little account of the hugely sophisticated and technically intriguing theory that people who actually get to grips with the physical reality have built up. These developers of the adventurous science, building on the work of those pioneers from the early part of the last century, are the people who will take our understanding of our universe to new levels. The descriptions posited by the basic mathematics need explicit expressions. While the basic theory itself is still at an exploratory stage. The automorphism group of the Turing universe is far from being characterised.

The third question is more one for the physicists: Determine whether or not the various particles and forces can be unified in a theory that explains them all as manifestations of a single, fundamental entity.

The exact relationships between the particles and forces is clearly a very complex problem. One does expect gravity to be describable in terms of more basic entities. One would hope that a more refined computability theoretic modelling of the physics might reveal some specific structure pointing to an eventual solution to this deep and intractable problem.

And as for question 5: Explain dark matter and dark energy. Or, if they don't exist, determine how and why gravity is modified on large scales. More generally, explain why the constants of the standard model of cosmology, including the dark energy, have the values they do.

Various possibilities come to mind. It may be that a more intimate relationship with the definability, engaging with the organic development of *everything*, may change the way the current physics fits together, removing the need for dark matter or energy. Or it may be that the darkness of these entities is explained by the relationship between levels and their relative definability. There are cosmological ramifications too, which might be discussed elsewhere.

For further discussion of such issues, see [12], [15], [16], [17] and [18].

A final comment: Definability/ emergence entailing globality and quantification does involve incomputability. And this may give an impression of randomness, without there actuality being any mathematical randomness embodied. This is very much in line with the research of Cris Calude and Karl Svozil on the nature of quantum randomness [11].

References

1. Aspect, A., Dalibard, J., Roger, G.: Experimental test of Bell's inequalities using time-varying analyzers. Phys. Rev. Letters 49, 1804–1807 (1982)

2. Aspect, A., Grangier, P., Roger, G.: Experimental realization of Einstein-Podolsky-Rosen-Bohm gedanken experiment; a new violation of Bell's inequalities. Phys. Rev. Letters 49, 91 (1982)
3. Bell, J.: On the Einstein Podolsky Rosen paradox. Physics 1, 195 (1964)
4. Bell, J.S.: Einstein-Podolsky-Rosen experiments. In: Proceedings of the Symposium on Frontier Problems in High Energy Physics, Pisa, pp. 33–45 (June 1976)
5. Bohm, D.: A suggested interpretation of the quantum theory in terms of 'hidden' variables, I and II. Phys. Rev. 85, 166–193 (1952); reprinted in Wheeler, J.A., Zurek, W.H. (eds.) Quantum Theory and Measurement. Princeton University Press, Princeton (1983)
6. Born, M.: The Restless Universe. Blackie & Son, London (1935)
7. Born, M.: Natural Philosophy of Cause and Chance, Clarendon Press (1949)
8. Callender, C. (ed.): The Oxford Handbook of Philosophy of Time. Oxford University Press, Oxford (2011)
9. Calude, C., Campbell, D.I., Svozil, K., Stefanescu, D.: Strong determinism vs. computability. In: DePauli-Schimanovich, W., Köhler, E., Stadler, F. (eds.) The Foundational Debate: Complexity and Constructivity in Mathematics and Physics, pp. 115–131. Kluwer, Dordrecht (1995)
10. Calude, C.: Algorithmic Randomness, Quantum Physics, and Incompleteness. In: Margenstern, M. (ed.) MCU 2004. LNCS, vol. 3354, pp. 1–17. Springer, Heidelberg (2005)
11. Calude, C.S., Svozil, K.: Quantum randomness and value indefiniteness. Advanced Science Letters 1, 165–168 (2008)
12. Cooper, S.B.: Clockwork or Turing U/universe? - Remarks on causal determinism and computability. In: Cooper, S.B., Truss, J.K. (eds.) Models and Computability. London Mathematical Society Lecture Notes Series, vol. 259, pp. 63–116. Cambridge University Press, Cambridge (1999)
13. Cooper, S.B.: Upper cones as automorphism bases. Siberian Advances in Math. 9, 1–61 (1999)
14. Cooper, S.B.: Computability Theory. Chapman & Hall/CRC, Boca Raton, London, New York, Washington, D.C (2004)
15. Cooper, S.B.: Definability as hypercomputational effect. Applied Mathematics and Computation 178, 72–82 (2006)
16. Cooper, S.B.: How Can Nature Help Us Compute? In: Wiedermann, J., Tel, G., Pokorný, J., Bieliková, M., Štuller, J. (eds.) SOFSEM 2006. LNCS, vol. 3831, pp. 1–13. Springer, Heidelberg (2006)
17. Cooper, S.B.: Computability and emergence. In: Gabbay, D.M., Goncharov, S.S., Zakharyaschev, M. (eds.) Mathematical Problems from Applied Logic I. Logics for the XXIst Century. Springer International Mathematical Series, vol. 4, pp. 193–231 (2006)
18. Cooper, S.B., Odifreddi, P.: Incomputability in Nature. In: Cooper, S.B., Goncharov, S.S. (eds.) Computability and Models, pp. 137–160. Kluwer Academic/Plenum, New York, Boston, Dordrecht, London, Moscow (2003)
19. Einstein, A.: Autobiographical Notes. In: Schilpp, P. (ed.) Albert Einstein: Philosopher-Scientist. Open Court Publishing (1969)
20. Einstein, A., Podolsky, B., Rosen, N.: Can quantum mechanical description of physical reality be considered complete? Phys. Rev. 47, 777–780 (1935)
21. Feynman, R.P., Leighton, R.B., Sands, M.: The Feynman Lectures on Physics, vol. 3. Addison-Wesley (1965)

22. Funtowicz, S.O., Ravetz, J.R.: A New Scientific Methodology for Global Environmental Issues. In: Costanza, R. (ed.) Ecological Economics: The Science and Management of Sustainability, pp. 137–152. Columbia University Press, New York (1991)
23. Jaffe, A.: Quantum Theory and Relativity. In: Doran, R.S., Moore, C.C., Zimmer, R.J. (eds.) Contemporary Mathematics Group Representations, Ergodic Theory, and Mathematical Physics: A Tribute to George W. Mackey, vol. 449, pp. 209–246 (2008)
24. Kleene, S.C.: Recursive functionals and quantifiers of finite types I. Trans. of the Amer. Math. Soc. 91, 1–52 (1959)
25. Kleene, S.C.: Recursive Functionals and Quantifiers of Finite Types II. Trans. of the Amer. Math. Soc. 108, 106–142 (1963)
26. Kreisel, G.: Some reasons for generalizing recursion theory. In: Gandy, R.O., Yates, C.E.M. (eds.) Logic Colloquium, vol. 69, pp. 139–198. North-Holland, Amsterdam (1971)
27. Kuhn, T.S.: The Structure of Scientific Revolutions, 3rd edn. University of Chicago Press, Chicago (1996)
28. Maudlin, T.: Quantum Non-Locality & Relativity: Metaphysical Intimations of Modern Physics, 3rd edn. Wiley-Blackwell, Malden (2011)
29. Meinhardt, H.: The Algorithmic Beauty of Sea Shells, 4th edn. Springer, Heidelberg (2009)
30. Omnès, R.: The Interpretation of Quantum Mechanics. Princeton University Press, Princeton (1994)
31. Penrose, R.: Quantum physics and conscious thought. In: Hiley, B.J., Peat, F.D. (eds.) Quantum Implications: Essays in honour of David Bohm, pp. 105–120. Routledge & Kegan Paul, London, New York
32. Post, E.L.: Degrees of recursive unsolvability: preliminary report (abstract). Bull. Amer. Math. Soc. 54, 641–642 (1948)
33. Richards, B.: Turing, Richards and Morphogenesis. The Rutherford Journal 1 (2005), http://www.rutherfordjournal.org/article010109.html
34. Rieper, E., Anders, J., Vedral, V.: Entanglement at the quantum phase transition in a harmonic lattice. New J. Phys. 12, 025017 (2010)
35. Slaman, T.A.: Degree structures. In: Proceedings of the International Congress of Mathematicians, Kyoto, pp. 303–316 (1990/1991)
36. Smolin, L.: The Trouble With Physics: The Rise of String Theory, the Fall of Science and What Comes Next. Allen Lane/Houghton Mifflin, London, New York (2006)
37. Taleb, N.N.: The Black Swan. Allen Lane, London (2007)
38. Turing, A.: On computable numbers, with an application to the Entscheidungsproblem. Proceedings of the London Mathematical Society 42, 230–265 (1936/37); reprinted in A.M. Turing, Collected Works: Mathematical Logic, pp. 18–53
39. Turing, A.: Systems of logic based on ordinals. Proceedings of the London Mathematical Society 45, 161–228 (1939); reprinted in A.M. Turing, Collected Works: Mathematical Logic, pp. 81–148
40. Turing, A.M.: The Chemical Basis of Morphogenesis. Phil. Trans. of the Royal Society of London. Series B 237, 37–72 (1952)
41. van Rijsbergen, K.: The Geometry of Information Retrieval. Cambridge University Press, Cambridge (2004)
42. White, M.: Isaac Newton – The Last Sorcerer. Fourth Estate, London (1997)

Exponential Decay in Quantum Mechanics

V. Kruglov[1], K.A. Makarov[2], B. Pavlov[3,4], and A. Yafyasov[4]

[1] Department of Physics, University of Auckland, New Zealand
[2] Department of Mathematics, University of Missouri, Columbia, USA
[3] The New Zealand Institute for Advanced Study,
Massey University (Albany Campus), Auckland, New Zealand
[4] V.A. Fock Institute for Physics, Department of Physics,
St. Petersburg University, Russia

First of all: disregard all of so called "preliminary physical expectations", because all these expectations are just a pseudonym for prejudices elaborated by the older generation. These words belong to Dirac, not to me. The right way of creating new physics is different: one should begin with a beautiful mathematical idea. But it should be really beautiful! No special relations to physics is compulsory. But if it is really beautiful, it will certainly match useful physical applications, though it is not predefined, what sort of applications and where: it depends on physical consequences which may be extracted from the mathematical scheme.

V. Arnold, in the TV interview with S. Kapitsa. Abridged text of the interview is published as "*Why do we study mathematics*" in " Kvant", **1** (1993) (in Russian).

We do not have any theorems here - this research is not dead yet.

Answer given by Ildar Hamitov, a student in L. Faddeev's group (1980), on the request of Professor V. Buldyrev to present the main result of his master thesis in a formal way.

1 Exponential Decay: Physical Needs versus Mathematical Beauty

V. Arnold, in his prominent interview (1993) with Sergey Kapitsa, commented on the controversial idea by Paul Adrien Moris Dirac (formulated, in particular in his lecture at Moscow University in 1955) that *Physical laws should have mathematical beauty* (see the above epigraph). Arnold's comment is even more straightforward than the original version by Dirac which was softer by nature. On top of Dirac's receipt about choosing a new step: "it's future development should affect something, which was out of any doubts before, something which could not be revealed by the axiomatic formulation", see [1], Arnold comment contains an inspiring advice on how and where to find the new physics. Of course, both statements by Dirac and Arnold are literally wrong. Both are about the final formulation of the theory, when "research is already dead", but not the first revolutionary movement in the new direction. Max Planck's formulation of the leading idea of quantum physics – on the discrete nature of the light radiation from the cavity – had no connection with any beautiful mathematics at that time (in the beginning of the 20^{th} century). The initial mathematical formulation of the essence of Quantum Physics appeared almost 30 years later due to John von Neumann [2]. Since

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 268–288, 2012.

that moment a lot of new important details were added to it, but, really, the subject noticeably drifted towards mathematics.

In this paper, oriented to a wide audience of theoretical physicists and researchers in natural sciences, we provide a review of a beautiful chapter of modern mathematics: the Harmonic Analysis of Operators in Hilbert Spaces, see [13], which arose in the first half of the previous century as a branch of Complex Analysis. However, it was never used by physicists as it deserves, according to our vision. Following the above receipt of Arnold, we choose the decay problem in Quantum Physics, formulated in th framework of Harmonic Analysis. We hope that our attempt may eventually bring this beautiful piece of mathematics into the arsenal of mathematical tools of natural sciences.

The classical question on the validity of Quantum Mechanics for the description of the decay of the wave-packet was rarely on the front line of research in Physics. In fact, for almost 90 years after the revolutionary paper by Gamow [3], it has been considered more as an annoying nuisance. Nevertheless, many great minds contributed with competing points of view on the subject. In 1930, Weisskopf and Wigner suggested a persuading concept (further referred to as the WW concept) of the exponential decay rate for a quantum system with discrete spectrum, see [4]. Their proposal was recognized by most experimentalists as a viable treatment of the subject. Unfortunately, 17 years later, Fock and Krylov spoiled the happy end, see [5], by showing "from the first principles", that exponential decay cannot be explained based on the discrete spectrum hypothesis, leaving only one way out: considering quantum systems with continuous spectrum. This new concept (further denoted as KF concept), proposed by Fock and Krylov, also sounds natural. Indeed, Fock was the first physicist to suggest, in his textbook on Quantum Mechanics [6], an accurate treatment of the continuous spectrum. This remained unchanged up to now in all modern texts on Mathematics and Mathematical Physics. Yet, the KF concept did not become a gravestone for the question on decay. A paper by L. Khalfin, [7], communicated by Fock to the Russian Academy Doklady, contained an accurate calculation, again "from the first principles", on the decay of wave-packets for the simplest quantum problems, in particular for the 1D Schrödinger equation with compactly supported real potential.

Represent the evolution of the wave-packet by the Riesz integral of the resolvent $R_\lambda \equiv (H - \lambda I)^{-1}$ of the corresponding Hamiltonian H,

$$e^{iHt} = -\frac{I}{2\pi i} \int_\Gamma R_\lambda \, e^{i\lambda t} \, d\lambda, \tag{1}$$

on the contour Γ enclosing the spectrum σ of H. Using analyticity of the integrand on the two-sheets Riemann surface of the spectral parameter, one can deform $\Gamma \to \Gamma'$ to reveal i) the unitary component of the evolution associated with eigenvalues λ_s of H, exponentially decreasing terms caused by resonances μ_s – the poles of the resolvent kernel- situated on the second spectral sheet and iii) a loop around the threshold $\lambda = 0$ of the continuous spectrum, the branching point of the resolvent kernel. Unfortunately, the corresponding contribution to evolution (1) of the latter is estimated by the power function Const $t^{-\beta}$ of the time t, with β depending on the incident data.

This result actually revealed the error by Gamov and could possibly resolve the problem of the decay, if the non-exponential component of the decay would be observed in an experiment. Surprisingly, that was not the case up to now. Nevertheless, in the

modern textbook on Quantum Mechanics like [8], the result of L. Khalfin is quoted as an ultimate truth on decay. Absence of an experimental confirmation can be blamed on the admissible precision of the experiment, rather than on a theoretical fault.

In this paper we revisit again the quoted proposals by [4,5], attempting to find a point of view that would permit for the ends to meet one another. Our program does not eliminate the naive theoretical analysis presented in [7], but reduces the discussion of its validity to the problem of the *choice of measurement tools* that deliver the data from the quantum system to the observer. In this paper, we consider the case when the role of the "delivering tool" is played by the electromagnetic field, or, generally, by another zero-mass field. In the 1D example considered in this paper, the role of the delivering tool is played by a massless field governed by the wave equation, a 1D analog of photons. We will postpone for an upcoming publication a more realistic choice of the delivering tool as a classical electro-magnetic field in R_3 which also satisfies all natural assumptions we are basing on now.

Some of the basic mathematical tools that we use to interpret the exponential decay are already prepared by mathematicians. A similar situation was observed in Quantum Mechanics, where an exact understanding of self-adjointness (the physicists required for the Schrödinger theory by mid 1920s) was already prepared by Hermann Weyl in 1916, see an adapted text in [9]. In our treatment of the exponential decay, we use the analysis of the acoustic scattering problem. Again, the mathematics was prepared by Peter Lax and Ralph Phillips in the 1960s of the last century (see [10]). Actually, Lax and Phillips succeeded to overcome, without even noticing it, the horror physicists survived when they discovered that the evolution of a quantum system with positive Hamiltonian L may be generated by another operator $\mathcal{L}$, which has both negative and positive branches of spectrum. This phenomenon, discovered by Dirac in the 1930s, was rigorously analyzed by Hegerfeld in his prominent theorem [11] only at the end of previous century. Hegerfeldt was able to show that the evolution of a quantum system with positive Hamiltonian always has "infinite tails". For instance, the component of the 1D acoustic evolution in the positive frequency sector is represented by D'Alembertian waves that admit analytic continuation into the upper half-plane, and thus cannot vanish on a set with positive measure on the real axis. This was an essential step to legitimizing the non-semi-bounded generators of evolution. Another example of a non-semi-bounded generator is given by the supercharge in super-symmetric quantum mechanics. So, by the end of previous century, the trick suggested by Lax and Phillips would not look surprising any more. But in the mid-century, it was probably still too special and suspicious for physicists: Lax and Phillips represented the evolution of the Cauchy data $\mathbf{u} \equiv \left(u, c^{-1}u_t\right)$ of the wave equation

$$c^{-2}\, u_{tt} - \Delta u = 0$$

as a unitary transformation in the energy-normed space of the Cauchy data

$$\| \mathbf{u} \|^2 = \frac{1}{2} \int_{\Omega_{out}} \left[c^{-2}|u_t|^2 + |\nabla u|^2\right] dx^3. \tag{2}$$

It was shown in [10] that the unitary evolution group $e^{i\mathcal{L}t} \equiv U_t \ : \mathbf{u}(0) \longrightarrow \mathbf{u}(t)$ is generated by a non-semi-bounded operator $\mathcal{L}$, an analog of the Dirac operator, that can be represents as an appropriate block operator matrix

$$\frac{1}{i}\frac{\partial \mathbf{u}}{c\partial t} = i\begin{pmatrix} 0 & -1 \\ -\Delta & 0 \end{pmatrix}\mathbf{u} \equiv \mathcal{L}\,\mathbf{u}, \quad \mathcal{L}^2 = -\Delta. \tag{3}$$

It turns out that (i) the generator $\mathcal{L}$ is self-adjoint in the energy-normed space $\mathcal{E}$ of all Cauchy data with finite energy, (ii) the spectrum of $\mathcal{L}$ in the energy-normed space of Cauchy data supported by the complement Ω_{out} of a compact domain $\Omega \subset R_3$ is absolutely continuous and it fills in the whole real axis (iii) the unitary group U_t has incoming and outgoing subspaces $\mathcal{D}_{in}$, $\mathcal{D}_{out}$ that are invariant with respect to the positive and negative semi-groups U_t and U_{-t}, $t \geq 0$, respectively. In fact, these subspaces consist of the data vanishing on the positive and negative light-cones respectively

$$U_t\,\mathcal{D}_{out} \subset \mathcal{D}_{out}, \quad t \geq 0, \qquad U_t\,\mathcal{D}_{in} \subset \mathcal{D}_{in}, \quad t \leq 0, \tag{4}$$

and Cauchy data of the incoming and outgoing waves on the complement $R_3 \backslash \Omega \equiv \Omega_{out}$ are mutually orthogonal with respect to the energy dot-product.

Fortunately, by that time the question on the description of invariant subspaces of an important isometry group in the Hilbert space was already solved by Beurling [12] *with no connection to the above acoustic problem*. Beurling considered in 1947 the shift operator T (right translation) in the Hilbert space l_2 of all complex square-summable sequences $\boldsymbol{x} = (x_0, x_1, x_1, x_2, x_3, \ldots)$

$$(x_0, x_1, x_1, x_2, x_3, \ldots) \stackrel{T}{\longrightarrow} (0, x_0, x_1, x_1, x_2, x_3, \ldots) \equiv t\boldsymbol{x}.$$

One of Beurling's problems in [12] was the description of all invariant subspaces $\mathcal{D}$ of $T : T\mathcal{D} \subset T$. Obviously, the space of all sequences, $\sum_s |x_s|^2$, with (several) zeros on the first positions, like $(0, x_1, x_1, x_2, x_3, \ldots)$, is invariant with respect to T.

What are the others invariant subspaces? It is not that easy to answer the question using the language of the l_2 space. But if we change the language by translating the question into the one of complex analysis and substituting sequences by functions analytic in the unit disc $\mathrm{B} = \{\zeta : |\zeta| < 1\}$:

$$\boldsymbol{X}(x_0, x_1, x_1, x_2, x_3, \ldots) \stackrel{T}{\longrightarrow} x_0 + \zeta x_1 + \zeta^2 x_2 + \zeta^3\, x_3 + \ldots \equiv x(\zeta) \equiv T\boldsymbol{x},$$

the problem of the description of invariant subspaces becomes almost trivial, *and this is the beauty*. In this way we get a marvellous chance to view the problem from a completely new point, substituting T by the multiplication operator: $T\boldsymbol{x} \longrightarrow \zeta x(\zeta)$. Indeed, this transformation is a unitary mapping of l_2 onto the class of all analytic functions on the unit disc, with square integrable boundary data on the circle $\Gamma = \{\zeta : |\zeta| = 1\}$. This is the celebrated Hardy class H^2_+: a subspace of $L_2(\Gamma)$ consisting of all functions which admit an analytic continuation onto the unit disc equipped with the norm

$$\frac{1}{2\pi}\int_\Gamma |x(e^{i\theta})|^2 d\theta = |\boldsymbol{x}|^2_{l_2}.$$

The subspace of all sequences $(0, x_1, x_1, x_2, x_3, \ldots)$, with zero on the first position, is transformed into the class ζH^2_+ of all analytic functions in the unit disc vanishing at the center of the disc. It is clear now that all subspaces of functions vanishing at an inner point a are invariant with respect to T and are represented as $\frac{a-\zeta}{1-\bar{a}\zeta}H^2_+$.

Of course, all subspaces of the analytic functions in the unit disk generated by finite or infinite Blaschke products $\Pi_{\boldsymbol{a}}(\zeta) \equiv \Pi_s \frac{a_s-\zeta}{1-\bar{a}_s\zeta}\frac{a_s}{|a_s|}$, with convergent series $\sum_s(1-|a_s^2|) < \infty$, are invariant subspaces $\mathcal{D}_{out} = \Pi_{\boldsymbol{a}} H_+^2$ of the shift operator $T : T\Pi_{\boldsymbol{a}} H_+^2 = \zeta\Pi_{\boldsymbol{a}} H_+^2 \subset \Pi_{\boldsymbol{a}} H_+^2$.

Some uniform limits of the Blaschke products give rise to so called singular inner functions $\Theta_\mu(\zeta)$ on the unit disc. They are represented via positive singular measures μ supported by the unit circle as $\Theta_\mu(\zeta) = \exp\int_{|\eta|=1}\frac{\zeta+\eta}{\zeta-\eta}d\mu(\eta)$. The functions Θ_μ also produce invariant subspaces $\Theta_\mu H_+^2$ of the shift, [14,13].

The full answer to the question about the structure of the *outgoing* invariant subspaces of the shift, $\zeta\mathcal{D}_{out} \subset \mathcal{D}_{out} \subset H_+^2$, is given by the formula

$$\mathcal{D}_{out} = \Theta_\mu\ \Pi\ H_+^2.$$

Similarly, the problem of the description of the invariant subspaces of the left shift U_t, $t < 0$, in the space of all sequences $x = (\ldots, -3, -2, -1)$ can be considered in the Hardy class H_-^2 of analytic functions on the complement to the unit disc. These subspaces can be constructed from the singular inner factor and the Blaschke product Θ, Π based on the symmetry principle $\bar{\Pi}(\zeta) = \Pi(\bar{\zeta}^{-1})$ as:

$$\mathcal{D}_{in} = \bar{\Theta}_\mu\ \bar{\Pi}\ H_-^2.$$

It is a remarkable fact, that the positive semi-group $\{\zeta^l\}$, $l = 0, 1, 2, 3, \ldots$, of the unitary group ζ^l on $L_2(\Gamma)$, restricted to the *co-invariant subspace* $H_+^2 \ominus \mathcal{D}_{out} \equiv \mathcal{K} = H_+^2 \ominus \Pi H_+^2 \equiv K$ *proves to be a contracting semi-group*

$$P_{\mathcal{K}}\zeta^l\Big|_{\mathcal{K}} \equiv Z^l,\ l = 0, 1, 2, 3, \ldots\ ,$$

with the generator Z. Indeed, since $\zeta P_{H_+^2} \in P_{H_+^2} \perp K$, for $l = 2$, we have:

$$Z^2 = P_K\zeta^2 P_K = P_K\zeta[P_{H_+^2} + P_K + P_{H_-^2}]\zeta P_K$$

$$= P_K\zeta[P_{H_+^2} + P_K]\zeta P_K = P_K\zeta P_K\zeta P_K = Z^2.$$

Moreover, the eigenvalues of the generator Z coincide with the zeros a_s of the Blaschke product $\Pi_{\boldsymbol{a}}$ and the corresponding eigenfunctions are $\psi_s[\zeta] = \frac{\Pi_{\boldsymbol{a}}(\zeta)}{a_s-\zeta}$. In addition, the bi-orthogonal system of eigenvectors of the adjoint operator Z^+ is constituted by the reproducing kernels $\phi_s(\zeta) = \frac{1}{1-\bar{a}_s\zeta}$, so that the spectral decomposition of Z, with simple discrete spectrum, is given by the interpolation series

$$f = \sum_s \frac{\Pi_{\boldsymbol{a}}(\zeta)}{a_s-\zeta}\frac{f(a_s)}{\frac{d\Pi_{\boldsymbol{a}}}{d\zeta}(a_s)}, \quad f \in K.$$

Similar explicit formulae are also true for the continuous shift of the real axis $f(x) \to f(x-t) \equiv U_t f$. The role of the incoming and outgoing subspaces $\mathcal{D}_{in,out}$ for the continuous shift group in the spectral (Fourier) representation $U_t \equiv e^{ipt}$ is played by the

Hardy classes of square-integrable functions $H^2_{\pm} \subset L_2(R)$ that admit an analytical continuation to the upper and lower half-planes, respectively. In particular, the subspaces ΠH^2_+ generated by the Blaschke products in the upper half-plane are invariant with respect to the (continuous) shift in the Fourier representation.

In general, the invariant subspaces of the positive semi-group $U_t, t \geq 0$ are parameterized by the inner functions $\Theta\Pi$ in the upper half-plane as $\Theta\Pi H^2_+ \equiv \mathcal{D}_{out}$. For the negative semi-group, the corresponding representation is of the form $\bar{\Theta}\bar{\Pi} H^2_- = \mathcal{D}_{in}$. The restriction of the positive semi-group of the continuous shift onto the orthogonal complement of $L_2(R) \ominus [\mathcal{D}_{in} \oplus \mathcal{D}_{out}] \equiv \mathcal{K}$ of the "incoming" and "outgoing" subspaces $\mathcal{D}_{in,out}$ in $L_2(R) \equiv \mathcal{E}$, with K the corresponding co-invariant subspace, is a strongly-continuous *Lax-Phillips semi-group* $P_{\mathcal{K}} U_t \big|_{\mathcal{K}} =: e^{i\mathcal{B}t}$, $t > 0$, of contractions generated by a dissipative operator $\mathcal{B}$. The spectral properties of the generator $\mathcal{B}$ are completely determined by the scattering matrix $S \equiv \Theta\Pi$ associated with the unitary group U_t and the corresponding unperturbed group U^0_t which is a colligation of the components of the evolution on the reduced space $\mathcal{E}_0 =: \mathcal{D}_{in} \oplus \mathcal{D}_{out}$, see [19].

Again, similarly to the above discrete case, the spectral analysis of the Lax-Phillips semi-group can be done in an explicit form in terms of the corresponding inner function $\Theta\Pi$, the scattering matrix.

Here is another source of beauty: the duality between the geometrical problem on invariant subspaces and relevant spectral questions for contracting and dissipative operators and classical questions on interpolation and approximation from the theory of analytic functions. Unfortunately, the simple calculations above never appeared in elementary courses of complex analysis for physicists or engineers.

The question on *exponential decay* for the acoustic problem on the complement of the scatterer Ω in a large ball B_R served as a central motivation for [10]. This problem is reduced to the study of spectral properties of the generator B of the Lax-Phillips semi-group: if all eigenvalues of the generator B are situated strictly in the upper spectral half-plane $\Im\lambda > \beta > 0$, then the Lax-Phillips semi-group admits an exponential estimation

$$\| e^{i\mathcal{B}t}\mathbf{u}_0 \| \leq Ce^{-\beta' t} \| \mathbf{u}_0 \|, \quad t \geq 0,$$

for any $\beta' < \beta$, with an appropriate absolute constant C, depending on β'. Highly nontrivial analysis was developed to prove the bound $\Im\lambda > \beta > 0$, $\lambda \in \sigma_{\mathcal{B}}$, for compact obstacles Ω that satisfy the exterior cone condition.

Generally, the whole machinery, developed in [10] to reach the quoted exponential estimate for acoustic scattering, is based on harmonic analysis of matrix-valued analytic functions $u \in L_2(E)$. It was motivated by the problem of description of all invariant subspaces of the standard shift groups $u(p) \to e^{ipt}u(p) \equiv u(p,t)$ in the space $L_2(E)$ of vector-valued, square-integrable functions $u(p) \in E$ on the real axis $-\infty < p < \infty$.

In fact, the above evolution group U_t is unitarily equivalent to the shift group, and the incoming subspaces of the evolution group U_t are equivalent to subspaces of the Hardy class $H^2_-(E) \subset L_2(E)$ of all square integrable functions admitting an analytic continuation into the lower half-plane $\Im p < 0$, see [14]. The outgoing subspaces of the evolution group are unitarily equivalent either to the Hardy class $H^2_+(E) \subset L_2(E)$, or to subspaces ΘH^2_+ of the Hardy class defined by the *inner factors* Θ, which are unitary on the real axis and admit an analytic continuation into the upper half-plane $\Im p > 0$. In the case when Π is a Blaschke product

$$\Pi(p) = \prod_l \left[\frac{p - p_l}{p - \bar{p}_l} \theta_l P_l + P_l^{\perp} \right],$$

with appropriate phase factors θ_l and projections $P_l,\ P_l^{\perp} = I - P_l$, the quantities $\bar{p}_l$ coincide with the eigenvalues of the adjoint generator $\mathcal{B}^+$, and the eigenfunctions of the adjoint generator, in the "incoming" spectral representation of the original unitary group U_t in the energy-normed space $\mathcal{E}$, coincide with the reproducing kernels $\varphi_l = \frac{e_l}{p - \bar{p}_l}$. The bi-orthogonal system of eigenfunctions of the original operator $\mathcal{B}$ is formed as $\psi_l = \frac{\Theta(p)}{p - p_l} e_l^+$, with $e^+ \in \ker \Theta(p_l)$, see [13,15,16].

In the general case, these facts are derived from an extended theory of the "functional model" (see, for instance, [13,15,16]), which covers the Lax-Phillips generators with absolutely continuous spectrum. The modern theory of the functional model allows one to reduce all the questions of the spectral theory of the Lax-Phillips semi-group to the relevant questions of the theory of analytic functions and/or harmonic analysis.

The crucial role of the theory of analytic functions for the theory of nonselfadjoint operator was predicted by M. G. Krein in his talk at the Moscow International Congress of Mathematicians in 1966, (see, [17,18]). The problem on exponential decay should be connected, from the point of view of mathematicians, with the list of problems on spectral analysis of dissipative or contracting operators. In the simplest case of a one-dimensional acoustic problem that we discuss in Section 3, most of the above facts of spectral analysis of the Lax-Phillips semi-group are established via straightforward calculations.

It must be noted that the first attempt to bridge the general theory of nonself-adjoint (in particular, dissipative) operators with relevant physics was undertaken by Livshits [20]. He was motivated by the observation that the problem of analysis of nonself-adjoint details of complex physical systems appears each time we attempt to substitute a whole complex system by a simpler surrogate system with similar properties. In [20] Livshits suggested a simplified model of a waveguide attached to a resonator, produced by substitution of a nonself-adjoint detail of the original system by a "triangular model", which, at the time, was the only available general model of a dissipative operator. Based on Livshits' discovery, a new, more convenient "functional model" was suggested by B. Sz.-Nagy and C. Foiaş (see [13]). But the role of the scattering matrix as a basic parameter of the functional model was not yet recognized at that stage. Few years later, a seminal paper [19] provided an important connection between the Lax–Phillips scattering theory and the Sz.-Nagy–Foiaş functional model, see [10,13]. One of he most important achievements of the theory was to give the spectral meaning to resonances, which never happened in the pure quantum mechanical treatment of the problem of the exponential decay.

All these important events succeeded just inside mathematics. Physicists did not see, until now, any connection between an elegant analysis used by the community of analysts in their study of the acoustic problem or the corresponding abstract shift groups. One of the reasons for that is that the unitary group generated by the semi-bounded Schrödinger operator does not have orthogonal incoming and outgoing subspaces, as it follows from the Hegerfeldt Theorem [11].

Nevertheless, an elegant analysis provided by the Lax–Phillips approach served as a motivation for the further research in a close area followed by publishing

numerous physical papers. In particular, in [22,23], the standard Hilbert space L_2 of square-integrable functions was supplied with additional structures transforming it into a space similar to the one used in [10]. In [24], a model Hamiltonian is constructed and an artificial analytic scattering matrix is suggested. In the case studied by Horwitz and Piron, the most important property of the model system in the Lax-Phillips approach, the orthogonality of the incoming and outgoing subspaces, was just formally derived from the analyticity of the constructed model scattering matrix. In recent papers, Baumgärtel with coauthors attempted to match the quantum mechanical condition of positivity of the generator of the evolution with spectral interpretation of the resonances to give the spectral meaning to the corresponding "Gamov vectors" (see [25,26]). Unfortunately, in this way all essential advantages of the Nagy-Foiaş functional model, such as explicit expressions for the eigenvectors of the Lax-Phillips semigroup, the Gamov vectors, completeness of the corresponding bi-orthogonal system, and the relevant spectral decomposition, were lost because of the absence of natural, physically motivated, orthogonal pair of incoming and outgoing subspaces. Besides, no physical consequences were derived in [25,26] from the proposed matching of quantum mechanics with the corresponding analog of the Lax-Phillips theory. This most likely suggests that the scheme proposed in the papers is sentenced, according to the Arnold algorithm, to remain, for another period inside mathematics until all these details are completed.

Contrary to that, in our version of bridging standard quantum mechanics with the Lax-Phillips theory, instead of inventing an artificial construction added on top of the standard quantum space of all square-integrable functions to imitate the Lax-Phillips structure, we consider excitations of the zero mass field playing the role of a channel passing information to the outside observer on the inner quantum system. Although the evolution of the "inner" the quantum system, for a finite time, can be represented in the Schrödinger form as e^{iLt} with a positive Hamiltonian L, the study of its asymptotics as $t \to \infty$ requires a treatment based on the complete zero-mass field evolution. The substitution of the Lorenz invariant picture by the Schrödinger picture of evolution can only be done under the "positivity of mass" condition (see next section). It is not trivial to match this requirement with the zero-mass condition for the Lax-Phillips scheme.

Thus, the central question in our treatment becomes the matching of the Lax-Phillips scattering scheme with quantum mechanics with the positive Hamiltonian, that is the question of the physical realization. And again, the answer to this question is not general and does not look obvious.

2 Scattering of Photons by a Superconductor: An Interplay between the Schrödinger Equation and the Klein-Gordon-Fock Equation

Yet an interesting example of similar matching can be found in the scattering of photons by a superconductor. Indeed, due to the Meissner effect, magnetic field cannot penetrate the super-conducting medium. The theoretical treatment of the phenomenon by Ginzburg and Landau (see [27]) is based on acquiring a non-zero mass by photons in the process of spontaneous symmetry breaking, the loss of abelian gauge invariance

of the Lagrangian of the electromagnetic field in the superconductor. Hence, both contradictory requirements of zero-mass in the outer space, and the non-zero mass in the inner space are satisfied. Thus, we may hope to "put both ends together" in the problem.

Consider a compact domain in R_3 filled with a superconductor. The Lagrangian of the electro-magnetic field in the outer space is represented in terms of the field A, the electromagnetic potential, as

$$\frac{1}{4}\int_{\hat{\Omega}_s} F^+F, \text{ where } F = dA,$$

(see for instance [28]). Here dA is an exterior differential of the field A, and F^+F is a 3-form obtained as an exterior product of 2-form F and its (hermitian) complement. In the inner space, due to the interaction of the electromagnetic field with the boson field of Cooper pairs, the Lagrangian is modified, in the boundary area of the superconductor, by additional massive terms containing the product of the electromagnetic field and the field of Cooper pairs see [28]. The depth of penetration of the magnetic field into the superconductor is estimated by the size δ of the Cooper pair, which is normally relatively large, greater than 10^{-7} cm. If the energy of photons does not exceed the Bardeen-Cooper-Schrieffer gap (the BCS - gap), the field of Cooper pairs can be eliminated and the scattering of photons by the superconductor can be treated in the one-body photon's sector, similar to the scattering problem in the classical quantum mechanics. In the one-body photon's sector, the scattering problem in vacuum $\hat{\Omega}_s$ can be reduced to the wave equation (the Klein-Gordon-Fock equation with zero mass). Similarly, the problem in Ω_δ is also reduced to the Klein-Gordon-Fock equation with non-zero mass. The corresponding scattered waves satisfy smooth matching conditions on the common boundary of $\hat{\Omega}_s$ and Ω_δ.

If the domain Ω_s is filled with a superconductor, then the electromagnetic potential should vanish on the common boundary $\partial\Omega_s \cap \partial\Omega_\delta$. Thus, one can consider, as a representative model, the Klein-Gordon-Fock equation in $R_3 = \Omega_s \cup \Omega_\delta \cup \hat{\Omega}_s$ assuming that the compact domain $\Omega_s \cup \Omega_\delta$ is filled with the superconductor, and $\hat{\Omega}_s$ is the vacuum. The mass is zero on $\hat{\Omega}_s$, but is non-zero on the δ-thin shell Ω_δ, separating the inner and the outer spaces. While Ω_s is filled by the superconductor, the electromagnetic field does not penetrate Ω_s, so that we can apply a zero boundary condition on $\partial\Omega_s \cup \partial\Omega_\delta$.

Then the spectrum of the Klein-Gordon-Fock operator in Ω_δ is discrete, and the one on the complement $\hat{\Omega}_s$ is continuous. Hence, the scattering in the small energy region, for energy not exceeding the creation threshold of the Cooper pair, has a resonance character. The scattering matrix of the problem is unitary and analytic with respect to the energy on the complement of the discrete set of resonances.

For small values of the added energy $E' = E - mc^2,\ E' << mc^2$, the evolution on Ω_δ can be described in a Schrödinger form:

$$E = c\sqrt{m^2c^2 + p^2} \approx mc^2 + \frac{p^2}{2m}.$$

Indeed, considering on Ω_δ the Klein-Gordon-Fock equation with non-zero mass

$$\frac{\hbar^2}{c^2}\frac{\partial^2\psi}{\partial t^2} = \left[\hbar^2\Delta - m^2c^2\right]\psi,$$

permits to split off the fast oscillations by the unitary transformation $\psi = e^{-imc^2\hbar^{-1}t}\phi$:

$$\frac{\partial\psi}{\partial t} = \left[\frac{\partial\phi}{\partial t}e^{-imc^2\hbar^{-1}t} - imc^2\hbar^{-1}\phi\, e^{-imc^2\hbar^{-1}t}\right] \approx -\frac{imc^2}{\hbar}\,\phi\, e^{-imc^2\hbar^{-1}t},$$

$$\frac{\partial^2\psi}{\partial^2 t} \approx -\left[\frac{2imc^2}{\hbar}\frac{\partial\phi}{\partial t} + \frac{m^2c^4}{\hbar^2}\phi\right]\, e^{-imc^2\hbar^{-1}t}, \tag{5}$$

which yields, for small momenta, the Schrödinger equation for ϕ

$$i\hbar\frac{\partial\phi}{\partial t} + \frac{\hbar^2}{2m}\Delta\phi = 0. \tag{6}$$

A nice feature of this equation is the possibility to interpret $|\phi|^2$ as the probability density for the particle to bound at the location marked by space coordinates (x, t) of the wave function $\phi(x, t)$, when the total probability to find the particle in the space is conserved $\int |\phi(x,t)|^2 dx = const$. But its formal use in the large time scale would give a non-exponential decay of the wave packet of the magnetic field. Moreover, vice versa, a straightforward analysis based on the Lax-Phillips scattering arguments for the zero-mass field in $\hat{\Omega}_s$ and the non-zero mass in the Klein-Gordon-Fock equation on Ω_δ shows an exponential decay, and even reveals the spectral meaning of resonances.

Another interesting example of the exponential decay can be connected with a similar problem for a thin compact super-conducting shell Ω_δ separating the inner *vacuum* domain Ω_s from the outer domain $\hat{\Omega}_s$. Considering the one-particle scattering problem with smooth matching conditions on the inner and the outer components of the boundary of the shell, we again obtain a Lax-Phillips scattering system. Taking into account the non-zero mass of the field on the shell, we see that the low-energy resonances arise from the discrete spectrum of the Dirichlet problem for the Klein-Gordon-Fock equation on the shell. A relevant physical phenomenon was observed on a multi-layer shell constructed of carbon nano-structures (see, for instance, [29]). In that paper, the resonance pumping phenomenon was discovered. Our previous analysis of the super-conducting shells allow us to formulate a question on the super-conduction nature of the carbon shell in the experiment, which would explain the nature of pumping based on the classical Lax-Phillips resonance scattering (see next section).

The fields with nonzero mass play an important role in the transition from the Klein-Gordon-Fock evolution to the Schrödinger evolution. One may guess that other possible experiments revealing an exponential decay in quantum physics can be considered with involvement of some scalar boson fields playing the role of the field of Cooper pairs in above problems. This gives us a pretext to underline a unique role of measurements based on zero-mass fields in quantum physics. In combination with the symmetry breaking and mass creation, these measurements may help to explain the exponential decay and resonance pumping in these experiments.

3 An Example: Analysis of the Model of Decay Observed in a 1D Analog of the Electro-Magnetic Experiment

Our review of the Lax-Phillips technique and the basic results presented in Section 1 shows just the tip of the iceberg. The rest of the estimations, the complex and harmonic

analysis remained under cover. In the previous sections of this paper we provided only a sketch of the results that could be obtained by the classical Lax-Phillips technique for the corresponding multi-dimensional decay problem. We now aim at the simplest 1D model, for which all analytical details of the Lax-Phillips resonance scattering theory can be derived *explicitly* with the use of standard tools of spectral theory of ordinary differential operators. Note that the 1D model of photons was legitimized by [30]. We consider here a 1D model of scattering of 1D photons by a super-conducting shell Ω_δ in a form of a zero-mass Klein-Gordon-Fock equation with quantum well potential supported by $(-a, -\delta)$ with zero boundary condition at the endpoint $x = -a$. The potential on the interval $(-\delta, 0) \equiv \Omega_\delta$ is determined by the mass of photons in a thin surface layer of the super-conductor, presented by a rectangular potential barrier. The quantum well is attached to the positive half-axis $(0, \infty)$:

$$c^{-2}u_{tt} - \frac{\partial^2 u}{\partial x^2} + V_H(x)u = 0,\ -a < x < \infty,\ u(-a) = 0. \tag{7}$$

Instead of the super-conducting layer Ω_δ supporting the non-zero mass photon's field, we may assume that the potential has a a repulsing singularity $H\delta(x)$, $H > 0$, at the origin, $V_H(x) = V(x) + H\delta(x)$, and a smooth real component $V(x)$, $-a < x < 0$. This δ-singularity emulates the condition of domination of the energy of photons by the BCS gap and plays the role of a high potential barrier that separates the inner part and the outer parts $\Omega_s \equiv (-a, 0),$, $\hat{\Omega}_s = (0, \infty)$, with the zero-mass field in the outer vacuum space $x > 0$ and on the inner vacuum space $(-a, 0)$. Changing the "height" H of the barrier, one can approach the limit $H = \infty$, which corresponds to the zero boundary condition $u(0) = 0$ decoupling the inner and the outer subsystems. The role of excitations in the model is played by the one-dimensional "photons" in the outer space $x > 0$. The corresponding excitations inside the well $[-a, 0] = \Omega_s$ are not observed independently, but only due to their connection to the photon's field in vacuum $[0, \infty)$ via the excitations on the shell.

Following our proposal in previous section, we introduce the slow varying component ψ of the wave-function $u = \psi(x)\, e^{-imc^2\hbar^{-1}t}$ on the shell and assume that the variation of the kinetic energy associated with slow variables $\frac{d^2}{dt^2}c^{-2} \parallel \psi_t \parallel^2$ is relatively small and can be neglected so that we get the Schrödinger equation with $\omega = mc^2/\hbar$ (see (5) in the previous section). The corresponding Schrödinger equation describes the evolution of the slow component of the excitation's field in the quantum well, passed from the evolution inside the well to the evolution of the 1D photon's field outside. Analysis of the wave-packets based on the Schrödinger equation (7), derived by the separation of the fast and slow variables, reveals a polynomial decay rate caused by the branching point at the origin $p = 0$ in the plane of the spectral parameter (see [7]). This theoretical proposal was never confirmed experimentally (see the corresponding discussion in Section 2). We conjecture that the realistic decay rate can be theoretically extracted from the original equation (7) based on the analysis of the corresponding Lax-Phillips dynamics (see below and more technical details in [10,15]).

Notice, first of all, that the basic Hilbert space associated with the Schrödinger equation is the space of all square-integrable functions $L_2(-a, \infty)$, while the Hilbert space associated with (7) is an energy-normed space $\mathcal{E}$ of the Cauchy data $\mathbf{u} = (u, c^{-1}u_t) \equiv (u_0, u_1)$,

$$\| \mathbf{u} \|^2_{\mathcal{E}} = \frac{1}{2} \int_{-a}^{\infty} \left[|u_x|^2 + V_H u\bar{u} + c^{-2}|u_t|^2 \right] dx. \tag{8}$$

The basic equation (7) can be represented as a first order equation for the vector of Cauchy data, with a symmetric (self-adjoint) generator $\mathcal{L}$:

$$\frac{1}{i\,c}\frac{\partial \mathbf{u}}{dt} = i \begin{pmatrix} 0 & -1 \\ -\frac{d^2}{dx^2} + V_H & 0 \end{pmatrix} \mathbf{u}. \tag{9}$$

The evolution (9) of the Cauchy data is defined by the unitary group $\exp i\mathcal{L}t \equiv U_t$, which has an orthogonal pair of incoming and outgoing subspaces $\mathcal{D}_{in,out}$, consisting of Cauchy data $\{(u,\, u_x)\}$, $\{(u,\, -u_x)\}$ of the corresponding d'Alembertian waves $\mathbf{u}(x \pm ct)$, and supported by the positive half-axis $0 < x < \infty$, see [10]. The orthogonal complement $\mathcal{K} \equiv \mathcal{E} \ominus [\mathcal{D}_{in} \oplus \mathcal{D}_{out}]$, the corresponding co-invariant subspace, consists of the Cauchy data supported essentially by the quantum well $[-a, 0]$ and equal to $\mathbf{u} = (\text{const}, 0)$ on the half-axis $(0, \infty)$. It is very easy to derive the semi-group property of the evolution reduced onto the co-invariant subspace-the Lax-Phillips semi-group:

$$\mathcal{P}_{\mathcal{K}}\ e^{i\mathcal{L}t}\Big|_{\mathcal{K}} \equiv e^{i\mathcal{B}t},\ t > 0, \tag{10}$$

and calculate the corresponding generator as

$$\mathcal{B} = i \begin{pmatrix} 0 & -1 \\ -\frac{d^2}{dx^2} + V_H & 0 \end{pmatrix}$$

with the zero boundary condition at the end $x = -a$ and the impedance boundary condition at the origin $\left[u_1 + \frac{du_0}{dx}\right]\Big|_{x=+0} = 0$. Similarly, the generator $-\mathcal{B}^+$ of the adjoint semi-group $e^{-i\mathcal{B}^+ t}$ is determined by the same differential expression with the dual impedance boundary condition at the origin $\left[u_1 - \frac{du_0}{dx}\right]\Big|_{x=+0} = 0$. Both the generators $\mathcal{B}, -\mathcal{B}^+$ are dissipative operators (see [10]), with discrete spectrum. It is important that the spectrum of $\mathcal{B}$ is defined by the zeros of the corresponding Lax-Phillips *scattering matrix*, the resonances.

Indeed, the incoming and outgoing subspaces $\mathcal{D}_{in,out}$ of the Cauchy data are constituted by the Cauchy data of D'Alembertian waves $\Phi(x \pm ct)$ supported by the positive half-axis. Then the spectral images of them with the use of the incoming scattered waves Ψ_{in} define the description $\mathcal{J}_{in}$ of the problem in the "incoming" spectral representation of $\mathcal{L}$, attributing $\mathcal{D}_{in}$ to the Hardy class H^2_- of all square-integrable functions admitting an analytic continuation to the lower half-plane $\Im p < 0$ of the spectral parameter p. This spectral representation is defined by the incoming scattered waves of $\mathcal{L}$

$$\mathbf{\Psi}_{in}(x,p) = \begin{pmatrix} \frac{1}{ip} \\ 1 \end{pmatrix} \psi_{in}(x,p), \tag{11}$$

where $\psi_{in}(x,p)$ is the solution of the equation $-\frac{d^2\psi_{in}}{dx^2}+V_H(x)\psi_{in}=p^2\,\psi_{in}$, satisfying the zero boundary condition at the end $x=-a$ and matching the scattering Ansatz

$$\psi_{in}(x,p)=e^{ipx}+S(p)e^{-ipx},x>0,\quad \psi_{out}(x,p)=\bar{\psi}_{in}(x,p)$$

at the origin to an appropriate solution $\varphi_{in,out}(x,p)$ of the original equation $-\frac{d^2\varphi}{dx^2}+V_H(x)\varphi=\lambda\varphi\equiv p^2\,\varphi$ on the well $(-a,0)$, and also satisfying the zero boundary condition at the end $x=-a:\varphi(-a,p)=0$. The corresponding Weyl function $m_H(\lambda)\equiv\varphi'(0,p)\,\varphi^{-1}(0,p)+H=m(\lambda)+H$ has a negative imaginary part in the upper half-plane $\Im\lambda>0$. The stationary scattering matrix is found from the matching condition at the origin, taking into account the δ-function: $[\psi']\Big|_0-H\psi(0)=0$:

$$S(p)=\frac{ip-m_H(\lambda)}{ip+m_H(\lambda)},\quad \lambda=p^2. \tag{12}$$

This function is analytic in the lower half-plane $\Im p<0$, and it has a sequence of zeros p_s, $\Im p_s<0$, which is symmetric with respect to reflection $p_s=-\bar{p}_{-s}$. The scattered waves ψ obtained by matching ψ_{in},ψ_{out} to $\phi_{in,out}$ form a complete orthogonal in $L_2(-a,\infty)$ systems of eigenfunctions of the spectral problem $-\frac{d^2\psi}{dx^2}+V(x)\psi=p^2\,\psi,\ \psi(-a,p)=0$ in $L_2(-a,\infty)$:

$$\delta(x-s)=\frac{1}{2\pi}\int_0^\infty\psi(x,p)\bar{\psi}(s,p)dp,$$

and the corresponding eigenfunctions $\Psi(x,p)=\begin{pmatrix}\frac{1}{ip}\\1\end{pmatrix}\psi(x,|p|)$, $-\infty<p<\infty$, play the role of eigenfunctions of the generator $\mathcal{L}$ of the evolution of the Klein-Gordon-Fock equation, $\mathcal{L}\Psi_{in}(*,p)=p\Psi_{in}(*,p)$. The spectrum of $\mathcal{L}$ is $(-\infty,\infty)$. The incoming spectral representation

$$\mathbf{u}\xrightarrow{\mathcal{J}_{in}}\langle\Psi_{in},\mathbf{u}\rangle_{\mathcal{E}}$$
$$=\frac{1}{2}\int_0^\infty\left[\bar{\Psi}'_{0,in}(x)u'_0(x)+V_H(x)\bar{\Psi}_{0,in}(x)u'_0(x)+\bar{\Psi}_{1,in}(p,x)u_1(x)\right]dx=\mathcal{J}_{in}\mathbf{u} \tag{13}$$

transforms the incoming subspace $\mathcal{D}_{in}$ into the Hardy class H^2_- of square-integrable functions on the real axis and the outgoing subspace $\mathcal{D}_{out}$ gets mapped to the invariant subspace $\bar{S}(p)H^2_+$ of the positive shift semi-group $f(p)\to e^{ipt}f(p)$, $t>0$. Thus, the co-invariant subspace $\mathcal{K}$ is transformed into $H^2_+\ominus\bar{S}H^2_+\equiv K$, and the Lax-Phillips semi-group becomes $P_Ke^{ipt}\Big|_K\equiv e^{iBt}$. In this representation, the spectrum of the generator $B=\mathcal{J}_{in}\mathcal{B}\,\mathcal{J}^+_{in}$ coincides with the zeros $\bar{p}_s$ of $\bar{S}(\bar{p})$, and the eigenfunctions are just given by

$$\phi_s\equiv\bar{S}(\bar{p})\sqrt{2|\Im p_s|}(p-\bar{p}_s)^{-1}. \tag{14}$$

Together with the eigenfunctions

$$\phi^+\equiv\sqrt{2|\Im p_s|}(p-p_s)^{-1} \tag{15}$$

of the adjoint generator B^+, they form a complete bi-orthogonal system in K. So that

$$B = \sum_s \phi_s\rangle \frac{p_s}{\langle \phi_s, \phi_s^+\rangle} \langle \phi_s^+ \quad \text{and} \quad e^{iBt} = \sum_s \phi_s\rangle \frac{e^{i\bar{p}_s t}}{\langle \phi_s, \phi_s^+\rangle} \langle \phi_s^+. \tag{16}$$

Here, $\langle \phi_s, \phi_s^+\rangle = \prod_{r\neq s} \frac{1-\bar{p}_s/\bar{p}_r}{1-p_s/p_r} \equiv \Pi_s$. Recall that the system $\{\phi_s\}, \{\phi_s^+\}$ is similar to an orthonormal basis if and only if the Carleson condition (see [15]) is fulfilled:

$$\inf_r \prod_{s\neq r} \frac{|p_s - p_r|}{|\bar{p}_s - p_r|} > 0.$$

Under the Carleson condition there exists an orthogonal basis $\{\nu_s\}$ that is connected with the normalized families $\{\phi_s\}, \{\phi_s^+\}$ by an invertible transformation:

$$\phi_s = \mathcal{T}\nu_s, \quad \phi_s^+ = \left[\mathcal{T}^{-1}\right]^+ \nu_s, \quad \text{with} \quad \|\mathcal{T}\|, \left\|\left[\mathcal{T}^{-1}\right]^+\right\| < \infty.$$

Unfortunately, the Carleson condition is never fulfilled for potential of the type V_H. However, this condition may be fulfilled for the corresponding polar problem with the potential substituted by a density, a coefficient in front of the spectral parameter, and the non-stationary equation $\rho/c^2\ u_{tt} + u_{xx} = 0$.

Notice that the eigenvalues $\bar{p}_s, p_s$ of $\mathcal{B}$, $\mathcal{B}^+$ depend on the parameter H and approach the eigenvalues of the Schrödinger operator $L_H = -\frac{\partial^2 u}{\partial x^2} + V_H(x)$ in $L_2(-a, 0)$ with zero boundary conditions at the ends $-a, 0$. The resonances $\bar{p}_s$, the zeros of the Lax-Phillips Scattering matrix $S_{LP} = [S_H(p)]^{-1}$,

$$S_{LP}(p) = \frac{ip + [m(\lambda) + H]}{ip - [m(\lambda) + H]}, \quad \text{with} \quad \lambda = p^2,$$

can be found from the equation $ip + [m(\lambda) + H] = 0$. For large values of H, the resonances are situated in the upper half-plane near the poles of $m(\lambda)$, the eigenvalues λ_s^D of the Dirichlet spectral problem on the interval $(-a, 0)$:

$$ip + H + \frac{q_s}{\lambda - \lambda_s^D} + b_s = 0.$$

Denoting $\lambda_s^D = [p_s^D]^2$, we have the approximate expression for resonances p_s approaching p_s^D as $H \to +\infty$,

$$p_s \approx p_s^D + \frac{q_s(ip_s^D + H)}{2p_s^D(|p_s^D|^2 + (b_s + H)^2)} \approx p_s^D + \frac{q_s}{2p_s^D H} + \frac{iq_s}{2H^2}. \tag{17}$$

The eigenfunctions ϕ_s, ϕ_s^+ of the generators $\mathcal{B}$, $\mathcal{B}^+$ of the Lax-Phillips semi-group are calculated in spectral representation of the generator $\mathcal{L}$ of the evolution of their second components, for large H are close to the bound states of the eigenvalues $\left(p_s^D\right)^2$.

We can develop even more constructive explicitly solvable **abstract model of a quantum dot** based on a zero-range potential with an inner structure, which allows a reasonably precise fitting to the experimental data, similarly to the one suggested in [37], which would explain recently discovered giant "topological resonances " occurring in scattering of the electromagnetic waves by carbon nano-structures, see, for instance, [29]. But we leave this interesting material for another publication.

4 Physics of the Exponential Decay via the Lax-Phillips Scheme

The spectral analysis of the Lax-Phillips semi-group, described in the brief review above, was based, on the one hand, on the presence of the continuous spectrum of the zero-mass Klein-Gordon-Fock evolution group generator $\mathcal{L}$, and, on the other hand, on the observation that the group possesses a pair of orthogonal incoming and outgoing subspaces.

More specifically, the continuous spectrum of $\mathcal{L}$ fills in the whole real axis and the parts of the evolution in the incoming and outgoing subspaces are unitarily equivalent to the negative and positive semi-groups generated (in the p-representation) by the shift $f \to e^{ipt}f$ in the subspaces H^2_- and $S_{LP}H^2_+$, respectively. As a result, the remaining part of the corresponding positive evolution semi-group e^{iBt}, $t > 0$, reduced onto the co-invariant subspace $\mathcal{J}_{in} : \mathcal{K} \longrightarrow H^2_+ \ominus S_{LP}H^2_+ \equiv K$, is unitarily equivalent to the Lax-Phillips semi-group

$$P_{\mathcal{K}}U_t\big|_{\mathcal{K}} \xrightarrow{J^+_{in}} P_K e^{ikt}\big|_K, \quad t > 0.$$

4.1 The Lax-Phillips Concept as a Bridge between the WW and KF Concepts of Decay

One can see that these Lax-Phillips features were waived in the KF concept. Without them, the concept cannot guarantee the exponential decay. Adding these details to the KF proposal makes it sufficient not only to explain the exponential decay, but also to construct a solid bridge between the WW and KF schemes and even give a spectral meaning to resonances, which would be absolutely impossible in the pure Schrödinger approach.

Indeed, firstly, the spectrum of the Lax-Phillips semi-group $P_K e^{ipt}\big|_K = e^{iBt}$, $t > 0$, associated with a compact scatterer, is discrete, which meets the basic requirement of the WW approach. Secondly, the corresponding eigenfunctions in the incoming spectral representation $\mathcal{J}_{in} : \mathcal{D}_{in} \xrightarrow{\mathcal{J}_{in}} H^2_-$ are calculated explicitly, as illustrated by (14,15) and, moreover, the corresponding eigenvalues of the dissipative generator coincide with the zeros $\bar{p}_s$ of the scattering matrix S_{LP}. In the case when the singular spectrum of the Lax-Phillips generator is absent and the discrete spectrum is simple, one can use a rational approximation to the scattering matrix given by a finite Blaschke product $S^N_{LP} = \Theta_0 \prod_{s=1}^{N} \frac{p-\bar{p}_s}{p-p_s}$, $\Im p_s < 0$, with Θ_0 a unitary constant.

Based on this approximation we can obtain an approximate description of the exponential decay. In particular, the Lax-Phillips evolution of an initial state that coincides with the eigenvector ϕ_s can be described explicitly as

$$e^{i\mathcal{B}t}\phi_s = e^{i\bar{p}_s t}\ \phi_s.$$

Here the normalized eigenvectors ϕ_s are to be found as solutions of the impedance boundary problem for the Schrödinger equation, with a subsequent restriction on the co-invariant subspace, and the decrements $\Im\bar{p}_s$ can be obtained from the asymptotics (17). The resulting formula can be considered to be a unification of both the Fock–Krylov and the Weisskopf–Wigner approaches to resonances.

In a previous example, see Section 3, we derived the formula (16) based on interaction of the inner quantum system (on a compact domain $[-a,0]$) with the Klein-Gordon-Fock equation on the exterior domain defined by the appropriate matching at the common boundary $x=0$. Using the Lax-Phillips approach we recovered the spectral meaning of resonances interpreting them as eigenvalues of the generator of the Lax-Phillips semi-group. In this particular case, the generator has a discrete spectrum located in the neighborhood of the spectrum of the unperturbed conservative system, the one which is defined by the same Schrödinger differential equation with zero boundary conditions at the end-points of the interval $[-a,0]$.

This permits to observe the WW concept of the decay from the LP spectral point of view. In particular, in [4], an averaged decay is considered. Using the spectral representation for the Lax-Phillips semi-group, one can calculate the decrement by observing the decay on the initial stage for a relatively small t. Indeed, taking into account that $\langle \psi_r, \psi_r^+ \rangle = \Pi_r$ and that $\langle \psi_s, \psi_r \rangle = \frac{\sqrt{2\Im \bar{p}_s}\sqrt{2\Im \bar{p}_r}}{\Im \bar{p}_s + \Im \bar{p}_r}$, we get:

$$\left\| P_K U_t \big|_K u \right\|^2 = \sum_{s,r}^{N} e^{i[\bar{p}_s - i p_r]t} \langle \phi_s, \phi_r \rangle \frac{\langle \phi_s^+, u \rangle \ \langle \phi_r^+, u \rangle}{\langle \phi_s, \phi_s^+ \rangle \ \langle \phi_r, \phi_r^+ \rangle}$$

$$\leq \sum_{s,r}^{N} e^{-[\Im \bar{p}_s + \Im \bar{p}_r]t} \left| \langle \phi_s, \phi_r \rangle \frac{\langle \phi_s^+, u \rangle \ \langle \phi_r^+, u \rangle}{\langle \phi_s, \phi_s^+ \rangle \ \langle \phi_r, \phi_r^+ \rangle} \right| \tag{18}$$

$$= \sum_{s,r}^{N} e^{-[\Im \bar{p}_s + \Im \bar{p}_r]t} \ \frac{\sqrt{2\Im \bar{p}_s}\sqrt{2\Im \bar{p}_r}}{\Im \bar{p}_s + \Im \bar{p}_r} \frac{\langle \phi_s^+, u \rangle \ \overline{\langle \phi_r^+, u \rangle}}{\Pi_r \ \bar{\Pi}_s}.$$

One can see from (18) that $\left\| P_K U_t \big|_K u \right\|^2 \leq C(u) e^{-\gamma t}$. The integral parameter γ can be estimated based on the asymptotics of (18) for small t. Thus, we have

$$C(u)\gamma \approx t^{-1} \left[\| \ P_K U_t \big|_K u \ \|^2 - \| \ P_K u \ \|^2 \right]$$

$$\leq 2 \sum_{s,r}^{N} \frac{\sqrt{\Im \bar{p}_s}\sqrt{\Im \bar{p}_r} \langle \phi_s^+, u \rangle \ \overline{\langle \phi_r^+, u \rangle}}{\Pi_r \ \bar{\Pi}_s}, \tag{19}$$

as $t \to 0$. Note that the incoming spectral representation transforms $\mathcal{K}$ in to $K = H_+^2 \ominus S_{LP} H_+^2$. Then, for $u \in K$, we have $\langle \psi_s^+, u \rangle = \frac{1}{2\pi} \int_R \frac{u(p)dp}{p - \bar{p}_s} = iu(\bar{p}_s)$, with u calculated as $\mathcal{J}_{in}\mathbf{u}$ according to (13).

The ultimate formula (19) bears some features of the exponential decay formulae derived according to the WW and KF concepts. Indeed, the derivation of the exponential decay rate in the WW manner presented in [32], see the formula (80.13, chapter IX), gives the decay rate via the matrix elements of the perturbation in the interaction representation. If the perturbation is small, then the decay rate of the LP resonance state ϕ_s, see can be interpreted as the decay of the bound state state with the eigenvalue $\left(p_s^D\right)^2$ close to the resonance p_s, according to (17).

4.2 The Spectral Meaning of Resonances

Nevertheless, bridging together both of the contradictory concepts of the WW and KF is not the main achievement of the Lax-Phillips point of view. We suggest that the main achievement is the discovery of the *spectral meaning of resonances:* once we reduce the unitary evolution onto the co-invariant space $K = H^2_+ \ominus S_{LP}H^2_+$, the result is represented, via $\mathcal{J}_{in}$, by the Lax-Phillips semi-group

$$e^{iBt}u = \sum_s e^{-i\bar{p}_s t}\frac{\phi_s\langle\phi_s^+, u\rangle}{\langle\phi_s, \phi_s^+\rangle}. \tag{20}$$

Here the "Gamov vectors" ϕ_s, ϕ_s^+ have an unambiguous spectral meaning as the eigenvectors of the Lax-Phillips semigroup generator $\mathcal{B}$, and $\bar{p}_s$ are the corresponding eigenvalues. The spectrum of the generator is discrete, but the whole picture of the restricted evolution on the co-invariant subspace arose because of the specific features of the Lax-Phillips dynamics, first of all of those that are due to the presence of the constant multiplicity continuous spectrum on $R = (-\infty, \infty)$ for the shift group, exactly as it was expected in [5]. But the authors of [5] missed another essential point: the orthogonality in the energy-normed space of the incoming and outgoing invariant subspaces of the wave equation evolution.

So, one can conclude that in the special case when the condition of orthogonality on the incoming and outgoing subspaces for the wave evolution is satisfied, the KF scheme of the exponential decay is confirmed mathematically. In that case, both the KF and WW schemes give expected results including that of the discreteness of the spectrum of resonances.

4.3 Quality of an Oscillation System and Resonance Pumping

Note that the spectral decomposition for the Lax-Phillips semi-group ensures an exponentially decaying evolution for any single term of the spectral expansion of the semigroup, with the decrement $\Im p_s$. It is customary to interpret the slow decay of the terms of the spectral expansion as a "high quality" of the corresponding oscillatory system.

There is, in principle, another method for the estimation of quality of the oscillatory system that is based on estimating the growth of the amplitudes of forced oscillations under periodic excitation. In radio-physics, these two estimations of "quality", based on the decay and on the "pumping", are considered to be alternative estimations of the quality, but the equivalence of them needs a justification using the spectral formulation of the decay problem.

Indeed, let us consider the periodic excitation of the oscillatory system in the form

$$\frac{1}{i}\frac{du}{dt} = Bu + e^{i\omega t}\nu$$

with zero incident value. Using the spectral representation of the Lax-Phillips semigroup, one obtains that

$$u(t) = \sum_s i\int_0^t e^{i(\omega-\bar{p}_s)\tau}d\tau e^{i\bar{p}_s t}\frac{\phi_s\langle\phi_s^+, \nu\rangle}{\langle\phi_s, \phi_s^+\rangle} = e^{i\omega t}\sum_s \frac{1-e^{i(\bar{p}_s-\omega)t}}{(\omega-\bar{p}_s)}\frac{\phi_s\langle\phi_s^+, \nu\rangle}{\langle\phi_s, \phi_s^+\rangle}.$$

The phenomenon of resonance pumping is then observed when the frequency ω is close to one of the eigenvalues of the Lax–Phillips generator. For instance, if $\bar{p}_s - \omega = -i\Im p_s$, recall that $-\Im p_s > 0$, then the forced oscillation regime is

$$u(t) = \frac{e^{\Im p_1 t} - 1}{\Im p_1}\, e^{i\omega t} \frac{\phi_1 \langle \phi_1^+, \nu\rangle}{\langle \phi_1, \phi_s^+\rangle} + \sum_{s>1} \frac{1 - e^{i(\bar{p}_s - \omega)t}}{i(\omega - \bar{p}_s)} \frac{\phi_s \langle \phi_s^+, \nu\rangle}{\langle \phi_s, \phi_s^+\rangle}.$$

Therefore, the forced amplitude of the first term is linearly growing with time, until $t \approx (\Im p_1)^{-1}$, but eventually, at large time scale, it saturates at the value

$$-(\Im p_1)^{-1} \frac{\phi_1 \langle \phi_1^+, \nu\rangle}{\langle \phi_1, \phi_1^+\rangle}.$$

4.4 Complementarity of the Lax-Phillips Scattering Scheme and the Quantum Zeno Effect

The celebrated Zeno Paradox, see [31], can also be treated from the viewpoint of the Lax–Phillips evolution. Indeed, consider the Lax-Phillips evolution defined by the unitary group $U_t = e^{i\mathcal{L}t}$ in an energy normed space $\mathcal{E}$ and suppose that the group possesses an orthogonal pair $\mathcal{D}_{in,out}$ of incoming and outgoing subspaces.

The restriction $P_{\mathcal{K}} U_t P_{\mathcal{K}}$, $t > 0$, of the positive semi-group onto the co-invariant subspace $\mathcal{K} \equiv \mathcal{E} \ominus [\mathcal{D}_{in} \oplus \mathcal{D}_{out}]$ is the Lax-Phillips semi-group $P_{\mathcal{K}} U_t P_{\mathcal{K}} \equiv e^{i\mathcal{B}t}$ with the simple (with no self-adjoint/symmetric parts) dissipative generator $\mathcal{B}$ with discrete spectrum (parameterized by the characteristic function S_{LP}, the Lax-Phillips scattering matrix, defined by a Blaschke product). Introducing the amplitude $\langle e^{i\mathcal{L}t}\phi, \phi\rangle_{\mathcal{E}} \equiv a_\phi(t)$ of the returning probability $p_t \equiv \bar{a}_\phi\, a_\phi$, for "smooth" elements $\phi \in \mathcal{K} \cap \mathcal{D}_{\mathcal{B}}$ such that $\mathcal{B}\phi \in \mathcal{D}_{\mathcal{B}}$ we represent the amplitude as $a(t) = \langle e^{i\mathcal{B}t}\phi,\, \phi\rangle_{\mathcal{E}} = 1 + it\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}} - \frac{t^2}{2}\langle \mathcal{B}^2\phi,\, \phi\rangle_{\mathcal{E}} + \dots$. Then, Taylor's Theorem up to second order applied to the returning probability yields

$$p(t) = \bar{a}_\phi\, a_\phi = 1 - 2t\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}} - t^2\left[\Re\langle \mathcal{B}^2\phi,\, \phi\rangle_{\mathcal{E}} - |\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}}|^2\right] + \cdots .$$

If $\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}} \neq 0$, then $1 - 2t\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}} \approx e^{-2t\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}}}$, and hence, despite a multiple control of the evolution we have $p(t) \approx [p(t/n)]^n$. This is the case of an exponential decay with the decrement $\Gamma = 2\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}}$. The alternative condition $\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}} = 0$ implies

$$p(t) = 1 - t^2\left[\Re\langle \mathcal{B}^2\phi,\, \phi\rangle_{\mathcal{E}} - |\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}}|^2\right] + \ldots \approx 1 - At^2$$

which would give the following asymptotics for the probability under the evolution with the multiple control at the sequence of moments $t_m = \frac{m}{n}\, t$, $m = 1, 2, \dots$,

$$[p(t/n)^n] \approx [1 - A/n^2]^n \approx [e^{-A}]^{1/n} \approx 1 \;\text{ as }\quad t \to \infty.$$

This result corresponds to the quantum Zeno effect. The condition $\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}} = 0$ is not compatible with dissipativity of the simple (with no self-adjoint parts) generator $\mathcal{B}$ with Riesz-basis property of eigenfunctions. Indeed the opposite condition $\langle \Im\mathcal{B}\phi, \phi\rangle > 0$ is obviously satisfied for all vectors from the domain of $\mathcal{B}$ in the coinvariant subspace,

if the system of it's eigenvectors is a Riesz basis. Thus, we conclude that the Zeno effect is not compatible with the Lax–Phillips evolution for elements ϕ from the coinvariant subspace such that $\Im\langle \mathcal{B}\phi,\, \phi\rangle_{\mathcal{E}} > 0$.

Vice versa, the general Schrödinger type unitary evolution $U_t\phi = e^{iLt}\,\phi$ of a smooth state ϕ is compatible with the Zeno effect (whenever L is a self-adjoint generator in the Hilbert space E).

Indeed, the corresponding infinitesimal evolution for a smooth normalized state ϕ yields

$$p(t) = \langle e^{iLt}\,\phi, \phi\rangle_{\mathcal{E}} \langle\, \phi, e^{iLt}\,\phi\rangle_{\mathcal{E}} \approx 1 - t^2\left[\langle L^2\phi, \phi\rangle_{\mathcal{E}} - (\langle L\phi, \phi\rangle_{\mathcal{E}})^2\right] + \dots\ .$$

Hence, in an experiment with the multiple control at the moments of time $t_m = \frac{m}{n}\,t$, $m = 1, 2, \dots$, we obtain:

$$[p(t/n)]^n \approx \left(1 - \frac{t^2}{n^2}\left[\langle L^2\phi, \phi\rangle_{\mathcal{E}} - (\langle L\phi, \phi\rangle_{\mathcal{E}})^2\right]\right)^n$$

$$\approx e^{-\left[\langle L^2\phi, \phi\rangle_{\mathcal{E}} - (\langle L\phi, \phi\rangle_{\mathcal{E}})^2\right] t^2 n^{-1}} \to 1, \text{ when } n \to \infty.$$

This corresponds to the standard zeno effect in quantum mechanics, see [8]. It is worth mentioning that quantum mechanics description of dynamics and probability is not intrinsically involved in that. But probability arises as a detail of the measurement process: it is clearly seen from the preceding analysis that the interplay between the dynamics and the measurement process is different for the Schrödinger evolution [8] and for the Lax-Phillips one.

5 Conclusion

Our version of matching of a zero-mass field in the outer space with the Schrödinger evolution on the inner space of the quantum system allows one to derive the exponential decay based on the classical Lax-Phillips technique. Contrary to the constructions suggested in [22,23] and those in the recent papers [25,26], we use explicit functional model formulae for the eigenvalues and eigenvectors of the corresponding dissipative generator that gives rise to the reduced dynamics on the corresponding co-invariant subspace. For low energy, the dynamics on the inner space is matched with the corresponding Schrödinger dynamics that provides the standard probabilistic interpretation of the wave-function but would formally produce non-exponential terms in the large-time scale. But the original dynamics, before being reduced to Schrödinger's scenario, exhibits an exponential decay for large time, with non-exponential terms absent. Our approach also reveals the spectral meaning of the resonances and the resonance states, and permits to bridge, on this base, the alternative concepts of resonances and the exponential decay proposed by Weisskopf–Wigner and Krylov–Fock. In turn, this proves that the lifetime of a resonance and the velocity of the resonance pumping are directly connected. We also establish the duality between the exponential decay and the absence of the quantum Zeno effect on resonance initial data for the quantum system under a permanent control.

Acknowledgement. We are grateful to Professor L. Prokhorov for a profound discussion of massive photons in superconductors and the relevant references provided, and to K. K. Makarov who helped us a lot with the text improvement on the last stage of our work. One of the authors (B.P.) is grateful to the Russian Academy for the support from the grant RFBR 03-01-00090.

References

1. Dirac, P.A.M.: Development of the physicists's conception of Nature. In: Mehra, J. (ed.) The Physicists Conception of Nature, pp. 1–14. D. Reidel Publ. (1973)
2. von Neumann, J.: Mathematical Foundations of Quantum Mechanics. Princeton University Press, Princeton (1996) (12th printing)
3. Gamow, G.: Zur Quantentheorie des Atomkernes. Zeitshrift für Physik 51, 204–2012 (1928)
4. Weisskopf, V.E., Wigner, E.P.: Zeitshrift für Physik 63, 54 (1930); 65, 18 (1930)
5. Fock, V.A., Krylov, V.A.: Journal of Experimental and Theoretical Physics (USSR) 17, 93 (1947)
6. Fock, V.A.: Selected Works. Quantum Mechanics and Quantum Field Theory. In: Faddeev, L.D., Khalfin, L.A., Komarov, I.V. (eds.), Chapman & Hall/CRC (2004)
7. Khalfin, L.: On the theory of decay of a quasi-stationary state. Soviet Phys. Doklady 2, 340 (1958)
8. Sakurai, J.: Modern Quantum Mechanics. Revised Edition. Addison-Wesley (1994)
9. Titchmarsh, E.C.: Eigenfunction Expansion Associated with Second-order Differential Equations, Part 1. Clarendon Press, Oxford (1962)
10. Lax, P., Phillips, R.: Scattering Theory. Academic Press, New York (1967)
11. Hegerfeldt, G.C.: Causality, particle localization and positivity of the energy. In: Irreversibility and Causality: Semigroups and Rigged Hilbert Spaces. Lecture Notes in Physics, vol. 504, pp. 238–245 (1998)
12. Beurling, A.: On two problems concerning linear transformations in Hilbert Space. Acta. Math. 81, 239–255 (1948)
13. Nagy, B.S., Foiaş, C.: Harmonic Analysis of Operators on Hilbert Space. Akademiai Kiado, Budapest (1970)
14. Koosis, P.: Introduction to Hp Spaces, 2nd edn. Cambridge University Press, Cambridge (1998)
15. Pavlov, B.: Spectral analysis of a dissipative singular Schrödinger operator in terms of a functional model. In: Shubin, M. (ed.) Partial Differential Equations. Encyclopedia of Mathematical Sciences, vol. 65, pp. 87–153. Springer, Heidelberg (1995)
16. Nikol'skii, N.K., Khruschev, S.V.: A functional model and some problems of the spectral theory of functions. Trudy Mat. Inst. Steklov. 176, 97–210, 327 (1987)
17. Krein, M.G.: Selected Works. II: Banach Spaces and Operator Theory, Natsional'naya Akademiya Nauk Ukrainy, Institut Matematiki, Kiev (1996) (Russian)
18. Krein, M.G.: Selected Works. III. Topics in Differential and Integral Equations and Operator Theory. In: Gohberg, I. (ed.), Birkhauser Verlag, Basel (1983)
19. Adamjan, V.M., Arov, D.Z.: On scattering operators and contraction semigroups in Hilbert space. Dokl. Akad. Nauk SSSR 165, 9–12 (1965) (Russian)
20. Livshits, M.S.: Method of non-selfadjoint operators in the theory of waveguides. Radio Engineering and Electronic Physics, American Institute of Electrical Engineers 1, 260–275 (1962)
21. Pavlov, B.: The theory of extensions and explicitly-soluble models. Russian Math. Surveys 42(6), 127–168 (1987)
22. Flesia, C., Piron, C.: Helv. Phys. Acta 57, 697 (1984)

23. Horwitz, L.P., Piron, C.: Helv. Phys. Acta 66, 694 (1993)
24. Strauss, Y., Horwitz, L.P., Eisenberg, E.: Representation of quantum mechanical resonances in the lax-Phillips Hilbert space. Journal of Mathematical Physics 41, 12 (2000)
25. Baumgartel, H.: Gamov vectors for resonances: a Lax-Phillips point of view. International Journal of Theoretical Physics 46(8), 1960–1985 (2007)
26. Baumgartel, H.: Resonances of quantum-mechanical scattering systems and lax-Phillips scattering theory. Journal of Mathematical Physics 51(113508), 1–20 (2010)
27. Ginzburg, V., Landau, L.: Toward the superconductivity theory. Zhurnal Eksp. Yheoret. Physics 29, 1064 (1950) (Russian)
28. Okun, L.B.: Leptons and Quarks. North Holland, Amsterdam (1981)
29. Ponomarev, A., Yudovich, M., Gruzdev, M., Yudovich, V.: Theoretical estimations of topological factor in interaction of the nano-particles with electromagnetic waves. Scientific Israel-Technological Advancements 11(3), 20–26 (2009)
30. Gribov, V.N.: Quantum Electrodynamics, Moscow, Igevsk (2001) (Russian)
31. Misra, B., Sudarshan, E.C.G.: The Zeno paradox in quantum theory. Journal of Mathematical Physics 18(4), 753–756 (1977)
32. Davydov, A.S.: Quantum Mechanics, ch. IX, Section 80. Pergamon (1965),
33. Krasnosel'skij, M.A.: On self-adjoint extensions of Hermitian operators. Ukrainskij Mat. Journal 1, 21 (1949) (Russian)
34. Shirokov, J.: Strongly singular potentials in three-dimensional quantum mechanics. Teor. Mat. Fiz. 42(1), 45–49 (1980) (Russian)
35. Albeverio, S., Kurasov, P.: Singular Perturbations of Differential Operators. London Math. Society Lecture Note Series, vol. 271. Cambridge University Press (2000)
36. Akhiezer, N.I., Glazman, I.M.: Theory of Linear Operators in Hilbert Space, vol. 1. Frederick Ungar, Publ., New-York (1966)
37. Adamyan, V.A., Calude, C.S., Pavlov, B.S.: Transcending the limits of Turing computability. In: Hida, T., Saitô, K., Si, S. (eds.) Quantum Information Complexity. Proceedings of Meijo Winter School 2003, pp. 119–137. World Scientific, Singapore (2004)
38. Pavlov, B.: A star-graph model via operator extension. Mathematical Proceedings of the Cambridge Philosophical Society 142(02), 365–384 (2007)
39. Yafyasov, A., Martin, G., Pavlov, B.: Resonance one-body scattering on a junction. Nanosystems: Physics, Chemistry, Mathematics 1(1), 108–147 (2010)

Randomness Increases Order in Biological Evolution

Giuseppe Longo and Maël Montévil

CNRS and École Normale Supèrieure, Paris, France
CREA, École Polytechnique, Paris, France
{Giuseppe.Longo,Mael.Montevil}@ens.fr

Abstract. In this text, we revisit part of the analysis of anti-entropy in [4] and develop further theoretical reflections. In particular, we analyze how randomness, an essential component of biological variability, is associated to the growth of biological organization, both in ontogenesis and in evolution. This approach, in particular, focuses on the role of global entropy production and provides a tool for a mathematical understanding of some fundamental observations by Gould on the increasing phenotypic complexity along evolution. Lastly, we analyze the situation in terms of theoretical symmetries, in order to further specify the biological meaning of anti-entropy as well as its strong link with randomness.

Notions of entropy are present in different branches of physics, but also in information theory, biology ... even economics. Sometimes, they are equivalent under suitable transformations from one (more or less mathematized) domain to another. Sometimes, the relation is very mild, or may be at most due to a similar formal expressions. For example, one often finds formulas describing a linear dependence of entropy from a quantity formalized as $\sum_i p_i \log(p_i)$, where p_i is the "probability" of the system to be in the i-th (micro-)state. Yet, different theoretical frames may give very different meanings to these formulas: somehow like a wave equation describing water movement has a similar mathematical formulation as Schrdingers wave equation (besides some crucial coefficients), yet water waves and quantum state functions have nothing to do with each other. Another element seems though to be shared by the different meanings given to entropy. The produ! ction of entropy is strictly linked to *irreversible* processes.

But ... what is entropy? The notion originated in thermodynamics. The first law of thermodynamics is a conservation principle for energy. The second law states that the total entropy of a system will not decrease other than by increasing the entropy of some other system. Hence, in a system isolated from its environment, the entropy of that system will tend not to decrease.

More generally, increasing entropy corresponds to *energy dispersion*. And here we have the other element shared by the different views on entropy: in all of its instances, it is linked to randomness, since diffusions, in physics, are based on *random walks*. Thus, energy, while being globally preserved, diffuses, randomly. In particular, heat flows from a hotter body to a colder body, never the inverse. Only the application of work (the imposition of order) may reverse this flow.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 289–308, 2012.

As a matter of fact, entropy may be locally reversed, by pumping energy. For example, a centrifuge may separate two gazes, which mixed up by diffusion. This separation reduces the ergodicity (the amount of randomness, so to say) of the system, as well as its entropy.

Living beings construct order by absorbing energy. In Schrdingers audacious little book, *What is life?* [32], it is suggested that organisms *also* use order to produce order, which he calls *negentropy*, that is entropy with a negative sign. And this order is produced by using the order of the chromosomes a-periodic structure (his audacious conjecture) *and* by absorbing organized nutrients (dont we, the animal, eat mostly organized fibers?). Of course, a lot can be said, now, against these tentative theorizations by the great physicist.

But is really entropy the same as disorder? There is a long lasting and sound critique, in physics, of the myth of entropy as disorder. F. L. Lambert (see [16]) is a firm advocate of this critical attitude. This is perfectly fair since entropy is just energy dispersal in physics, regardless of whether the system is open or closed[1]. Yet, as explained in [13], *in physics, a lowered energy state is not necessarily disorder, because it simply results in the identical molecule with a lowered energy state. The fact that such a molecule might be biologically inactive may not concern the physicist, but it definitely does concern the biologist* .... In this perspective, it is then sound to relate entropy to ! disorder in biological dynamics: a lesser activity of a molecule may mean metabolic instability, or, more generally, less coherent chemical activities of all sorts. As a consequence, this may result in less bio-chemical and biological order.

In either case, though, and by definition, entropy has to be related to energy dispersal. As a matter of fact, the analysis of heat diffusion in animals and humans has a long history that dates back to the 30s [12]. Since then, several approaches tried to bridge the conceptual gap between the purely physical perspective and the biologists concern with organization and with its opposite, disorder, in particular when increasing, in aging typically [1,13,24,30].

Let's now summarize the perspective of this paper in a very synthetic way: Evo/devo processes (Evolution and development or ontogenesis) may be globally understood as the "never identical iteration of a morphogenetic process". Randomness is at the core of that " *never identical* iteration". By adding selection and following Gould's remarkable insight, we will in particular understand below the increasing compexity of organisms along Evolution, as the result of a purely random diffusion in a suitable phase space (and its defintion is the crucial issue).

1 Entropy in Ontogenesis

In an organism, the internal entropy production has in primis a physical nature, related to all thermodynamic processes, that is to the transformation and exchange of matter and energy. Yet, we will add to this a properly biological

[1] However, the argument that disorder is an epistemic notion, not suitable to physics, is less convincing, since classical randomness, at the core of entropy, is also epistemic (see above and [2]).

production for entropy, the production due to all *irreversible* processes, including biological (re-)construction, that is both embryogenesis and cell replacement and repair (ontogenesis, globally).

Observe first that, in a monocellular organism, entropy is mostly released in the exterior environment and there are less signs of increasing disorder within the cell. Yet, changes in proteome and membranes are recorded and may be assimilated to aging, see [19,28]. In a metazoan, instead, *the entropy produced, under all of its forms, is also and inevitably transferred to the environing cells, to the tissue, to the organism,* [4]. Thus, besides the internal forms of entropy (or disorder) production, a cell in a tissue, the structure of the tissue itself ... the organism, is affected by this dispersal of energy, as increasing disorder, received from the (other) cells composing the tissue (or the organism). Aging, thus, is also or mostly a tissular and organismic process: in an organism, it is the network of interactions that is affected and that may have a fall-out also in the cellular activities (metabolism, oxidativ! e stress ... , see below).

Moreover, the effect of the accumulation of entropy during life contributes, mathematically, to its *exponential increase* in time. Thus, with aging, this increase exceeds the reconstructive activities, which oppose global entropy growth in earlier stages of life (this theory, articulated in four major life periods, is proposed in [4]). Now, we insist, entropy production, in all its forms, implies increasing disorganization of cells, tissues, and the organism. This, in turn, may be physically and biologically implemented by increasing metabolic instability, oxidative effects, weakening of the structure and coherence of tissues (matrix, collagene's links, tensegrity) ... and many more forms of progressive disorganization [7,6,33,29]. Of course, there may be other causes of aging, but the entropic component should not be disregarded and may also help in proposing a unified understanding of diff! erent phenomena.

Our second observation is that entropy production is due to *all irreversible processes*, both the thermodynamic ones and the permanent, irreversible, (re-)construction of the organism itself. This generating and re-generating activity, from embryogenesis to repair and turnover, is typically biological and it has been mathematically defined as anti-entropy (see [4] and below[2]). In other words, irreversibility in biology is not only due to thermodynamic effects, related to the production of energy, typically, but also to all processes that establish and maintain biological organization — that is, it is concomitantly due to entropy production and its biological opposite, anti-entropy production: embryogenesis, for example, is an organizing and highly irreversible process per se. And it

[2] The word anti-entropy has already been used, apparently only once and in physics, as the mathematical dual of entropy: its minimum coincides with the entropy maximum at the equilibrium, in mixture of gases at constant temperature and volume [8]. This is a specific and a very different context from ours. Our anti-entropy is a new concept and observable with respect to both negentropy and the mathematical dual of entropy: typically, it does not add to an equal quantity of entropy to give 0 (as negentropy), nor satisfies minimax equations! , but it refers to the quantitative approach to biological organization, as opposing entropy by the various forms of biological morphogenesis, replacement and repair.

produces entropy not only by the thermodynamic effects due to energy dispersion, but also, in our view, by the very biological constitutive activities.

Cell mitosis is *never an identical reproduction*, including the non-identity of proteomes and membranes. Thus, it induces an *unequal diffusion of energy* by largely random effects (typically, the never identical bipartition of the proteome). That is, biological reproduction, as morphogenesis, is *intrinsically joint to variability* and, *thus, it produces entropy also by lack of (perfect) symmetries*. By this, it induces *its proper irreversibility*, beyond thermodynamics.

As a comparison, consider an industrial construction of computers. The aim is to produce, in the same production chain, identical computers. Any time a computer is doubled, an identical one (up to observability) is produced and "organization" (locally) grows, at the expenses of energy. Entropy is then produced, in principle, only by the required use and inevitable dispersal of energy, while the construction per se just increases organization, along the production chain. Moreover, if, in the construction chain of computers, one destroys the second computer, you are back with one computer and you can iterate identically the production of the second. The process can be reverted (destroy one computer) and iteratable (produce again an identical machine), by importing a suitable amount of energy, of course. Imperfection should be (and are for 99% of the machine) below observability and functionality: they are errors and "noise".

As we said, it is instead a fundamental feature of life that a cell is *never* identical to the mother cell. This is at the core of biological variability, thus of diversity, along Evolution as well as in embryogenesis (and ontogenesis, as permanent renewal of the organism, never identically). In no epistemic nor objective way this may be considered a result of errors nor noise: variability and diversity are the main "invariants" in biology, jointly to structural stability, which is never identity, and, jointly, they all make life possible.

Thus, while producing new order (anti-entropy), life, as iteration of a never identical and an always *slightly disordered* morphogenetic process, generates also entropy (disorder), by the reproductive process itself. In a metazoan, each mitosis produces two slightly different cells, both different also from the "mother" cell: the asymmetry is a form of disorder and, thus, of entropy growth, within the locally increasing order. And this, of course, in addition to the entropy due to free energy consumption. It is this variability that gives this further, and even more radical, form of irreversibility to all biological dynamics (in Evolution and ontogenesis). There is no way to neither revert nor iterate an evolutionary or embryognetic process: if you kill a cell after mitosis, you are not back to the same original cell and this cell will not iterate its reproduction, *identically*[3].

[3] The incompetent computationalist (incompetent in Theory of Computation), w! ho would say that also computers are not identical and misses the point: the *theory* of programming is based on identical iteration of software processes on reliable hardware, i.e. functionally equivalent hardware (and it works, even in computer networks, see the analysis of primitive recursion and portability of software in [20])). Any biological theory, instead, must deal with variability, *by principle*. As recalled above, variability as never identical iteration, in biology, is not an error.

It should be clear that this theoretical frame concerning the overall increase of entropy in biology says nothing about how this disorganization takes place in the various processes, nor anything about its timetable. The analyses of the detailed phenomena that implement it in ontogenesis are ongoing research projects. So far, we could apply these principles to an analysis of growing complexity in Evolution, as summarized next.

2 Randomness and Complexification in Evolution

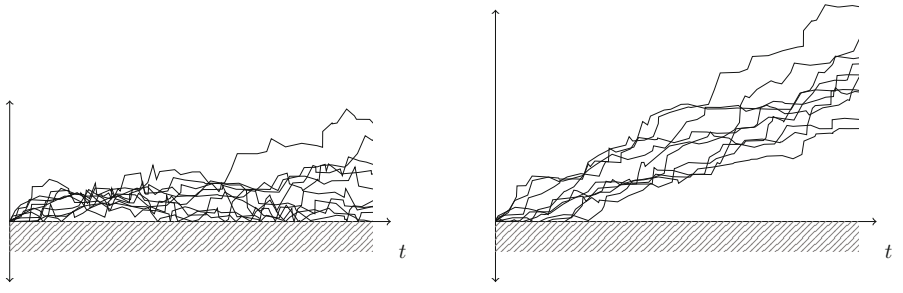
Available energy production and consumption are the unavoidable physical processes underlying reproduction and variability. At the origin of life, bacterial reproduction was (relatively) free, as other forms of life did not contrast it. Diversity, even in bacteria, by random differentiation, produced competition and a slow down of the exponential growth (see diagram 3). Simultaneously, though, this started the early variety of live, a process never to stop.

Gould, in several papers and in two books [10,11], uses this idea of random diversification in order to understand a blatant but too often denied fact: the increasing complexification of life. The increasing complexity of biological structures has been often denied in order to oppose finalistic and anthropocentric perspectives, which viewed life as *aiming* at *Homo sapiens* as the highest result of the (possibly intelligent) evolutionary path.

Yet, it is a fact that, under many reasonable measures, an eukaryotic cell is more complex than a bacterium; a metazoan, with its differentiated tissues and its organs, is more complex than a cell ... and that, by counting also neurons and connections, cell networks in mammals are more complex that in early triploblast (which have three tissues layers) and these have more complex networks of all sorts than diplobasts (like jellyfish, a very ancient animal). This non-linear increase can be quantified by counting tissue differentiations, networks and more, as hinted by Gould and more precisely proposed in [4], that we will extensively summarize and comment, next. The point is: how to understand this change towards complexity without invoking global aims? Gould provides a remarkable answer based on the analysis of the *asymmetric* random diffusion of life. Asymmetric because, by principle, life cannot be less complex than bacterial life[4]. So, reproduction by variability, along evolutionary time and space, randomly produces, just as *possible paths*, also more complex individuals. Some happen to be compatible with the environment, resist and proliferate (a few even very successfully) and keep going, further and randomly producing *also* more complex forms of life. *Also*, since the random exploration of possibilities may, of course, decrease the complexity, no matter how this is measured. Yet, by principle: *any asymmetric random diffusion propagates, by local interactions, the original symmetry breaking along the diffusion.* Thus there is no need for a global design or

[4] Some may prefer to consider viruses as the least form of life. The issue is controversial, but it would not change at all Goulds and ours perspective: we only need a minimum which differs from inert matter.

aim: the random paths that compose *any* diffusion, also in this case help to understand a random growth of complexity, *on average*. On average, as, of course, there may be local inversion in complexity; yet, the asymmetry randomly forces to the right. This is beautifully made visible by figure 1, after [10], page 205. The image explains the difference between a random, but oriented development (on the right, 1b), and the non-biased, purely random diffusive bouncing of life expansion on the left wall, on the left 1a.



(a) Passive trend, there are more trajectories near 0

(b) Driven trend, the trajectories have a drift towards an increased mean

Fig. 1. *Passive and driven trends.* In one case, the *boundary condition*, materialized by a left wall, is the only reason why the mean increases over time, and this increase is therefore slow. In the case of a driven trend, or biased evolution, however, it is the *rule* of the random walk that leads to an increase of the mean over time (there is an intrinsic trend in evolution). Gould's and our approach are based on passive trends, which means that we do ume that there is an intrinsic bias for increasing complexity in the process of evolution.

Of course, time runs on the vertical axis, but ... what is in the horizontal one? Anything or, more precisely, anywhere the random diffusion takes place or the intended phenomenon diffuses in. In particular, the horizontal axis may quantify biological complexity whatever this may mean. The point Gould wants to clarify is in the difference between a fully random vs. a random *and* biased evolution. The biased right image does not apply to evolution: bacteria are still on Earth and very successfully. Any finalistic bias would instead separate the average random complexification from the left wall.

Note that, in both cases, complexity may *locally* decrease: tetrapodes may go back to the sea and lose their podia (the number of folding decreases, the overall body structure simplifies). Some cavern fishes may loose their eyes, in their new dark habitat; others, may lose their red blood cells [31]. Thus, the local propagation of the original asymmetry may be biologically understood as follows: on average, variation by simplification leads towards a biological niches that has *more chances* to be already occupied. Thus, *global* complexity increases *as a purely random effect of variability* and on the grounds of *local effects*: the greater chances, for a "simpler" organism, to bump against an already occupied niche. Thus, more complex variants have just slightly more probabilities to survive and

reproduce — but this slight difference is enough to produce, in the long run, very complex biological organisms. And, of c! ourse, variability and, thus, diversity are grounded on randomness, in biology. No need for finalism nor a priori global aim nor design at all, just a consequence of an original symmetry breaking in a random diffusion on a very peculiar phase space: biomass times complexity times time (see figure 3 for a complete diagram)[5].

Similarly to embryogenesis, the complexification is a form of local reversal of entropy. The global entropy of the Universe increases (or does not decrease), but locally, by using energy of course, life inverses the entropic trend and creates organization of increasing complexity. Of course, embryogenesis is a more canalized process, while evolution seems to explore all "possible" paths, within the ecosystem-to-be. Most turn out to be incompatible with the environment, thus they are eliminated by selection. In embryogenesis increasing complexity seems to follow an expected path and it is partly so. But only in part as failures, in mammals say, reach 50% or more: the constraints imposed, at least, by the inherited DNA and zygote, limit the random exploration due to cell mitosis. Yet, their variability, joint to the many constraints added to development (first, a major one: DNA), is an essential component of cell differentiation. Tissue differentiation i! s, for our point of view, a form of (strongly) regulated/canalized variability along cell reproduction.

Thus, by different but correlated effects, complexity as organization increases, on average, and reverts, locally, entropy. We called *anti-entropy*, [4], this observable opposing entropy, both in evolution and embryogenesis; its peculiar nature is based on reproduction with random variation, submitted to constraints. As observed in the footnote above, anti-entropy differs from negentropy, which is just entropy with a negative sign, also because, when added to entropy, it never gives 0, but it is realized in a very different singularity (different from 0): extended criticality [5,22]. In the next section, we will use this notion to provide a mathematical frame for a further insight by Gould.

3 (Anti-)Entropy in Evolution

In yet another apparently nave drawing, Gould proposes a further visualization of the increasing complexity of organisms along Evolution. It is just a qualitative image that the paleontologist draws on the grounds of his experience. It contains though a further remarkable idea: it suggests the phase space (the space of description) where one can analyze complexification. It is *bio-mass density* that diffuses over *complexity*, that is, figure 2 qualitatively describes the diffusion of the frequency of occurrences of individual organisms per unity of complexity.

[5] By our approach, proposed in [4], we provide a theoretical/mathematical justification of the ZFEL principle in [26], at the core of their very interesting biological analysis: "ZFEL (Zero Force Evolutionary Law, general formulation): In any evolutionary system in which there is variation and heredity, there is a tendency for diversity and complexity to increase, one that is always present but may be opposed or augmented by natural selection, other forces, or constraints acting on diversity or complexity."

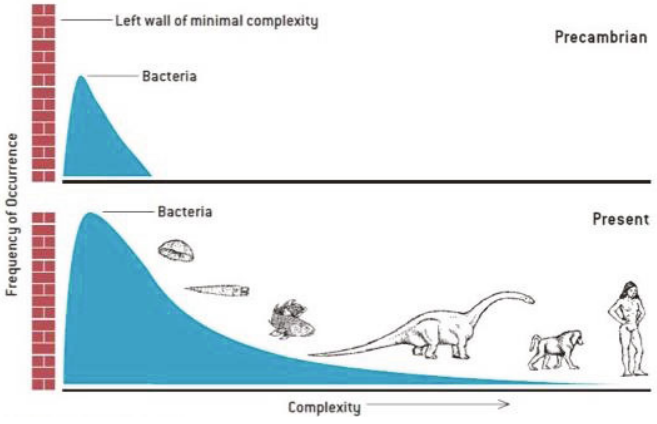


Fig. 2. *Evolution of complexity as understood by Gould.* This illustration is borrowed from [11], page 171. This account is provided on the basis of paleontological observations.

This is just a mathematically naive, global drawing of the paleontologist on the basis of data. Yet, it poses major mathematical challenges. The diffusion, here, is not along a spatial dimension. Physical observables usually diffuse over space in time; or, within other physical matter (which also amounts to diffusing in space). Here, diffusion takes place over an abstract dimension, complexity. But what does biological complexity exactly mean? Hints are given in [11]: the addition of a cellular nucleus (from bacteria to eukaryotes), the formation of metazoa, the increase in body size, the formation of fractal structures (usually — new — organs) and a few more.... In a sense, any added novelty provided by the random bricolage of Evolution and at least for some time compatible with the environment, contributes to complexity. Only a few organisms become more complex over time, but, by the original symmetry breaking mentioned above, this is! enough to increase the global complexity. Of course, the figure above is highly unsatisfactory. It gives two slices over time where the second one is somewhat inconsistent: where are dinosaurs at present time? It is just a sketch, but an audacious one if analyzed closely. Mathematics may help us to consistently add the third missing dimension: time.

A simple form of diffusion equation of q in time t over space x is:

$$\frac{\partial q}{\partial t} = D\frac{\partial^2 q}{\partial x^2} + Q(t,x) \tag{1}$$

where $Q(t,x)$ is a source term describing the situation at the origin of the process. Yet, in our case, the diffusion of this strange quantity, m, a *bio-mass density*, takes place over an even more unusual space, biological complexity, whatever the latter may mean. In [4], we dared to further specify Goulds hints for biological complexity, as a quantity $K = \alpha K_c + \beta K_m + \gamma K_f$ where α, β, and γ are the respective weights of the different types of complexity within the total complexity.

The details are in [4], lets just summarize the basic ideas. So, K_c (combinatorial complexity) corresponds to the possible cellular combinatoric; K_m (morphological complexity) is associated to the forms which arise (connexity and fractal structures); K_f (functional complexity) is associated to the relational structures supporting biological functions (metabolic and neuronal relations). We will discuss this approach in section 4.

K is a tentative quantification of complexity as *anti-entropy*, in particular in biological evolution: the increase of each of its components (more cellular differentiation, more or higher dimensional fractal structures, richer networks ... yield a more "complex" individual). Of course, many more observables and parameters may be taken into account in order to evaluate the complexity of an organism: [4] provides just a mathematical basis and a biological core for a preliminary analysis (an application to ontogenesis as an analysis of *C. Elegans* development is also presented). They suffice though for a qualitative (geometric) reconstruction of Goulds curve, with a sound extension to the time dimension.

As mentioned above, anti-entropy opposes, locally, to entropy: it has the same dimension, yet it differs from negentropy, since it does not sum up to 0, in presence of an equal quantity of entropy. It differs also from information theoretic frame, where negentropy has been largely used, as negentropy (= information) is *independent from coding and Cartesian dimensions*. This is crucial for Shannon as well as for Kolmogorof-Chaitin information theories. Anti-entropy, instead, as defined above, depends on foldings, singularities, fractality ... it is a *geometric* notion, thus, by definition, it is *sensitive to codings* (and to dimension).

The next step is to adapt eq. 1 to these new dimensions. Just use Goulds observables and parameters, m and K, that we specified some more, and write:

$$\frac{\partial m}{\partial t} = D\frac{\partial^2 m}{\partial K^2} + Q(t, K) \tag{2}$$

But what is here $Q(t, K)$, the source term? In order to instantiate Q by a specific function, but also in order to see the biological system from a different perspective (and get to the equation also by an "operatorial approach"), we then gave a central role, as an observable, to the "global entropy production".

Now, in physics, energy, E, is the main observable, since Galileo inertia, a principle of energy conservation, to Noethers theorems and Schrdingers equation. Equilibria, geodetic principles etc directly or indirectly refer to energy and are understood in terms of symmetry principles (see [5]). At least since Schrdinger and his equation, in (quantum) physics, one may view energy as an operator and time as a parameter[6].

As hinted above, in biology, also constitutive processes, such as anti-entropy growth (the construction and reconstruction of organization), *produce entropy*, since they also produce some (new) disorder (recall: at least the proteome, after a mitosis, is non-uniformly and randomly distributed in the new cells). In these

[6] In short, Schrdinger transforms an equation with the structure $E = \frac{p^2}{2m} + V(x)$, where $V(x)$ is a potential, by associating E and p to the differential operators $\partial/\partial t$ and $\partial/\partial x$, respectively, see[4].

far form equilibrium, dissipative (possibly even non-stationary) processes, such as Evolution and ontogenesis, energy turns out to be just one (very important) observable, a parameter to be precise. One eats (and this is essential) and gets fatter: production and maintenance of organization requires energy, but it yields a different observable, one that has a different dimension, tentatively defined by K above, as organization. Typically, in allometric equations, so relevant in biology, energy or mass appear as a parameter. Thus, in our approach, the key observable is organization that is formed or renewed (anti-entropy production).

Moreover, *entropy*, as associated to all irreversible processes, from energy flows to anti-entropy production, is the observable which summarizes all ongoing phenomena; by its irreversibility, it is strongly linked (conjugated) to time.

In summary, we proposed to change the conceptual frame and the conceptual priorities: we associated the global entropy production σ to the differential operator given by time, $\partial/\partial t$ (Schrdinger does this for energy, which is conjugated to time, in quantum physics). Thus, our approach allows to consider biological time as an "operator", both in this technical sense and in the global perspective of attributing to time a key constitutive role in biological phenomena, from evolution to ontogenesis. But how to express this global observable?

In a footnote to [32], Schrdinger proposes to analyze his notion of negative entropy as a form of Gibbs free energy G. We applied this idea to our anti-entropy S^-, where $S^- = -kK$ (k is a positive dimensional constant and K is the phenotypic complexity). Now, $G = H - TS$, where T is temperature, S is entropy and $H = U + PV$ is the systems enthalpy (U is the internal energy, P and V are respectively pressure and volume). By definition, the *metabolism* R has the physical dimension of a power and corresponds to the difference between the fluxes of *generalized free energy* G through the surface Σ:

$$R = \sum [J_G(x) - J_G(x + dx)] = -\sum dx(\mathrm{Div}\, J_G) \tag{3}$$

Locally the conservation (or balance) equation is expressed in the general form:

$$R = -\,\mathrm{Div}\, J_G = \frac{dG}{dt} + T\sigma \tag{4}$$

where σ represents the speed of global production of entropy, that is σ is the entropy produced by *all* irreversible processes, including the production of biological organization or anti-entropy. Thus, the global balance of metabolism for the system of life (the biosphere) has the following form, where S^- and S^+ are anti-entropy and entropy, respectively:

$$R = \frac{dH}{dt} - T\left(\frac{dS^-}{dt} + \frac{dS^+}{dt}\right) + T\sigma \simeq a\frac{dM}{dt} - T\left(\frac{dS^-}{dt} + \frac{dS^+}{dt}\right) + T\sigma \tag{5}$$

where $H \simeq aM$, for a mass M and a coefficient a, which has the magnitude of a speed squared.

$T\sigma$ is a crucial quantity: it contains our σ, modulo the temperature T, since R is a power. $T\sigma$ corresponds to the product of forces by fluxes (of matter, of

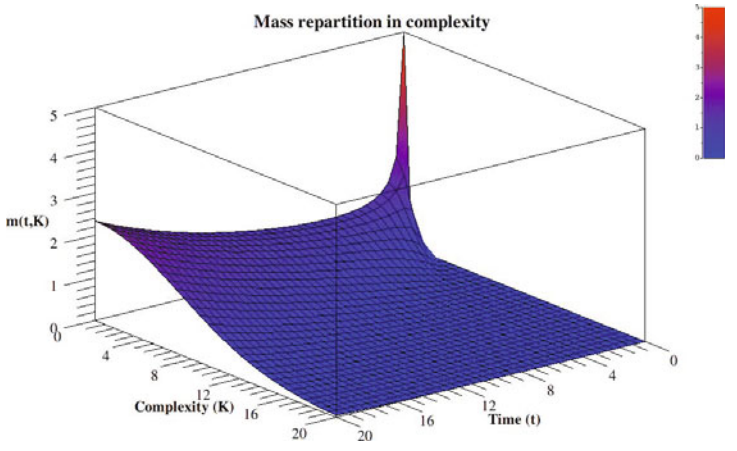


Fig. 3. *Time evolution of mass repartition over anti-entropy.* The initial condition is a finite mass at almost 0 anti-entropy, thus having the shape of a pulse.

energy — chemical energy, for instance — etc.). Now, a flux is proportional to a force, thus to a mass, and hence $T\sigma$ is proportional to a mass squared. It can then be written, up to a coefficient ζ_b and a constant term $T\sigma_0$ as:

$$T\sigma \approx \zeta_b M^2 + T\sigma_0 \tag{6}$$

ζ_b is a constant that depends only on the global nature of the biological system under study and it is 0 in absence of living matter.

Without entering into further details, by using as state function a *bio-mass diffusion function* over complexity K, that is the bio-mass density $m(t, K)$ in t and K, the operatorial approach applied to equation 6 gave us the equation, with a linear source function $\alpha_b m$:

$$\frac{\partial m}{\partial t} = D_b \frac{\partial^2 m}{\partial K^2} + \alpha_b m \tag{7}$$

Its solution, bellow, yields the diagram in figure 3.

$$m(t, K) = \frac{A}{\sqrt{t}} \exp(at) \exp(-K^2/4Dt) \tag{8}$$

In summary, by skipping all the technical details in [4], we could derive, by mathematics and starting from Goulds informal hints, a general understanding as well as the behavior of the Evolution of complexity function w. r. to time. And this fits data: at the beginning the linear source term gives an exponential growth of free bacteria. Then, they complexify and compete. Of course, this diagram, similarly to Goulds, is a global one: it only gives a qualitative, geometric, understanding of the process. It is like looking at life on Earth from Sirius. Analogously to Goulds diagram, the punctuated equilibria, say, and the major

extinctions are not visible: the insight is from too far and too synthetic to appreciate them. It only theoretically justifies Goulds proposal and soundly extends it to time dependence, by mathematically deriving it from general principles: the dynamics of a diffusion by random paths, with an asymmetric origin.! Its source is given by an exponential growth. Life expansion is then bounded, canalized, selected in the interaction with the ever changing, co-constituted ecosystem. The core random complexification persists, while its "tail" exponentially decreases, see equation 8 and figure 3. In that tail, some neotenic big primates, with a huge neural network, turn out to be the random complexification of bacteria, a result of variability and of the immense massacres imposed by selection.

Another (important) analogy can be made with Schrdingers approach (his famous equation, not his book on life) and further justifies the reference to it for the analysis of this (rather ordinary) diffusion equation. Schrdinger dared to describe the deterministic evolution of the wave function in Quantum Mechanics as the *dynamics of a law of probability* (and this gives the intrinsic indetermination of the quantum system). We synthetically represented Biological Evolution as the *dynamics of a potential of variability*, under the left wall constraint. Again, this idea is essentially Goulds idea in his 1996 book: he sees Evolution just as an asymmetric diffusion of random variability. We just made this point explicit and developed some computations as a consequences of the analogy with the determination in Quantum Mechanics and the operatorial approach of Schrdinger.

4 Anti-entropy as a Measure of Symmetry Changes

In [22], we proposed to understand biological phenomena, in comparison and contrast with physical theories, as a situation where the theoretical symmetries are "constantly" broken. We will now show that such considerations allows us to interpret anti-entropy, somewhat in the spirit of Boltzmann's approach of physical entropy. In [4], premises of these aspects are considered from a strictly combinatorial point of view, leading to a "constructive" definition of the three components of anti-entropy, we recalled in section 3. To show how symmetries come into play we will analyze now these components.

Combinatorial complexity, K_c: For a total number of cells N and for a number n_j of cells of cell type j, the combinatorial complexity is defined as:

$$K_c = \log\left(\frac{N!}{\prod_j n_j!}\right) \tag{9}$$

A classical combinatorial point of view consists in saying that it is the number of ways to classify N cells in j categories each of sizes n_j. More precisely, we recognize, inside the logarithm, the cardinal, $N!$, of the symmetry group S_N, that is the group of transformations, called permutations, that exchange the labels of N elements. Similarly, $n_j!$ is the number of permutations among n_j units, which has the biological meaning of permutations of cells within a cell type: in other words, permuting cells *within the same cell type* is a

combinatorial invariant of the complexity of an organism. Thus, the group of permutations leaving the cell types invariants is the group $G_{type} = \prod S_{n_j}$, that is the group obtained as direct product of the symmetries corresponding to permutations within each cell type. Formally, this group corresponds to the change of labels in each cell type, which can all be performed independently and conserve the cl! assification by cell types. The cardinal of this group is $\prod_j n_j!$.

Then, the number of cell type configurations is the number of orbits generated by the right action of G_{type} on S_N. In other words, a cell type configuration is first given by a permutation of $[\![1, N]\!]$, which gives the random determination for N cells. Moreover, these transformations must be computed modulo any transformation of G_{type} that gives the same configuration (as we said, cells within each cell type are combinatorially equivalent — we will discuss below this hypothesis, in more biological terms). Lagrange theorem then gives the number of remaining transformations $N!/\prod_j n_j!$, which is the number of possible configurations. Clearly, an organism with just one cell type (typically, a unicellular being) has combinatorial complexity 0. As a result, this measure of combinatorial complexity depends on the total number N of cells, but is actually *a measure of the symmetry breaking induced by the differentiation in cell types.*

Let's compare the situation with Boltzmann approach of entropy. If one has a number of microscopic phase space states Ω having the same energy, the corresponding entropy is defined as $S = k_b \log(\Omega)$. In the case of gases, one considers that the particles are indiscernible. This means that one does not count twice situations which differ only by permuting particles. In other words one formally understands the situation by saying that labels attached to particles are arbitrary. Thus, more soundly, S is defined by $S = k_b \log(\Omega) - k_b \log(N!) > 0$. This symmetry by permutation reduces the size of the microscopic possibility space, and, as a result, entropy.

In our approach, we have $K_c = \log(N!) - \sum_i \log(n_i!)$ which is greater than 0, as soon as there is more than one cell type. Thus, the increase of the possibility space (the diversity or the differentiations) increases the complexity. More precisely, the complexity, as absolute value of anti-entropy, is decreased by the remaining symmetries, quantified by the term $\sum_i \log(n_i!)$. We understand then that anti-entropy can be analyzed, at least in this case, as an account of how much biological symmetries are broken by the cascade of differentiations. Formally, we can sum the situation up by saying that the combinatorial complexity and its contribution to anti-entropy are based on a group of transformations, S_N, and a subgroup, G_{type}. The biologically relevant quantity is then the ratio of sizes of the groups S_N and G_{type}.

Morphological complexity, K_m: This complexity is associated to the geometrical description of biologically relevant shapes. It is computed in particular by counting the number of connex areas. Note that this number corresponds to *space symmetry breakings* for motions covering this space — or ergodic motions. Then, one has to consider the number of shape

singularities, in the mathematical sense, where singularities are invariants by action of diffeomorphisms. The fractal-like structures are particularly relevant since they correspond to an exponential increase of the number of geometrical singularities with the range of scales involved. Thus, fractal-like structures lead to a linear growth of anti-entropy with the order of magnitudes where fractality is observed.

Functional complexity, K_f (the last quantity proposed in [4]): This quantity is given by the number of possible graphs of interaction. As a result, the corresponding component of anti-entropy is given by the choice of one graph structure (with distinguished nodes) among the possible graphs. This involves the selection of the structure of possible graphs and, correspondingly, which resulting graphs are considered equivalent. In terms of symmetries, we first have a symmetry among the possible graphs which is reduced to a smaller symmetry, by the equivalence relation. For example, in [4], the case is considered where the number of edges is fixed, so the considered symmetry group is engendered by the transformations which combine the deletion of an edge and the creation of another one. The orbits preserve the total number of edges, so that the orbit of a graph with $\langle k \rangle N$ edges are the graphs with this number of edg! es.

We understand then that anti-entropy, or at least its proposed decomposition in [4], is strictly correlated to the amount of symmetry changes. We will now look more closely at the case of combinatorial complexity since it involves only the groups of permutations and their subgroups, but at the same time will also allow us to express a crucial conceptual and mathematical point.

We indeed encounter a paradox in the case of combinatorial complexity. On one side, we have an assumption that cells of the same cell type are symmetric (interchangeable). On the other, in section 1, we stressed that each cell division consists in a symmetry change. This apparent paradox depends on the scale we use to analyze the problem, as well as on the "plasticity" of the cells in a tissue or organ, as the possibility to be interchanged and/or to modify their individual organization. Typically, one can assume that liver cells function statistically (what matters is their average contribution to the function of the organ), while neurons may have strong specific activities, yet they may also deeply modify their structure (change number, forms and functionality of synaptic connections, for example). Thus, we will next consider the individual contribution of cells to the combinatorial complexity of an organism at different scales.

If we consider an organism with a large number of cells, N, and the proportion q_j for cell type j we get two different quantities for the combinatorial complexities, K_{c1} and K_{c2}:

$$\frac{K_{c1}}{N} = \frac{\log(N!)}{N} \simeq \log(N) \qquad \frac{K_{c2}}{N} = \frac{\log\left(\frac{N!}{\prod_j (q_j N)!}\right)}{N} \simeq \sum_j q_j \log(1/q_j) \quad (10)$$

We propose to understand the situation as follows. Basically, both levels of cellular individuation are valid; but they have to be arranged in the right order.

Cellular differentiation is the first and main aspect of the ability of cells to individuate in a metazoan, so we can assume that the main determinant of combinatorial complexity is K_{c2}. It is only after this contribution that the further process of cellular individuation occurs. The latter leads to a mean contribution which is $\sum_j a_j \left(q_j \log(q_j N) - 1\right)$ per cell, where a_j quantifies the ability of each cell type to change their organization. It seems reasonable to expect that the a_j are high in the cases, for example, of neurons or of cells of the immune system. On the contrary, the a_j should be especially low for red blood cells. The reason for this is not only their lack of DNA, but also their relatively simple and homogeneous cytoplasmic organization. Similarly, l! iver cells may have statistically irrelevant changes in their individual structure.

Thus, the contribution of cell types to anti-entropy derives first from the formation of new cell types, while considering the ability of cells to reproduce, with changes, within a cell type as a further important (numerically dominant) aspect of their individuation process. Note that this analysis does not suppose that a cell type for a cell is irreversibly determined, but it means that the contribution of cell type changes to anti-entropy are understood as changes of K_{c2}.

We can then provide a refined version of S_c^-, where a_{ct} is the "weight" accorded to the formation of different cell types:

$$\frac{S_c^-}{-Nk_b} = a_{ct} \sum_j q_j \log(1/q_j) + \sum_j a_j \left(q_j \log(q_j N) - 1\right) \tag{11}$$

$$= \left(a_{ct} - \langle a_j \rangle\right) \langle \log(1/q_j) \rangle + \langle \left(\langle a_j \rangle - a_j\right) \log(1/q_j) \rangle + \langle a_j \rangle \log(N) \tag{12}$$

where $\langle x \rangle$ is the mean of x among all cells (so that the contribution of each cell type is proportional to its proportion in the organism). Both equations 11 and 12 are biologically meaningful. The terms in equation 11 correspond, by order of appearance, to the contribution of the categorization by cell types and to the contribution of individuation among a cell type. In equation 12, we have obtained terms that can be assimilated to K_{c1} (last term) and to K_{c2} (first term), the latter being positive only if $a_{ct} - \langle a_j \rangle > 0$, meaning that the contribution associated to cell types is positive only if it is greater than the mean cellular individuation. This is logical since cell types make a positive contribution to the complexity only if the amount of cellular diversity they introduce is greater than the one that cellular individuation alone would introduce.

Last but not least, the second term has the sign of an anti-correlation between a_j and $\log(1/q_j)$, meaning that this term is positive when there are many low complexity cell types[7] and few high complexity cell types. More precisely, using the Cauchy-Schwartz equality case, we get that maximizing (and minimizing)

[7] In theory of information, $\log(1/q_j)$ is the information associated to j: it quantifies its scarcity. If one assume that $a_j = \langle a_j \rangle \pm a$ and that we keep the mean complexity of cells, the anti-correlation is typically obtained when we have more low complexity cell types, with fewer cells, than high complexity cell types (which have therefore more cells). If one consider again the a_j as a degree of freedom, the same result can be achieved high complexity cell types with very high complexity and therefore a high number of bellow average complexity cell types.

this term (everything else being kept constant), leads to $\langle a_!j\rangle - a_j \propto \log(1/q_j) - \langle\log(1/q_j)\rangle$. Then this optimization *a priori* leads to maximizing the *variance* of information (in informational terms), at constant entropy (=mean information).

Here, the issue derived from looking with an increasing finer resolution at the individuation potential. However, the reciprocal situation can also occur. Let's consider the functional complexity, understood as the possibility of interactions between cells (the paradigmatic example is neurons). Then, by assuming that there are N neurons with $\langle k\rangle$ average number of synapses for each neuron (where $\langle k\rangle$ is between 10^3 and 10^4 for humans), as presented in [4], we get:

$$N_G = \binom{\binom{N}{2}}{\langle k\rangle N} \qquad \frac{K_{f1}}{N} \simeq \langle k\rangle \log(N) \tag{13}$$

However, if we postulate that *any* graph of interaction is possible, then we get a total number of possible interactions which corresponds to a choice between interaction or no interaction for each entry of the interaction matrix (N^2 cells). However, the latter is symmetric; and we do not count the self-interactions (because they correspond to the complexity of the cell) so we obtain $N(N-1)/2$ binary choices, so $2^{n(n-1)/2}$ possibilities: $K_{f2}/N \simeq N/2$.

There is two main lines of reasoning we can follow to understand the situation. The first is to look at the time structure of symmetry changes. Indeed, the symmetry changes occur as a temporal cascade. As a result, the temporal hierarchy of individuation is crucial. Here, we can refer to some phenomena concerning the graph of interaction of neurons. A crude description of the formation of neural networks is the following. First, a large number of "disordered" connections take place. Only after, the functional organization really increases by the decay of unused synapses (see for example [23]. Then, the "bigger" symmetry group involved in the description is of the form K_{f1}, with $\langle k\rangle$ mean number of connections; but then this symmetry group is reduced to obtain a smaller symmetry group with $\langle l\rangle$ mean number of connections. This operation can be seen as a change of symmetry groups, from the transformations preserving the number of connections with $\langle k\rangle N$ connections to those preserving $\langle l\rangle N$ connections.

Of course there are many other possible components for a measure of biological complexity. This proposal, defined as anti-entropy, provides just a tentative backbone for transforming the informal notion of "biological organizational complexity" into a mathematical observable, that is into a real valued function defined over a biological phenomenon. It should be clear that, once enriched well beyond the definition and the further details given in [4], this is a proper (and fundamental) biological observable. It radically differs from the rarely quantified, largely informal, always discrete (informally understood as a map from topologically trivial structures to integer numbers) notion of "information", still dominating in molecular circles, see [21] for a critique of this latter notion.

5 Theoretical Consequences of This Interpretation

In the section above, we have been focused on technical aspects of the "microscopic" definition of anti-entropy. Using this method, we have seen that

anti-entropy can mainly be understood in terms of symmetry changes. We will now consider the theoretical meaning of this situation in a more general way. As we exposed in [22], we propose to understand biological systems as characterized by a cascade of symmetry changes. Now, our understanding of a "biological trajectory", a philogenetic and ontogenetic path, as a cascade of symmetry changes yields a proper form of randomness to be associated to the construction and maintenance of biological organization. This perspective is particularly relevant for us, since it links the two theoretical approaches of the living state of matter that our team has introduced: anti-entropy [4] and extended criticality [3,22].

More precisely, in phylogenesis, the randomness is associated to the "choice" of different organizational forms, which occurs even when the biological objects are confronted with remarkably similar physical environment and physiological constraints. For example, the lungs of birds and mammals have the same function in similar conditions; but they have phylogenetic histories which diverged long ago and, extremely different structures.

This example is particularly prone to lead to approximate common symmetries, since it relates to a vital function (respiration and therefore gas exchanges) shared by a wide class of organisms. It is noteworthy that numerous theoretical studies have analyzed lungs by optimality principles [14,34,9]. However, the optimality principles differ in these studies (minimum entropy production, maximum energetic efficiency, maximum surface/volume ratio, ...). Accordingly, even among mammals, structural variability remains high. For example, [27] describe the differences in the geometrical scaling properties of human lungs on one side, and of rats, dogs and hamsters lungs on the other side. Moreover, [25] show that the criteria of energetic optimality and of robustness for the gas exchanges, with respect to geometric variations, are incompatible. More generally, optimization criteria are not particularly th! eoretically stable. In particular robustness is a relative notion: it depends on the property considered as well as on the transformations with respect to which we expect it to be robust [17].

Similarly, the theoretical symmetries constituted in ontogenesis are the result of the interactions with the environment and of the developmental trajectory already followed at a given time. In our perspective, this trajectory must then be understood as a history of symmetry changes. And, of course, the situation at a given moment does not "determine" the symmetry changes that the object will undergo. This is a crucial component of the randomness of the biological dynamics, as we consider that random events are associated to symmetry changes. These events are given by the interplay of the organism with its own physiology (and internal milieu) and with its environment, the latter being partially co-constituted by the theoretical symmetries of the organism, since the relevant aspects of the environment depend also on the organism.

In other terms, the conservation, in biology, is not associated to the biological *proper observables*, the phenotype, and the same (physical) interface (e.g. energy exchange) with the environment may yield very different phenotypes; thus, there is no need to preserve a specific phenotype. In short, the symmetry changes occurring in an organism can only be analyzed in terms of the previous

theoretical symmetries (biology is, first, an historical science) and the differences of the possible changes can be associated to different forms of randomness.

In the cases of symmetry breakings, the symmetry change corresponds to the passage to a subgroup of the original symmetry group. As a result, the theoretical possibilities are predefinable (as the set of subgroups of the original group). This typically occurs in the case of physical phase transitions, and the result is then a randomness associated to the choice of how the symmetry gets broken. For example, if an organism has an approximate rotational symmetry, this symmetry can be broken in a subgroup, for example by providing a particular oriented direction. We then have a rotational symmetry along an axis. This can again be broken, for example into a discrete subgroup of order 5 (starfish).

Another situation corresponds to the case where the symmetry changes are constituted on the basis of already determined theoretical symmetries (which can be altered in the process). This can be analyzed as the formation of additional observables which are attached to or the result of already existing ones. Then these symmetry changes are associated with already determined properties, but their specific form is nevertheless not predetermined. A typical example is the case of physically non-generic behaviours that can be found in the physical analysis of some biological situations, see [18]. From the point of view of the theoretical structure of determination, it is then a situation where there are predetermined attachment points, prone to lead the biological system to develop its further organization on them. The form of the biological response to these organizational opportunities of complexification is not, however, predetermined and then generates an ! original form of randomness. This theoretical account is close to the notion of next adjacent niche, proposed in [15]; however, we emphasize, here, that the theoretical determination of these next organizational possibilities is only partially predetermined. For example imagine that a biological dynamic has approximately certain symmetries, which leads to a non-generic singular point; then it is possible (and maybe probable) that this point will be stabilized in evolution, in an unknown way.

The former case is constituted, in a sense, by a *specific* organizational opportunity. We can, however, consider cases where such opportunities are not theoretical predetermined. Now, the constitution of symmetry changes should be understood as even more random, and the associated predictability is extremely low. Gould's most quoted example of "exaptation', the formation of the bones of the internal hear from the double jaw of some tetrapods, some two hundred million years ago can fit in this category.

We have seen that the symmetry changes lead to a strong form of randomness. This randomness and its iterative accumulation are, however, the very fabric of biological organization. Therefore, we have a theoretical situation where order (biological organization) is a direct consequence of randomness. Its global analysis allowed us to give mathematical sense to Gould's evolutionary complexification along evolution, as a consequence of the random paths of a asymmetric diffusion (sections 2 and 3). A finer (or local) analysis suggested a way to understand also ontogenetic changes in these terms, that is as a random dynamics

of symmetry changes. This situation should be not confused with the cases of order by fluctuations or statistical stabilization (for example, by the central limit theorem). In our case, indeed, the order is not the result of a statistical regularization of random dynamics into a stable form, which would transform them into a deterministic frame. On the contrary, the random path of a cascade of symmetry changes yields the theoretical symmetries of the object (its specific phenotypes), which also determine its behaviour.

In this context, the irreversibility of these random processes is taken into account by entropy production. The latter, or more precisely a part of the latter, is then associated to the ability of biological objects to generate variability, thus adaptability. In ontogenesis, this point confirms our analysis of the contribution of anti-entropy regeneration to entropy production, in association with variability, including cellular differentiation. This situation is also consistent with our analysis of anti-entropy as a measure of symmetry changes. Notice that the symmetry changes, considered as relevant with respect to anti-entropy, may be taken into account, for example, in the coefficients corresponding to the individuation capacity of different cell types in our discussion above (see section 4).

References

1. Aoki, I.: Entropy production in human life span: A thermodynamical measure for aging. AGE 17, 29–31 (1994)
2. Bailly, F., Longo, G.: Randomness and determinism in the interplay between the continuum and the discrete. Mathematical Structures in Computer Science 17(02), 289–305 (2007)
3. Bailly, F., Longo, G.: Extended critical situations: the physical singularity of life phenomena. Journal of Biological Systems 16(2), 309 (2008)
4. Bailly, F., Longo, G.: Biological organization and anti-entropy. Journal of Biological Systems 17(1), 63–96 (2009)
5. Bailly, F., Longo, G.: Mathematics and the natural sciences; The Physical Singularity of Life. Imperial College Press, World Scientific (2011)
6. d'Alessio, P.: Aging and the endothelium. Experimental Gerontology 39(2), 165–171 (2004)
7. Demetrius, L.: Caloric restriction, metabolic rate, and entropy. The Journals of Gerontology Series A: Biological Sciences and Medical Sciences 59(9), B902–B915 (2004)
8. Duffin, R.J., Zener, C.: Geometric programming, chemical equilibrium, and the anti-entropy function. Proceedings of the National Academy of Sciences 63(3), 629 (1969)
9. Gheorghiu, S., Kjelstrup, S., Pfeifer, P., Coppens, M.-O.: Is the lung an optimal gas exchanger? In: Losa, G.A., Merlini, D., Nonnenmacher, T.F., Weibel, E.R. (eds.) Fractals in Biology and Medicine. Mathematics and Biosciences in Interaction, pp. 31–42. Birkhuser, Basel (2005)
10. Gould, S.J.: Wonderful life. Norton (1989)
11. Gould, S.J.: Full house: The spread of excellence from Plato to Darwin. Three Rivers Pr. (1997)
12. Hardy, J.D.: The radiation of heat from the human body iii. Journal of Clinical Investigation 13(4), 615–620 (1934)

13. Hayflick, L.: Entropy explains aging, genetic determinism explains longevity, and undefined terminology explains misunderstanding both. PLoS Genet. 3(12), e220 (2007)
14. Horsfield, K.: Morphology of branching trees related to entropy. Respiration Physiology 29(2), 179 (1977)
15. Kauffman, S.A.: Investigations. Oxford University Press, USA (2002)
16. Lambert, F.L.: Entropy and the second law of thermodynamics (2007), http://www.entropysite.com
17. Lesne, A.: Robustness: Confronting lessons from physics and biology. Biol. Rev. Camb. Philos. Soc. 83(4), 509–532 (2008)
18. Lesne, A., Victor, J.-M.: Chromatin fiber functional organization: Some plausible models. Eur. Phys. J. E. Soft. Matter 19(3), 279–290 (2006)
19. Lindner, A.B., Madden, R., Demarez, A., Stewart, E.J., Taddei, F.: Asymmetric segregation of protein aggregates is associated with cellular aging and rejuvenation. Proceedings of the National Academy of Sciences 105(8), 3076–3081 (2008)
20. Longo, G.: Critique of Computational Reason in the Natural Sciences. Imperial College Press/World Scientific (2009)
21. Longo, G., Miquel, P.-A., Sonnenschein, C., Soto, A.M.: From information to organization in biology (to appear, 2012)
22. Longo, G., Montvil, M.: From physics to biology by extending criticality and symmetry breakings. Progress in Biophysics and Molecular Biology 106(2), 340–347 (2011); Systems Biology and Cancer
23. Luo, L., O'Leary, D.M.: Axon retraction and degeneration in development and disease. Annual Review of Neuroscience 28, 127–156 (2005)
24. Marineo, G., Marotta, F.: Biophysics of aging and therapeutic interventions by entropy-variation systems. Biogerontology 6, 77–79 (2005)
25. Mauroy, B., Filoche, M., Weibel, E.R., Sapoval, B.: An optimal bronchial tree be dangerous. Nature 427, 633–636 (2004)
26. McShea, D.W., Brandon, R.N.: Biology's first law: the tendency for diversity and complexity to increase in evolutionary systems. University of Chicago Press (2010)
27. Nelson, T.R., West, B.J., Goldberger, A.L.: The fractal lung: Universal and species-related scaling patterns. Cellular and Molecular Life Sciences 46, 251–254 (1990)
28. Nystrm, T.: A bacterial kind of aging. PLoS Genet. 3(12), e224 (2007)
29. Olshansky, S., Rattan, S.: At the heart of aging: Is it metabolic rate or stability? Biogerontology 6, 291–295 (2005)
30. Pezard, J., Martinerie, L., Varela, F.J., Bouchet, F., Guez, D., Derouesn, C., Renault, B.: Entropy maps characterize drug effects on brain dynamics in alzheimer's disease. Neuroscience Letters 253(1), 5–8 (1998)
31. Ruud, J.T.: Vertebrates without erythrocytes and blood pigment. Nature 173(4410), 848 (1954)
32. Schrdinger, E.: What Is Life? Cambridge U.P. (1944)
33. Sohal, R.S., Weindruch, R.: Oxidative stress, caloric restriction, and aging. Science 273(5271), 59–63 (1996)
34. West, G.B., Brown, J.H., Enquist, B.J.: The fourth dimension of life: Fractal geometry and allometric scaling of organisms. Science 284(5420), 1677–1679 (1999)

Haunted Quantum Contextuality versus Value Indefiniteness

Karl Svozil

Institute of Theoretical Physics, Vienna University of Technology, Wiedner Hauptstraße 8-10/136, A-1040 Vienna, Austria
svozil@tuwien.ac.at

Abstract. Physical entities are ultimately (re)constructed from elementary yes/no events, in particular clicks in detectors or measurement devices recording quanta. Recently, the interpretation of certain such clicks has given rise to unfounded claims which are neither necessary nor sufficient, although they are presented in that way. In particular, clicks can neither inductively support nor "(dis)prove" the Kochen-Specker theorem, which is a formal result that has a deductive proof by contradiction. More importantly, the alleged empirical evidence of quantum contextuality, which is "inferred" from violations of bounds of classical probabilities by quantum correlations, is based on highly nontrivial assumptions, in particular on physical omniscience.

Discussion

Time and again, in coffee houses and elsewhere, members of the Viennese experimental physics community reminded me always to keep in mind that all our physical "facts" are ultimately derived and constructed from detector clicks. It is this basic wisdom that, when consequentially applied to recent experiments, suggests to rethink certain claims of empirical proof.

Let us, for the sake of properly assessing the situation, review some historical cornerstones. Motivated by certain, as it turned out inapplicable, no-go theorems by von Neumann regarding hidden parameters, Bell came forward with criteria for classical probabilities and expectations, resembling the *conditions of possible experience* that had been contemplated by Boole a century earlier [18]. Essentially, these criteria state that, if one forces the (counterfactual) physical co-existence upon certain finite sets of complementary, incompatible, potential observables—meaning that every single one could be measured, although due to complementarity it is impossible to simultaneously measure all of them—the associated potential measurement outcomes are subject to certain algebraic bounds.

As these probabilistic bounds are not satisfied by quantum observables, the respective measurements outcomes cannot consistently co-exist [16]; at least not under the classical presumptions entering the calculations leading to these bounds. These arguments have subsequently been strengthened by the Kochen-Specker and the Greenberger-Horne-Zeilinger theorems, as for the latter ones

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 309–314, 2012.

any violations of the conditions of possible experience must occur on every single quantum and at least for a single observable [24] rather than occasionally.

Those results relate to situations in which *omniscience* is assumed; that is, all observables which could potentially be observed can indeed be associated with actual elements of physical reality of a single quantum. For a realist this might appear self-evident [20]. Also for experimentalists this seems to be obvious; after all, any particular observation renders outcomes, regardless of the mutual complementarity of some of the observables involved; in this view, "potentially operational" means "existence." By this inkling, the situation suggests that the measurement "reveals" a pre-existing element of physical reality of the quantum observed. Stated pointedly, registration of some detector clicks is interpreted as a revelation about what is taken as the quantized object.

If these pre-existing elements of physical reality are taken for granted, it is not unreasonable to "solve" or "explain" the conundrum imposed by the various aforementioned theorems by assuming that any potential measurement outcome may depend on whatever other maximal co-measurable collection of observables (the context, interpretable as maximal operator [11, sect. 84]) are co-measured alongside. This dependence of the outcome of a single quantum measurement on its context—that is, the influence of what is (sometimes implicitly) co-measured alongside this single quantum measurement—is termed *quantum contextuality*.

Note that the Born rule, and also Gleason's theorem, requires the quantum probabilities and expectations, and thus all quantum statistical properties, to be *noncontextual*. Notice also that contextuality attempts to maintain a realistic, omniscient, quasi-classical framework by abandoning context independence for single quantum observables.

Now, if one maintains realistic omniscience—that is, the pre-existence of all outcomes of complementary potential observables (as is implicitly assumed in Bell- and Kochen-Specker-type arguments)—then it is indeed true that, as stated by Cabello [5], "the immense majority of the experimental violations of Bell inequalities [[proves]] quantum contextuality." Actually, the only difference between older evidence of violations of Bell-type inequalities and more recent ones ([12], [2], [13], [1] and [14]) seems to be based on the fact that the prior ones rely on spatially separated quanta in Einstein-Podolsky-Rosen "explosion" type schemes, whereas more recent ones are based on single quanta—a concept which appears to be more in the spirit of Kochen-Specker type theorems which apply to the structure of observables of single quanta [7]. But even these sorts of empirical findings referring to single quanta rely on the non-instantaneous measurement of all but a few (mostly two or three in cases involving two- or three-particle Einstein-Podolsky-Rosen and Greenberger-Horne-Zeilinger type) configurations, and therefore cannot even counterfactually assure the operational existence of all elements of physical reality at once [22].

Alas, these assumptions are neither necessary (and sufficient), as other, rather exotic options [15, 17] demonstrate, nor is there any more direct empirical evidence in their support. Indeed, quantum predictions of Einstein-Podolsky-Rosen

type setups involving singlet states of qutrits suggest that contextuality cannot be observed [23], although a direct experimental test is still lacking.

Thus with regards to quantum contextuality, that is, the "explanation" of Bell- and Kochen-Specker-type arguments, the situation is rather discomforting: insofar as contextuality seems to "explain" various findings related to quantum predictions and correlations, it can only be indirectly inferred by assuming some extra assumptions, including classical omniscience; otherwise it is not necessary. And insofar it could be directly testable it is very unlikely to show up. Because of this dilemma, it is suggested to re-evaluate recent empirical findings in terms of a much broader picture of *value indefiniteness;* including also the possibility that there needs not exist a pre-existing element of physical reality associated with certain observables.

Stated pointedly, value indefiniteness is the assumption that, with regards to certain potential observables, a quantum system cannot prepared to be in a specific, definite state, because the quantum system has been prepared in a definite state of a different, complementary observable. Hence there does not exist any entity or property of a physical system under observation which determines a measurement outcome of such a value indefinite observable completely. If some observer chooses to measure any such value indefinite observable—thus "forcing" an observation upon the *combined* system of measurement apparatus and quantum—the actual measurement outcome or event is also (if not entirely) determined by the disposition of the measurement apparatus [3]. This should be contrasted to the definition of an element of physical reality in the sense of Einstein, Podolsky, and Rosen [10]: in the latter case the measurement outcome is defined or linked to a physical property of the quantum measured, rather than to the combination of both measurement apparatus and the quantum measured.

Thus in situations involving counterfactual potential observables, such as in Bell- and Kochen-Specker-type arguments, the experimental outcomes actually measured might not originate from such pre-existence, but might depend on the interaction between the quantum measured and the measurement apparatus. Pointedly stated, the outcome might not reflect an intrinsic objective physical property of the quantized object, but rather originate in the way a measurement apparatus generates the outcome by interacting with the quantum. Already Bell [3] suggested that (cf. also Refs. [6, 8] for related experiments) "the result of an observation may reasonably depend ... on the complete disposition of the apparatus." Perhaps this was also what Bohr had in mind by mentioning [4] "the impossibility of any sharp separation between the behavior of atomic objects and the interaction with the measuring instruments which serve to define the conditions under which the phenomena appear."

So far no experiments have been performed to quantify the different empirical consequences of the assumption of quantum contextuality versus the assumption of quantum value indefiniteness. One possibility would be to measure the varying capacities of the measurement apparatus to translate between the context observed and a different context in which a quantum was prepared [21].

These considerations are highly relevant for the computational capacities of quantized system exhibiting incomputability [9], because, as it is commonly assumed [25], quantum systems are irreducibly indeterministic. How can we conceptualize and justify such computational capacities, in particular in view of the uniform one-to-oneness of the quantum evolution at certain devices such as fifty-fifty beam splitters generating a coherent superposition of classical states [19]? One possibility would take into account the combined action of a single quantum system, registered by a macroscopic measurement device with many degrees of freedom.

Acknowledgements. This research was partly supported by Seventh Framework Program for research and technological development (FP7), PIRSES-2010-269151-RANPHYS.

References

[1] Amselem, E., Rådmark, M., Bourennane, M., Cabello, A.: State-independent quantum contextuality with single photons. Physical Review Letters 103(16), 160405 (2009), http://dx.doi.org/10.1103/PhysRevLett.103.160405, doi:10.1103/PhysRevLett.103.160405

[2] Bartosik, H., Klepp, J., Schmitzer, C., Sponar, S., Cabello, A., Rauch, H., Hasegawa, Y.: Experimental test of quantum contextuality in neutron interferometry. Physical Review Letters 103(4), 040403 (2009), http://dx.doi.org/10.1103/PhysRevLett.103.040403, doi:10.1103/PhysRevLett.103.040403

[3] Bell, J.S.: On the problem of hidden variables in quantum mechanics. Reviews of Modern Physics 38, 447–452 (1966), http://dx.doi.org/10.1103/RevModPhys.38.447, doi:10.1103/RevModPhys.38.447

[4] Bohr, N.: Discussion with Einstein on epistemological problems in atomic physics. In: Schilpp, P.A. (ed.) Albert Einstein: Philosopher-Scientist, pp. 200–241. The Library of Living Philosophers, Evanston (1949), http://dx.doi.org/10.1016/S1876-05030870379-7, doi:10.1016/S1876-05030870379-7

[5] Cabello, A.: Experimentally testable state-independent quantum contextuality. Physical Review Letters 101(21), 210401 (2008), http://dx.doi.org/10.1103/PhysRevLett.101.210401, doi:10.1103/PhysRevLett.101.210401

[6] Cabello, A.: Proposal for revealing quantum nonlocality via local contextuality. Physical Review Letters 104, 220401 (2010), http://dx.doi.org/10.1103/PhysRevLett.104.220401, doi:10.1103/PhysRevLett.104.220401

[7] Cabello, A.: Quantum physics: Correlations without parts. Nature 474(7352), 456–458 (2011), http://dx.doi.org/10.1038/474456a, doi:10.1038/474456a

[8] Cabello, A., Cunha, M.T.: Proposal of a two-qutrit contextuality test free of the finite precision and compatibility loopholes. Physical Review Letters 106, 190401 (2011), http://dx.doi.org/10.1103/PhysRevLett.106.190401, doi:10.1103/PhysRevLett.106.190401

[9] Calude, C.S., Svozil, K.: Quantum randomness and value indefiniteness. Advanced Science Letters 1(2), 165–168 (2008), http://www.ingentaconnect.com/content/asp/asl/2008/00000001/00000002/art00004, doi:10.1166/asl.2008.016

[10] Einstein, A., Podolsky, B., Rosen, N.: Can quantum-mechanical description of physical reality be considered complete? Physical Review 47(10), 777–780 (1935), http://dx.doi.org/10.1103/PhysRev.47.777, doi:10.1103/PhysRev.47.777

[11] Halmos, P.R.: Finite-dimensional Vector Spaces. Springer, New York (1974)

[12] Hasegawa, Y., Loidl, R., Badurek, G., Baron, M., Rauch, H.: Quantum contextuality in a single-neutron optical experiment. Physical Review Letters 97(23), 230401 (2006), http://dx.doi.org/10.1103/PhysRevLett.97.230401, doi:10.1103/PhysRevLett.97.230401

[13] Kirchmair, G., Zähringer, F., Gerritsma, R., Kleinmann, M., Gühne, O., Cabello, A., Blatt, R., Roos, C.F.: State-independent experimental test of quantum contextuality. Nature 460, 494–497 (2009), http://dx.doi.org/10.1038/nature08172, doi:10.1038/nature08172

[14] Lapkiewicz, R., Li, P., Schaeff, C., Langford, N.K., Ramelow, S., Wieśniak, M., Zeilinger, A.: Experimental non-classicality of an indivisible quantum system. Nature 474, 490–493 (2011), http://dx.doi.org/10.1038/nature10119, doi:10.1038/nature10119

[15] Meyer, D.A.: Finite precision measurement nullifies the Kochen-Specker theorem. Physical Review Letters 83(19), 3751–3754 (1999), http://dx.doi.org/10.1103/PhysRevLett.83.3751, doi:10.1103/PhysRevLett.83.3751

[16] Peres, A.: Unperformed experiments have no results. American Journal of Physics 46, 745–747 (1978), http://dx.doi.org/10.1119/1.11393, doi:10.1119/1.11393

[17] Pitowsky, I.: Resolution of the Einstein-Podolsky-Rosen and Bell paradoxes. Physical Review Letters 48, 1299–1302 (1982), http://dx.doi.org/10.1103/PhysRevLett.48.1299, doi:10.1103/PhysRevLett.48.1299

[18] Pitowsky, I.: George Boole's 'conditions of possible experience' and the quantum puzzle. The British Journal for the Philosophy of Science 45, 95–125 (1994), http://dx.doi.org/10.1093/bjps/45.1.95, doi:10.1093/bjps/45.1.95

[19] Schrödinger, E.: The Interpretation of Quantum Mechanics. Dublin Seminars (1949-1955) and Other Unpublished Essays. Ox Bow Press, Woodbridge, Connecticut (1995)

[20] Stace, W.T.: The refutation of realism. In: Feigl, H., Sellars, W. (eds.) Readings in Philosophical Analysis, pp. 364–372. Appleton-Century-Crofts, New York (1949); previously published in Mind 53, 349–353 (1934)

[21] Svozil, K.: Quantum information via state partitions and the context translation principle. Journal of Modern Optics 51, 811–819 (2004), http://dx.doi.org/10.1080/09500340410001664179, doi:10.1080/09500340410001664179

[22] Svozil, K.: Are simultaneous Bell measurements possible? New Journal of Physics 8(39), 1–8 (2006), http://dx.doi.org/10.1088/1367-2630/8/3/039, doi:10.1088/1367-2630/8/3/039

[23] Svozil, K.: Proposed direct test of a certain type of noncontextuality in quantum mechanics. Physical Review A 80(4), 040102 (2009), http://dx.doi.org/10.1103/PhysRevA.80.040102, doi:10.1103/PhysRevA.80.040102

[24] Svozil, K.: Quantum value indefiniteness. Natural Computing online first, 1–12 (2010) ISSN 1567-7818, http://dx.doi.org/10.1007/s11047-010-9241-x, doi:10.1007/s11047-010-9241-x

[25] Zeilinger, A.: The message of the quantum. Nature 438, 743 (2005), http://dx.doi.org/10.1038/438743a, doi:10.1038/438743a

Is the Universe Like π or Ω?

Stephen Wolfram

Wolfram Research, Inc., USA

Can everything about the universe and its history be computed like the digits of π? Or is it instead uncomputable, like the digits of Chaitin's Ω?

We do not yet know the answer. But the question has great significance not only for science but also for our whole understanding of the nature of existence.

The notion that there might be an ultimate—in effect computable—model for the universe has a long history. Indeed, from the earliest days of Greek science until well into the 1900s, it seems to often have been believed that an ultimate model of the universe was not far away.

In antiquity, it was thought that perhaps everything was made of "elements" like fire and water. In the 1700s, after Newton and friends, the focus changed to "corpuscles," bound by gravity-like forces. In the 1800s, it was fields, with atoms perhaps being "knots in the ether." Then with the discovery of the electron, it was briefly thought that it might be what everything is made of. That then in turn gave way to electromagnetic fields, then gravitational fields, then some kind of extended "unified fields."

In the early 1900s, it was thought that all the "elementary particles" of the universe were known. But that turned out not to be true, and by the 1950s, new supposedly elementary particles were being discovered at a rapid rate.

In the 1960s, the quark model had explained most of these new particles, and by the 1970s work on quantum fields and gauge theories had led to what is now called the Standard Model of particle physics—which appeared to explain all forces except gravity.

Three more shocks occurred in quite rapid succession in the mid-1970s, however, with the discoveries of the c quark, τ lepton and b quark. But by the late 1970s, there was again widespread enthusiasm for an ultimate "grand unified" theory. But when the predicted phenomenon of proton decay was not discovered, enthusiasm again waned.

The Standard Model nevertheless seemed to be an adequate, if not particularly elegant, theory for what was known, except for gravity. It still did not explain the particular observed collection of fundamental particles. And there remained the nagging problem that gravity did not really fit into its formalism at all.

And for essentially 30 years, this is what the situation has been.

All sorts of elaborate—and elegant—mathematical structures have been constructed, notably in string theory. But despite various encouraging signs, none of them have convincingly been shown to reproduce the actual features of observed physical reality.

Looking at all this history, from the "elements" of antiquity to modern string theory, one might reasonably be pessimistic that an ultimate theory of physics

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 315–319, 2012.

would ever be found. For it seems that every level of description, upon more careful scrutiny, is found to have deficiencies, which eventually require a whole new level of description, with progressively more complex formalism.

And indeed, 30 years ago, I myself would have been skeptical that this process would ever end, and that any ultimate theory of the universe could ever finally be found.

But then something happened that forever changed my intuition. I started studying the computational universe of simple programs—and I found, to my great surprise, that even some of the very simplest possible programs can produce immense complexity.

The notion that more complex rules and more complex formalism are inevitably needed to explain apparently more complex phenomena is just not correct.

And by the 1990s I had formulated my Principle of Computational Equivalence: that the behavior of essentially any program that is not obviously simple corresponds to a computation of equivalent sophistication.

So that immediately makes one ask whether in fact all the complexity and richness of our universe can be captured by even a quite simple program.

In effect: can our physical universe be found out among the programs in the computational universe? And perhaps even among the simple such programs.

If the universe corresponds to a simple program, then there are some immediate conclusions that can be drawn. For a start, in a simple program there is no "room" to fit all the details of the observed universe—the dimensionality of space, the masses of particles, and so on. Rather, all of those details have to emerge from something much lower level—indeed lower level even than for example space and time.

One might think that there could be many programs that would almost agree in their predictions for the universe, and it would require elaborate new experiments to tell them apart. But among simple programs, it is almost inevitable that there have to be major changes from one program to the next, leading to dramatic differences in the universes they imply.

Of course, having a simple underlying program does not make it easy to deduce its consequences, or to compare them with known features of the universe. In fact, any program whose behavior is rich enough to be plausible as a model for our universe inevitably shows the phenomenon of computational irreducibility—which implies that to work out its consequences can in effect take explicitly following each step in its evolution.

As a practical matter, I have studied in some detail a particular class of models—equivalent to many others—in which the lowest level representation is a network of connected nodes. And rather remarkably, I have found that it is quite straightforward in such a model to derive an appropriate approximation of Special Relativity and standard Einsteinian gravitation—and there are indications that quantum mechanics will also emerge.

Will a model like this actually reproduce our whole universe? I do not know. But the indications so far are good enough that it would seem foolish not to find out for sure.

But even if such a model based on a simple program will work, how should one find the appropriate program? The tremendous tendency based on traditional physics thinking is to work backwards from known physical laws, to try to "reverse engineer" a program for the universe.

But if the program is simple, it seems inconceivable that this could work. It is just too far from the underlying rules to the observable physical phenomena.

However, if the program is simple, there is another—at first outlandish—approach to use: one can just start enumerating programs, and searching for the correct one.

In effect, one is searching the computational universe for our actual physical universe.

Traditional intuition might make one think this an absurd approach. But with the intuition one gets from studying the computational universe, it seems a lot less absurd.

And indeed I have continually been surprised just what can be found by searching the computational universe. The simplest universal Turing machine. The simplest axiom system for logic. And countless algorithms of great practical importance, notably in *Mathematica* and Wolfram|Alpha.

Now, of course, even if our universe can be represented by a program, why should it be a simple one?

It is a very basic observation about the universe that there is at least some order in it: every particle in the universe does not get to "do its own thing." But if the universe is a program, just how complicated a one might it be?

Perhaps a few lines of code. Perhaps a hundred. Or a million. Or still bigger.

I had thought that in searching for our universe the main issue would be scanning a large enough number of candidates. But in my practical efforts in this direction, this is not the problem.

Certainly there are lots of candidate universes that are plainly not our universe. They have no notion of time, no communication in space, or some other pathology. But even among the first thousand or so possibilities, one is already beginning to find programs whose behavior is complex enough that one cannot readily tell that they are not our universe.

Of course, there is still a great distance to go in determining whether any of them actually does correspond to our universe. And there are many difficulties—perhaps very great—associated with computational irreducibility.

But it is encouraging—and suggestive—that even among fairly simple programs, there are potentially good candidates.

I often wonder what it will be like if we actually do find that one of these simple programs can reproduce our universe. In a sense it will be a very anti-Copernican moment.

For ever since Copernicus, we have repeatedly been confronted with ways in which we are not special. Our planet is not at the center of the universe. Our chemistry is just like other chemistry. And so on.

But if our universe is a simple one in the space of all possible universes, then it would seem that in that way we are in fact special.

Perhaps that will be true. But I have a sneaking suspicion that something still more bizarre will be true. And that somehow almost any candidate universe—when viewed by observers inside that universe—will turn out somehow to be precisely equivalent, and equivalent to our universe.

So that in a sense the rules for our universe could be taken to be either simple or immensely complex—but for us inside the universe it will make no difference; there will be a precise equivalence.

If we are lucky, it could be only a fairly small number of years before we can tell if our universe can be reproduced by a simple program. But if in fact the universe cannot be represented in this way, my guess is that it will be a long time before this can reliably be known.

Over and over again I or others have thought that some phenomenon or another just could not realistically be produced by a simple program. And over and over this intuition has been wrong. And searching in the computational universe has discovered a program that perhaps works in some quite unexpected way, but that nevertheless produces the phenomenon one is looking for.

In the models I have studied in the greatest detail, one might for example imagine that quantum processes could never arise. For in some sense the models are deterministic, just successively replacing small parts of giant networks. But between the intrinsic randomness generated by the actual dynamics of the system, and the fact that one is dealing with networks, rather than explicit fixed-dimensional spaces, the standard signatures of underlying determinism disappear—clearly the way for quantum-like phenomena.

Usually in physics one imagines that one is just trying to make a model that somehow approximates a system. But if one is to find an ultimate model of physics, no approximation can be involved. The model, if run long enough, must reproduce in precise detail every aspect of the evolution of the physical universe, with no freedom whatsoever.

In effect, to find such a model would be to reduce physics to mathematics. To find a way to compute, from a known rule and known initial conditions, everything about how the universe evolves, just like one can compute the digits of π.

Just as in the digits of π, there would be lots of apparent randomness. But there would be no outside source of unknown, probabilistic, randomness. All randomness would arise intrinsically from the actual rules for the system.

So how might this be wrong?

In effect, a key assumption is that the underlying rules for the universe must be just like those in a standard computer, or Turing machine, or a cellular automaton. The details of the most obvious representation would surely be different (in my detailed models, everything is for example based on networks).

But the point is that there would be a precise correspondence to a standard computational universe.

When one looks at the current mathematical formalism of physics, one might be led to think that this could not possibly be correct. For physics is full of continuous numbers—for positions in space, sizes of quantum amplitudes, and so on.

And at least in the most obvious ways, no standard computational system can ever precisely represent such numbers.

My guess, though, is that the presence of such continuous quantities is an approximation, just as it is in standard large-scale physics. On a small scale there are discrete atoms, but on a large scale it is a good approximation, under many circumstances, to treat materials as continuous.

Likewise, there could be "true randomness," like in the non-computable constant Ω.

And until we can reproduce the universe from an explicit simple rule, we cannot exclude this possibility.

To me, it has the feel of something unnecessary. But ultimately we cannot be sure nothing like this is going on until we have successfully reproduced the universe without it.

I suppose in a sense it would be an anticlimax after all the development of science and physics to somehow come to the end: to finally be able to hold in our hands a complete representation of our whole universe.

And we might think that in doing this we would have exhausted the need for science.

But this would not be true. For the phenomenon of computational irreducibility in a sense guarantees an endless frontier: in this case, it guarantees that there are questions about our universe that are always arbitrarily hard to answer, and require arbitrarily long computations.

To find out that there is a simple computational rule for the universe—as there is for π—would however be a remarkable achievement for human intellect. For it would show us that our brains can successfully capture the underlying rules for our whole universe.

It might not be true. It might be that there is no final theory of our universe, and that, like Ω, there is a bottomless pit of surprises not just in the overall behavior of the universe, but within the theory of the universe itself.

But I think it is almost a responsibility for our times to find out if the universe is instead like π—and able to be captured in a finite way that our brains can comprehend. And I myself hope very much to be able to pursue this goal, and to see whether in fact all the remarkable richness and complexity of our universe can be reduced to something as simple as π.

Outerplanar Graphs and Delaunay Triangulations

Ashraful Alam[1,⋆], Igor Rivin[2], and Ileana Streinu[3,⋆⋆]

[1] Department of Computer Science, University of Massachusetts Amherst, MA
ashraful@cs.umass.edu
[2] Department of Mathematics, Temple University, Philadelphia, PA
rivin@math.temple.edu
[3] Department of Computer Science, Smith College, Northampton, MA
streinu@cs.smith.edu, istreinu@smith.edu

Abstract. Dillencourt [1] showed that all maximal outerplanar graphs can be realized as Delaunay triangulations of points in convex position. In this note, we give two new, alternate proofs.

1 Introduction

The Delaunay triangulation of a planar point set in convex position is, combinatorially, a maximal outerplanar graph. Dillencourt [1] showed that the other direction is also true: any graph which arises from a triangulation of the interior of a simple polygon can be realized as a Delaunay triangulation. Dillencourt's proof uses a simple and natural criterion on the angles of triangles in a Delaunay triangulation, and results in an $O(n^2)$ time incremental algorithm to calculate these angles and infer a realization. Lambert [5] adapted this method to a linear time algorithm.

The general question, of characterizing and reconstructing arbitrary Delaunay triangulations (in two- or higher dimensions), is substantially more difficult. A closely related problem, going back to Steiner (see Grünbaum [3], page 284), asks for a characterization of the graphs of inscribable or circumscribable polyhedra: those whose vertices lie on a sphere, resp. whose faces are tangent to a sphere. Such graphs are said to be of inscribable, resp. circumscribable type. The best result to date is due to Rivin [6], who proved necessary and sufficient conditions for a polyhedral graph to be of inscribable or circumscribable type, but these conditions have not led to intrinsic structural characterizations of the graphs. Dillencourt and Smith [2] linked inscribability of a graph to its realizability as a Delaunay triangulations, and gave a criterion relating Hamiltonicity to inscribability.

Our contribution. In this paper, we present two new simple and elementary proofs of Dillencourt's theorem:

⋆ Research supported by NSF CCF-1016988 grant of Streinu.
⋆⋆ Research supported by NSF CCF-1016988 and DARPA HR0011-09-0003 grants.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 320–329, 2012.

Theorem 1. *Any maximal outerplanar graph can be realized as a Delaunay triangulation.*

We are not aware of these proofs previously appearing in the literature. The first one is an easy consequence of the criterion of Dillencourt and Smith [2] relating Hamiltonianicity to inscribability. The second one, which occupies most of this note, uses Rivin's [6] inscribability criterion and constructs an explicit "witness" of this inscribability, in the form of weights assigned to the edges.

Preliminaries. We work with graphs $G = (V, E)$ with vertices $V = \{1, \ldots, n\}$, $n > 3$, and edges E, denoted as pairs of vertices $e = ij$, $i, j \in V$. Two paths between two vertices are *independent* if they do not share other vertices besides the end-points. A graph is connected if there is a path between any two vertices and it is *k-connected* if there are k independent paths between any two vertices. All graphs in this paper are 2-connected.

A graph is *planar* if it can be drawn in the plane (or, equivalently, on the sphere) with no crossings of endpoint-disjoint edges. A drawing of a planar graph on the sphere, called a *spherical graph*, subdivides the sphere into regions called faces. For 2-connected spherical graph drawings, the faces are topological disks. In the plane, exactly one face, called the *outer face*, is unbounded. A spherical graph is specified by its sets of vertices, edges and faces: $G = (V, E, F)$. A *plane graph* is obtained from a spherical graph by specifying a face f as the outer face: $G = (V, E, F, f)$.

The stellation G_s of a plane graph $G = (V, E, F, f)$ is the graph obtained by adding one new vertex (the *stellating* vertex s) and connecting it to all the vertices of f through edges called *stellating* edges. Stellation does not violate the planarity property: a stellated planar graph remains planar, and a plane realization of it is obtained by placing the stellating vertex inside the face f.

The dual $G^* = (V^*, E^*, F^*)$ of a spherical graph $G = (V, E, F)$ is obtained by switching the roles of vertices and faces: $V^* = F$, $E^* = E$, $F^* = V$.

A plane graph where *all* vertices lie on the outer face is called an *outerplanar graph*. In a *maximal outerplanar graph* all faces are triangular except the outer face. A *wheel graph* is obtained by stellating a cycle. We will consider later *stellated outerplanar graphs*.

A *cutset* of a graph is the minimal set of edges whose removal disconnects the graph. A cutset is *coterminous* if all the edges emanate from a single vertex and *noncoterminous* if its edges do not have a common endpoint. In the dual graph G^* of a spherical graph G, a coterminous cut set of G becomes the set of edges of a face in G^*; a noncoterminous cut set of G becomes a non-facial cycle of G^* (a cycle which is not a face).

A graph is *polyhedral* if it is planar and 3-connected. In this case, the faces of a spherical realization are uniquely determined. Any polyhedral graph can be realized as the 1-skeleton of a convex polyhedron in dimension 3 (Steinitz theorem, see Grünbaum [3]). A polyhedral graph is *inscribable* if it can be realized as the 1-skeleton of an convex polyhedron inscribed in a sphere.

Given a set P of points in the Euclidean plane, a *triangulation* of P is a plane graph where all faces, with the possible exception of the outer face, are triangles.

The *Delaunay triangulation* of a point set P is a triangulation of P with the property that the circumcircle of any face contains no other point of P inside. Not all planar triangulated graphs arise as graphs of Delaunay triangulations. We focus now on maximal outerplanar graphs.

2 The First Proof

Our first proof that an outerplanar graph can be realized as a Delaunay triangulation relies on two elegant results due to Dillencourt and Smith [2] and to Rivin [6,7]. They relate inscribability, realization as Delauney triangulation and Hamiltonicity.

A *Hamiltonian cycle* in a graph is a simple spanning cycle. Any graph that has a Hamiltonian cycle is called *Hamiltonian*. A graph is *1-Hamiltonian* if removing *any* vertex from the graph makes it Hamiltonian.

Dillencourt and Smith [2] proved that:

Theorem 2. [2] *A 1-Hamiltonian planar graph is of inscribable type.*

They also observed (see [2], page 66 and [6], page 575) that:

Lemma 1. *A plane graph $G = (V, E, F, f)$ is realizable as a Delaunay tessellation if and only if its stellation G_s is of inscribable type.*

The first proof of Theorem 1 follows from these two results and the following lemma:

Lemma 2. *A stellated outerplanar graph G_s is 1-Hamiltonian.*

Proof. Let us label the vertices of the underlying outerplanar graph G of G_s as $\{1, 2, \ldots, n\}$ in counterclockwise order along its outer face (Hamilton cycle). If we remove vertex s from G_s, we get the original, Hamiltonian graph G. If we remove a vertex i, $1 \leq i \leq n$, then we find the Hamiltonian cycle $i+1, i+2, \ldots, i-1, s, i+1$, with mod N index arithmetic (in the set $\{1, \cdots, n\}$). Hence G_s is 1-Hamiltonian. □

This completes the first proof of Theorem 1.

3 The Main Proof

We turn now to the main result. We give a more technical proof of Theorem 1, based on the following general characterization of inscribable graphs.

Theorem 3. **[Rivin]** *A planar graph $G = (V, E)$ is of inscribable type if and only if:*

1. *G is 3-connected, and*
2. *There exists an assignment $w : E \to \mathbb{R}$ of weights $w(e)$ to the edges $e \in E$ such that:*

(a) **(Edge condition)** *For each edge e, $0 < w(e) \leq 1/2$.*
(b) **(Vertex condition)** *For each vertex v, the sum of the weights of edges incident to v is 1.*
(c) **(Cutset condition)** *For each non-coterminous cutset $C \subseteq E$, the sum of the weights of its edges is at least 1.*

Theorem 3 can be found in [2] (page 65), as a reformulation in the Euclidean space of the general, hyperbolic space result of Rivin et al.[4] (page 247).

Combining this with Lemma 1, we have to prove that the stellation G_s of any outerplanar graph G is 3-connected and has the weight assignment properties from Theorem 3. We prove 3-connectivity first.

Lemma 3. *Any outerplanar graph is 2-connected.*

Proof. In an outerplanar graph, all the vertices lie on the unbounded face f. If we label the vertices as $1, 2, \ldots, n$ in the order in which they appear on the outer face, there are two independent paths between any pair of vertices i and j: one from $i, i+1, \ldots, j$ and another is $i, i-1, \ldots, j$. □

Lemma 4. *The stellation G_s of an outerplanar graph $G = (V, E, F, f)$ is 3-connected.*

Proof. By definition, the stellation of a planar graph is also a planar graph. We must show that there exist three independent paths between any pair of vertices i and j of G_s. If $i, j \in E$, the previous Lemma 3 gives two independent paths between i and j. A third independent path is (i, s, j). If $i \neq s$ and $j = s$, we obtain three independent paths (s, i), $(s, i-1, i)$ and $(s, i+1, i)$ (index arithmetic is done modulo n in the range $1, \ldots, n$). □

For convenience, we restate the weight assignment conditions in terms of the dual graph $G^* = (V^*, E^*, F^*)$.

Theorem 4. [Dual formulation] *A 3-connected planar graph $G = (V, E)$ is of inscribable type if and only if its dual G^* has an assignment $w : E \to \mathbb{R}$ of weights on its edges $e \in E^* = E$ such that:*

1. **(Edge condition)** *For each edge $e \in E^*$, $0 < w(e) \leq 1/2$.*
2. **(Face condition)** *For each face $f \in F^*$, the sum of its edge weights is 1.*
3. **(Cycle condition)** *For each non-facial cycle $C \subseteq E^*$, the sum of its edge weights is at least 1.*

With this formulation, and using Lemma 1, the proof of Theorem 1 is reduced to proving:

Theorem 5. *Let G_s be the stellation of an outerplanar graph. Then there exists a weight assignment on the edges of its dual graph G_s^* such that the three edge, face and cycle conditions of Theorem 4 are satisfied.*

In the next section, we show the existence of a weight assignment satisfying the face condition (Lemma 6). The proof of Theorem 5 is completed in Section 3.2, where we verify that the weight assignment satisfies the edge and cycle conditions (Lemmas 8 and 9).

3.1 Weight Assignment

We start by having a closer look at the structure of the dual of a stellated outerplanar graph G_s^*. If we remove the cycle C made of the duals of the stellating edges of G_s, what remains is a tree, whose leaves lie on the cycle C. Removing the leaf edges of the tree, what remains is a smaller tree called the *backbone*; this is actually the dual of the outerplanar graph without the outer face. We thus partition the edges of G_s^* into three classes: cycle, backbone and leaf edges. See Fig 1 for an example.

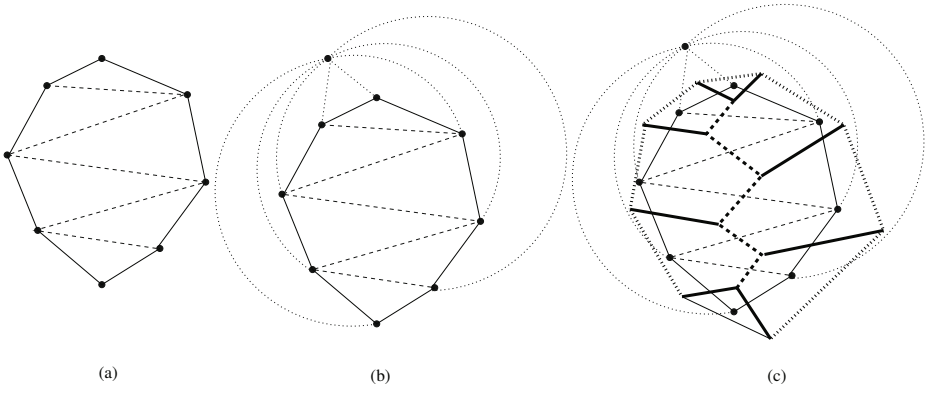


Fig. 1. (a) A maximal outerplanar graph, with the outer face cycle (solid edges) and interior edges (dashed). (b) Its stellation. Dotted edges are the stellating edges. (c) The dual graph of the stellated outerplanar graph. The dual edges are thicker. Bold dotted, dashed and normal edges are cycle, backbone and leaf edges, respectively.

The graph obtained by *contraction* of an edge ij has the two vertices i and j merged into one new vertex v and the edges incident to either i or j become incident to v. The opposite operation is called *edge expansion*.

A graph G has an *edge-expansion inductive construction*, starting from a base graph G_0 if there exists a sequence of graphs $G_0, G_1, \ldots, G_k$ such that G_{i+i} is obtained from G_i by an edge expansion and $G_k = G$.

Lemma 5. *The dual of a stellated maximal outerplanar graph G_s^* has an edge-expansion inductive construction starting from a wheel graph.*

Proof. We perform contraction and expansion operations on G_s^*. A contraction is applied on a backbone edge, one at a time, in an arbitrary order. When all backbone edges are contracted, we obtain a wheel graph where boundary edges are cycle edges and remaining edges are leaf edges of G_s^* (see Fig 2(b)). This sequence of contractions, taken in reverse, gives an edge-expansion inductive construction for G_s^*. □

Next we show that we can assign weights on the edges of G_s^* to meet condition 2 (the face condition) of Theorem 4.

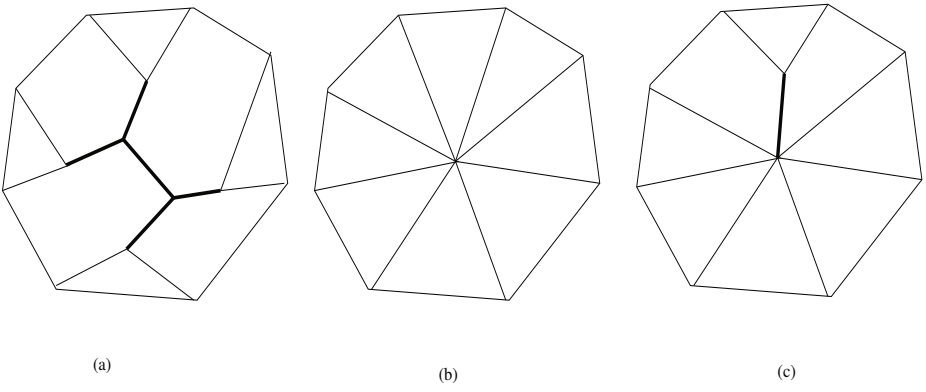


Fig. 2. (a) The dual of a stellated outerplanar graph. Thick edges are backbone edges. (b) Dual graph after contraction of all backbone edges. (c) Expansion of a single backbone edge.

Lemma 6. *There exists a weight assignment on the dual of a stellated outerplanar graph G_s^* such that the sum of the edge weights of each face is 1.*

Proof. We assign weights on G_s^* in an inductive fashion, based on an edge expansion sequence $G_0, G_1, \ldots, G_{n-3}$ for G_s^*.

Base case: G_0 is a wheel graph on n vertices (see Fig 2(b)). We assign weight $1/n$ to each cycle edge and $\frac{n-1}{2n}$ to each leaf edge.

Inductive step: Let $f_1, \ldots, f_n$ be the faces dual to the n vertices of G, labeled in counter clockwise order. Assuming that we have completed a weight assignment for G_k, let G_{k+1} be obtained from G_k by the expansion of edge ij between faces f_i and f_j. We assign a weight of ε, (for a value of $0 < \varepsilon \leq 1/2$ that will be determined later) on the edge ij. As a result the sum of the weights on faces f_i or f_j is imbalanced. We remove the imbalance by subtracting $\frac{\varepsilon}{2}$ from the cycle edges of f_i and f_j, and subtracting $\frac{\varepsilon}{4}$ from each of the two leaf edges of f_i and f_j, respectively. Although this restores the balance of weights for faces f_i and f_j, it creates an imbalance for faces $f_{i-1}, f_{i+1}, f_{j-1}, f_{j+1}$ and cycle edges of G_s^*. To fully balance the weights, we add $\frac{\varepsilon}{4}$ to the cycle edges of these four faces. See Figure 3. Now the sum of the weights of the edges of each face is 1. This weight assignment obviously satisfies the face condition. □

The maximum and minimum possible weights on the edges of G_s^* will be useful in completing the proof.

Lemma 7. *Over all the stellated outerplanar graphs, the maximum possible weight is $\frac{1}{n} + \frac{(n-2)\varepsilon}{4}$, and the minimum is $\frac{1}{n} - \frac{(n-3)\varepsilon}{2}$.*

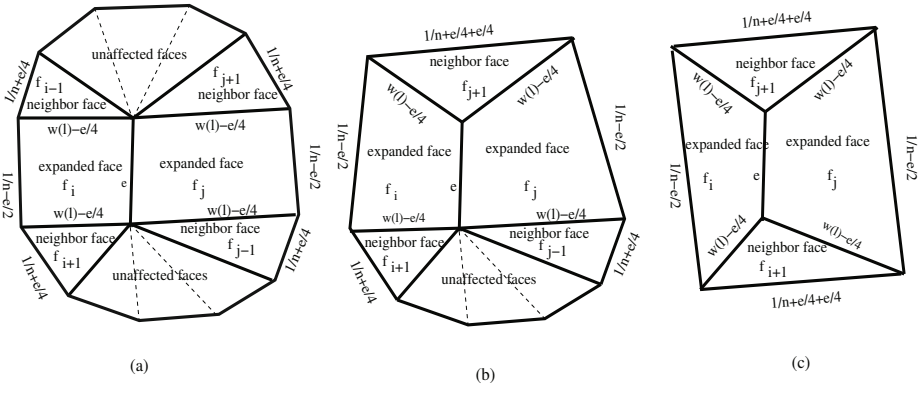


Fig. 3. Three possible cases when a backbone edge is expanded, leading to the expansion of two faces, which have, as neighbors, (a) four distinct faces, (b) two distinct faces and one common face and (c) two common faces. When a backbone edge is expanded, only the weights of these neighbor faces have to be adjusted; all the others remain unchanged.

Proof. As described in the inductive step of the proof for Lemma 6, when a backbone edge e incident to faces f_i and f_j is expanded, the weights of the two cycle edges of these faces are decremented by $\varepsilon/2$. Also, the weight of each of the cycle edges of four adjacent faces of f_i and f_j faces, $f_{i-1}, f_{i+1}, f_{j-1}$ and f_{j+1} is increased by $\varepsilon/4$. However, if the expanded backbone edge e is a leaf of the backbone tree, then $f_{i-1} = f_{j+1}$ (and/or $f_{i+1} = f_{j-1}$), as illustrated in Fig 3(b) and 3(c). In this case, the weight of the cycle edge of the face f_{i-1} is increased twice, each time by $\varepsilon/4$.

The minimum, resp. maximum weight of an edge is attained when its weight is reduced by $\frac{\varepsilon}{2}$, resp. increased by $\frac{\varepsilon}{4}$, at each backbone expansion step. This happens when the original outerplanar graph G has all diagonals emanating from a single vertex, or, equivalently, when the dual graph G^* has one face incident to all the backbone edges. See Fig 4. Consider a face f in G_s^* corresponding to such a vertex in G.

Each time a non-leaf edge of the backbone tree is expanded, the weight of each of the two cycle edges on the two faces adjacent to f is increased by $\frac{\varepsilon}{4}$. However, if the expanded edge is a leaf edge of the backbone tree, then one of the cycle edges' weight is increased twice, each time by $\varepsilon/4$. See Fig 4(b). Thus the weight of each such edge is increased by at most $\frac{\varepsilon(k+1)}{4}$, where k is the number of backbone edges in G_s^* or, equivalently, diagonals in G. Since k is $n-3$, we obtain the maximum weight on any edge of G_s^* as being $\frac{1}{n} + \frac{\varepsilon(n-2)}{4}$.

Similarly, each time a backbone edge is expanded, the weight of the cycle edge of f is reduced by $\frac{\varepsilon}{2}$ (e.g. edge c_1 in Fig 4(b)). When all backbone edges are expanded, the weight of this edge is at least $\frac{1}{n} - \frac{(n-3)\varepsilon}{2}$. This gives the minimum possible weight on any edge of G_s^*. □

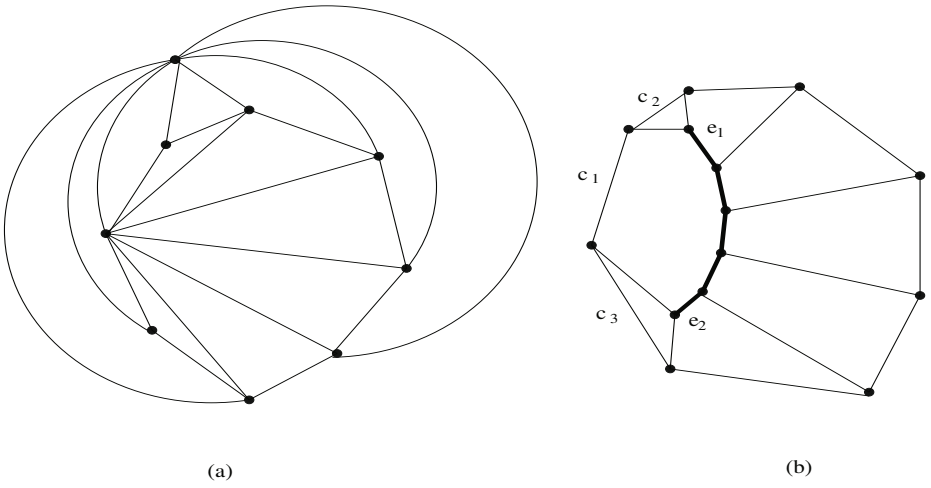


Fig. 4. (a) A stellated outerplanar graph where all diagonals emanate from a single vertex. (b) Its dual. Backbone edges are thickened. The cycle edge whose weight is decremented by $\varepsilon/2$ at each backbone edge expansion step is labeled by c_1. The cycle edges whose weights are increased by $\varepsilon/4$ for each expansion of the backbone edge are c_2 and c_3. A special case occurs when e_1 and e_2 are expanded: the weights of c_2 and c_3 are then increased by $\frac{\varepsilon}{2}$.

3.2 The Edge and Cycle Conditions

To conclude the main proof, we show now how to choose ε such that conditions 1 (edge) and 3 (cycle) of Theorem 4 are also satisfied. Obviously, ε has to be strictly greater than 0 to be a valid weight on backbone edges. Now we establish an upper bound of ε.

Lemma 8. **[Edge condition]** *If* $0 < \varepsilon < \frac{2}{n(n-3)}$, *then* $0 < w(e) \leq 1/2$ *for any edge* e *of* G_s^*.

Proof. We find an upper bound on ε from the constraint that the weight on any edge should be between 0 and $\frac{1}{2}$. Bounding from below the minimum possible weight on an edge of G_s^* (Lemma 7) and solving $\frac{1}{n} - \frac{(n-3)\varepsilon}{2} > 0$, we obtain $\varepsilon < \frac{2}{n(n-3)}$. Similarly for the maximum, bounded from above by 1/2: solving $\frac{1}{n} + \frac{(n-2)\varepsilon}{4} \leq 1/2$ results in $\varepsilon \leq \frac{2}{n}$. The final resulting bounds are $0 < \varepsilon < min\{\frac{2}{n(n-3)}, \frac{2}{n}\}$ or $0 < \varepsilon < \frac{2}{n(n-3)}$. □

Lemma 9. **[Cycle condition]** *If* $0 < \varepsilon < \frac{4(n^2-4n+9)}{3n(n-2)(n-3)}$, *then the sum of the weights on the edges of a non-facial cycle of* G_s^* *is strictly greater than* 1.

Proof. Any non-facial cycle C of G_s^* divides the plane into two regions, each one containing exactly one of the $n+1$ faces of G_s^*. Let R_i be the region containing

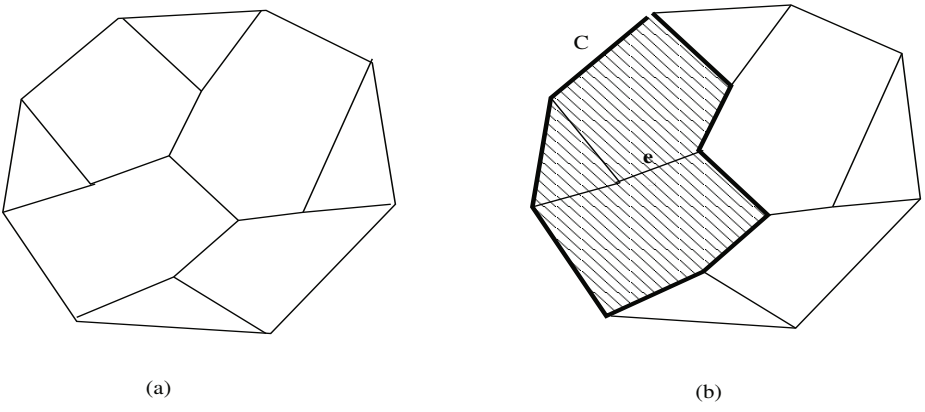


Fig. 5. (a) A dual G_s^* of a stellated outerplanar graph. (b) A non-facial cycle C of G_s^*, shown in thick lines, divides the plane into two regions. The shaded region, containing e as an internal edge, has the smaller number of faces.

the smallest number $k \leq \frac{n+1}{2}$ of faces. An edge is called *internal* to a region if it lies inside it, i.e. not in the complementary region and not on the boundary cycle. Since the sum of weights on each face is 1, then the sum of the weights on the cycle C is the sum k of the weights on the internal k faces, minus twice the sum of the weights on the internal edges of R_i. Since k faces are dual of k vertices in the stellated outerplanar graph G_s and the subgraph induced by k vertices is planar, there are at most $3k-6$ internal edges in R_i. Since $k \leq \frac{n+1}{2}$, there are at most $\frac{3(n+1)}{2} - 6$ or $\frac{3n-9}{2}$ internal edges of R_i. This bounds the sum of the weights on C by at most $\frac{n+1}{2}$ minus twice the sum of the weights on $\frac{3n-9}{2}$ internal edges. Let w_{max} be the maximum possible weight on any of these internal edges. Then it suffices to show that $\frac{n+1}{2} - \frac{2w_{max}(3n-9)}{2} > 1$. Rearranging the terms, we get $w_{max} < \frac{n-1}{6(n-3)}$.

The maximum possible weight on any edge of G_s^* is $\frac{1}{n} + \frac{(n-2)\varepsilon}{4}$ (Lemma 7). Thus $\frac{1}{n} + \frac{(n-2)\varepsilon}{4} < \frac{n-1}{6(n-3)}$. Solving this equation, we get $\varepsilon < \frac{2(n^2-7n+18)}{3n(n-2)(n-3)}$. Therefore, the cycle condition is satisfied if $\varepsilon < \frac{2(n^2-7n+18)}{3n(n-2)(n-3)}$. □

To satisfy both the edge and cycle conditions, we will choose an ε such that $0 < \varepsilon < min\{\frac{2}{n(n-3)}, \frac{2(n^2-7n+18)}{3n(n-2)(n-3)}\}$.

This concludes our main proof.

References

1. Dillencourt, M.B.: Realizability of Delaunay triangulations. Information Processing Letters 33(6), 283–287 (1990)
2. Dillencourt, M.B., Smith, W.D.: Graph-theoretic conditions for inscribability and Delaunay realizability. Discrete Mathematics 161(1-3), 63–77 (1996)

3. Grünbaum, B.: Convex Polytopes. John Wiley and Sons (1967)
4. Hodgson, C.D., Rivin, I., Smith, W.D.: A characterization of convex hyperbolic polyhedra and of convex polyhedra inscribed in the sphere. Bulletin of the AMS 27(2), 246–251 (1992)
5. Lambert, T.: An optimal algorithm for realizing a Delaunay triangulation. Information Processing Letters 62(5), 245–250 (1997), Implementation at http://www.cse.unsw.edu.au/~lambert/java/realize/
6. Rivin, I.: Euclidean structures on simplicial surfaces and hyperbolic volume. Annals of Mathematics 139(3), 553–580 (1994)
7. Rivin, I.: A characterization of ideal polyhedra in hyperbolic 3-space. Annals of Mathematics 143(1), 51–70 (1996)

Representing Reaction Systems by Trees

R. Brijder[1], A. Ehrenfeucht[2], and G. Rozenberg[2,3]

[1] Hasselt University and Transnational University of Limburg, Belgium
[2] Department of Computer Science,
University of Colorado at Boulder, USA
[3] Leiden Institute of Advanced Computer Science,
Leiden Center for Natural Computing,
Leiden University, The Netherlands

Abstract. Reaction systems formally model the functioning of the living cell. By representing sets of reactions by trees, we obtain a useful tool to investigate the state spaces of reaction systems. In particular, we give an upper bound on the fraction of inactive states within a subspace of the state space. This subspace represents partial knowledge of the (unknown) state under consideration.

1 Introduction

Reaction systems (see, e.g., [6] and [1]) are a formal model of the functioning of the living cell based on the idea/intuition that this functioning is determined by interactions of biochemical reactions (taking place in the cell) and these interactions are driven by two mechanisms: facilitation and inhibition.

Following the basic biochemical intuition, a *reaction* is formalized as a triplet $a = (R, I, P)$, where R, I, P are finite sets with R and I being disjoint. The sets R, I, P are called the set of *reactants*, the set of *inhibitors*, and the set of *products*, respectively. Then a *reaction system* is defined as an ordered pair $\mathcal{A} = (S, A)$, where A is a finite set of reactions, and S is a finite set such that, for each reaction in A, all three component sets are included in S. Hence a reaction system $(\mathcal{A})$ is basically a finite set of reactions (A) — we also specify the *background set* (S) which consists of *entities* needed to define the reactions and for reasoning about the system.

The behaviour of a reaction system $\mathcal{A} = (S, A)$ is formalized as follows. A state T of $\mathcal{A}$ is simply a set of entities, i.e., $T \subseteq S$. Then a reaction $a = (R, I, P) \in A$ is *enabled* by T, if all reactants of a are present in T (hence $R \subseteq T$) and none of the inhibitors of a is present in T (hence $I \cap T = \varnothing$). If a is enabled by T, then it produces its products (hence P will be included in the successor state of T). The effect of the whole set of reactions A on T (hence the effect of $\mathcal{A}$ on T) is cumulative: it is the union of the product sets of all reactions in A that are enabled by T.

Thus the behaviour of $\mathcal{A}$ is defined by its *state space* (the set of all subsets of S) together with all trajectories, i.e., all sequences of states such that

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 330–342, 2012.

each next (successor) state is produced from a current state T by all reactions of $\mathcal{A}$ enabled by T.

Research topics concerning reaction systems are motivated either by biological considerations or by the need to understand the underlying computations. As a matter of fact, although originally motivated/inspired by the functioning of the living cell, by now reaction systems became a novel, elegant and challenging model of computation. Examples of research topics include: the studies of result functions that determine the trajectories/processes (see, e.g., [3] and [4]), causalities between entities ([2]), formation of (biological and biochemical) modules ([5]), and the issue of time in reaction systems ([7]).

In this paper we consider a representation of (the sets of reactions of) reaction systems. The representation we provide allows one to reason about the state spaces of reaction systems. The underlying intuition of this connection is the fact that the current state of a biochemical system (the cell) is often unknown, and one may only determine the existence and absence of some entities. Given sets U and V for which $U \subseteq W$ and $V \cap W = \varnothing$ for some unknown state W, we deduce an upper bound on the fraction of the states X, satisfying $U \subseteq X$ and $V \cap X = \varnothing$, for which no reaction is enabled. We efficiently obtain this result by representing sets of reactions as trees.

This paper is organized as follows. In Section 2 we settle/recall the basic notation and terminology concerning set families, and (labelled) graphs and trees. In Section 3 we discuss a representation of families of (pairwise incomparable) sets by trees, and then show that each such tree may be "optimally selected". The setup is generic, and does not depend on the notion of reaction system. Next, in Section 4, we consider (generic) states and substates in relation to trees. In Section 5 we formally recall the notion of reaction system and related notions, and in Section 6 we apply the results of Sections 3 and 4 to reaction systems.

2 Preliminaries

In this section we recall some basic notions concerning sets, graphs and trees in order to fix notation and terminology for this paper.

Two sets X and Y are called *incomparable* if both $X \not\subseteq Y$ and $Y \not\subseteq X$. Let $\mathcal{F}$ be a family of subsets of a finite set F. A *selector* (or *choice function*) of $\mathcal{F}$ is a function $c : \mathcal{F} \to F$, where $c(X) \in X$ for all $X \in \mathcal{F}$. We say that $c(\mathcal{F})$ is a *selection in* $\mathcal{F}$. Note that if S is a selection, then $S \cap X \neq \varnothing$ for all $X \in \mathcal{F}$. The term "smallest" means minimal w.r.t. cardinality. For example, a smallest selection S in $\mathcal{F}$ is a selection S in $\mathcal{F}$ which is minimal w.r.t. cardinality among all selections in $\mathcal{F}$. Since F is finite, a smallest selection exists.

A *directed graph* (*digraph*) is an ordered pair $G = (V, E)$, where V is a finite set of *vertices*, and $E \subseteq V \times V$ is the set of (directed) *edges*. A *labelled digraph* G is a 4-tuple (V, E, Σ, l), where (V, E) is a digraph, Σ is a finite alphabet (of *labels*), and $l : E \to \Sigma$ an *edge labelling*. A *path* in G is a sequence $\pi = e_1 e_2 \cdots e_n$ of edges of G such that there is a (unique) sequence of vertices $v_1 v_2 \cdots v_{n+1}$ with $e_i = (v_i, v_{i+1})$ for all $i \in \{1, \ldots, n\}$. The *label set* of path π, denoted by $\mathrm{ls}(\pi)$, is the set $\{l(e) \mid e \text{ is an edge of } \pi\}$. The out-degree of a vertex v is denoted by $\deg(v)$.

A *tree* $T = (V, E)$ is a digraph, where $|E| = |V| - 1$, with a unique vertex $r \in V$, called the *root* of T, such that there is a (unique) path from r to any vertex of T. In this paper we consider mostly labelled trees. For $v \in V$, the subtree of T rooted in v is denoted by $T[v]$, and we let $\mathrm{hgt}(v)$ be the height (i.e., the maximal length among the paths from the root to a leaf) of $T[v]$. Finally, the set of leaves of T is denoted by $\mathrm{leav}(T)$.

3 Representing Families of Sets as Trees

In this section we discuss how to represent families of sets by (unambiguously labelled) trees.

For a vertex v of a labelled tree T, we define the *support* of v (in T), denoted by $\mathrm{sup}_T(v)$, as the set of labels that appear in the (unique) path from the root to v. This is more formally defined as follows.

Definition 1. *Let $T = (V, E, \Sigma, l)$ be a labelled tree. The support function, sup_T, is defined by: $\mathrm{sup}_T : V \to 2^\Sigma$, with $\mathrm{sup}_T(v) = \mathrm{ls}(\pi_v)$, for all $v \in V$, where π_v is the (unique) path in T from the root r to v.*

We write sup rather than sup_T whenever T is clear from the context. We say that T is *unambiguously labelled* if sup is injective, and we say that T is *ambiguously labelled* otherwise. Also, we set $\mathrm{supl}_T = \{\mathrm{sup}_T(v) \mid v \in \mathrm{leav}(T)\}$.

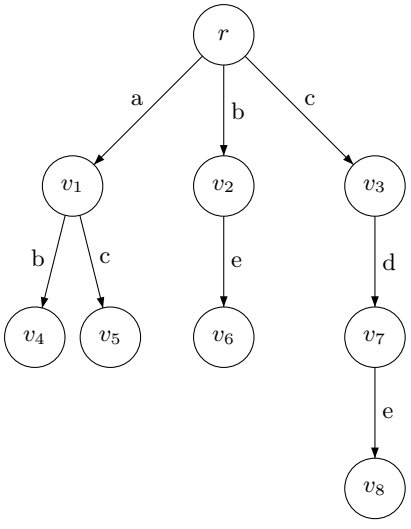


Fig. 1. An unambiguously labelled tree

Example 2. Consider the tree T of Figure 1. We have, e.g., $\mathrm{sup}_T(v_2) = \{b\}$, $\mathrm{sup}_T(v_8) = \{c, d, e\}$, and $\mathrm{sup}_T(r) = \varnothing$. It is easy to verify that T is unambiguously labelled.

The following result states two basic properties of unambiguously labelled trees.

Lemma 3. *Let $T = (V, E, l)$ be a unambiguously labelled tree. The following conditions hold.*

1. *If π is a path in T, then the labels of any two distinct edges of π are distinct.*
2. *If e_1 and e_2 are two outgoing edges of any $v \in V$, then the labels of e_1 and e_2 are distinct.*

Proof. Assume to the contrary that a path $\pi = e_1 \dots e_n$ is such that $l(e_i) = l(e_j)$ for some $1 \leq i < j \leq n$. If $e_j = (v_1, v_2)$, then $\sup(v_2) = \sup(v_1) \cup \{l(e_j)\} = \sup(v_1) \cup \{l(e_i)\} = \sup(v_1)$ (as $l(e_i) \in \sup(v_1)$). Hence T is ambiguously labelled — a contradiction.

Let $e_1 = (v, v_1)$ and $e_2 = (v, v_2)$ be outgoing edges of some $v \in V$. If $l(e_1) = l(e_2)$, then $\sup(v_1) = \sup(v) \cup \{l(e_1)\} = \sup(v) \cup \{l(e_2)\} = \sup(v_2)$ and therefore T is ambiguously labelled. A contradiction. □

From now on we consider only unambiguously labelled tree and use the simple term "*tree*" rather than "unambiguously labelled tree". Also, we often simply write V to denote the vertex set of the tree under consideration.

For a vertex v, we let $O(v)$ to denote the set of labels of edges outgoing from v. Note that as T is unambiguously labelled, Condition 2 of Lemma 3 implies that $|O(v)| = \deg(v)$ for all $v \in V$.

For $v \in V$, we let $\mathcal{F}_T[v] = \mathrm{supl}_{T[v]}$, i.e., $\mathcal{F}_T[v]$ is the family of label sets of all paths from v to leaves of T. For the root r, $\mathcal{F}_T[r] = \mathrm{supl}_T$, and $\mathcal{F}_T[v] = \{\varnothing\}$ iff v is a leaf. Note that $O(v)$ is a selection in $\mathcal{F}_T[v]$. Alternatively, $\mathcal{F}_T[v]$ can be defined recursively. Indeed, if v is a leaf, then $\mathcal{F}_T[v] = \{\varnothing\}$, and if v is not a leaf, then $\mathcal{F}_T[v] = \{Z \cup \{l(e)\} \mid Z \in \mathcal{F}_T[v'] \text{ and } e = (v, v') \in E\}$.

Example 4. Consider again Example 2. We have, e.g., $O(r) = \{a, b, c\}$. Also, e.g., $\mathcal{F}_T[v_1] = \{\{b\}, \{c\}\}$ and $\mathcal{F}_T[r] = \mathrm{supl}_T = \{\{a, b\}, \{a, c\}, \{b, e\}, \{c, d, e\}\}$.

Thus, trees can be used to define families of sets: we say that a tree T *represents* a family $\mathcal{F}$ of sets if $\mathcal{F} = \mathcal{F}_T[r]$.

Definition 5. *Let $\mathcal{F}$ be a finite family of sets. A tree T is called* optimally selected for $\mathcal{F}$ *if T represents $\mathcal{F}$ and, for each $v \in V$, $O(v)$ is a smallest selection in $\mathcal{F}_T[v]$.*

Note that the out-degree of the root of an optimally selected tree T for $\mathcal{F}$ is minimal among all trees representing $\mathcal{F}$. Hence, from this point of view, a tree is optimally selected if the out-degree of each vertex is minimized using a greedy minimization approach starting from the root vertex. This is more precisely demonstrated in the proof of Theorem 7.

Example 6. Consider again the tree T of Figure 1. Note that T is not optimally selected as $O(r) = \{a, b, c\}$, while $\{a, e\}$ and $\{b, c\}$ are (the smallest) selections in $\mathcal{F}_T[r]$. Let $\mathcal{G} = \mathcal{F}_T[r]$. An optimally selected tree for $\mathcal{G}$ is given in Figure 2.

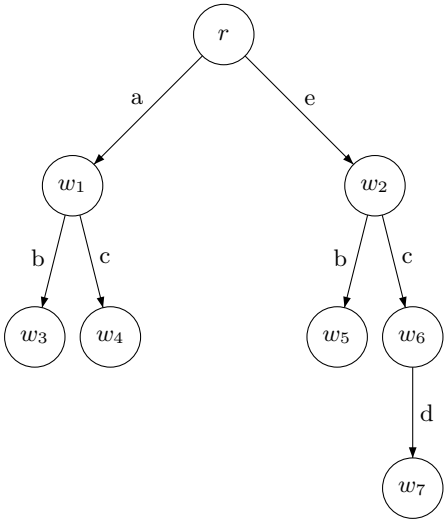


Fig. 2. An optimally selected tree, cf. Example 6

Let $\mathcal{F}$ be a family of pairwise incomparable sets (hence for any distinct $X, Y \in \mathcal{F}$ we have $X \not\subseteq Y$ and $Y \not\subseteq X$). Theorem 7 shows that one can iteratively construct an optimally selected tree for $\mathcal{F}$. This is done by starting from the root and in each step introducing all outgoing edges and vertices from a vertex v according to a smallest selection in $\mathcal{F}_T[v]$.

Theorem 7. *Let $\mathcal{F}$ be a finite family of pairwise incomparable sets. There exists an optimally selected tree for $\mathcal{F}$.*

Proof. Assume that $\mathcal{F} \subseteq 2^Q$, i.e., the sets in $\mathcal{F}$ are subsets of a ground set Q. We recursively construct an optimally selected tree $k(\mathcal{F})$ for $\mathcal{F}$.

If $\mathcal{F} = \{\varnothing\}$, then we define $k(\mathcal{F})$ to be a tree having only a single vertex — clearly $k(\mathcal{F})$ is an optimally selected tree for $\mathcal{F}$.

Assume now $\mathcal{F} \neq \{\varnothing\}$. Let $\{l_1, \ldots, l_n\}$ be a smallest selection in $\mathcal{F}$ and let $\mathcal{F}_i = \{Z \backslash \{l_i\} \mid Z \in \mathcal{F} \text{ and } l_i \in Z\}$. Then each $\mathcal{F}_i$ is a family of pairwise incomparable sets. Now, define $k(\mathcal{F})$ to be a tree obtained from the trees $k(\mathcal{F}_1), \ldots, k(\mathcal{F}_n)$ by introducing a new (root) vertex r and adding for each $i \in \{1, \ldots, n\}$ an edge labelled by l_i from r to the root of $k(\mathcal{F}_i)$.

Recall that by definition $\mathcal{F}_T[v] = \{\varnothing\}$ if v is a leaf, and $\mathcal{F}_T[v] = \{Z \cup \{l(e)\} \mid Z \in \mathcal{F}_T[v'] \text{ and } e = (v, v') \in E\}$ if v is not a leaf. Hence, $T = k(\mathcal{F})$ satisfies $\mathcal{F} = \mathcal{F}_T[r]$, and therefore T represents $\mathcal{F}$. By construction, for each $v \in V$, $O(v)$ is a smallest selection in $\mathcal{F}_T[v]$, and hence T is an optimally selected tree for $\mathcal{F}$. □

The following example shows that the condition requiring that the sets in $\mathcal{F}$ are pairwise incomparable is needed for Theorem 7 to hold.

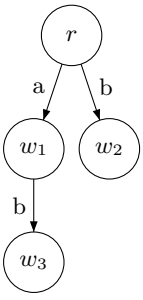


Fig. 3. An unambiguously labelled tree, cf. Example 8

Example 8. Let $\mathcal{F} = \{\{a, b\}, \{b\}\}$. Figure 3 depicts an unambiguously labelled tree that represents $\mathcal{F}$. However, there is no optimally selected tree for $\mathcal{F}$, because the unique smallest selection for $\mathcal{F}$ is $\{b\}$.

It is important to note that for a given family $\mathcal{F}$ of pairwise incomparable sets, an optimally selected tree for $\mathcal{F}$ may be not unique. For example, for $\mathcal{F} = \{\{a, b\}\}$ there are two trees (up to isomorphism) representing $\mathcal{F}$, and both are optimally selected. Indeed, both of these trees "are" a path π of length 2 from the root to the unique leaf, where the labels a and b appear in π either in the order a, b or in the order b, a.

Let $\mathcal{F}$ be a family of sets. Then $\mathcal{G} \subseteq \mathcal{F}$ is *maximally disjoint* w.r.t. $\mathcal{F}$ if the sets of $\mathcal{G}$ are pairwise disjoint and $\mathcal{G}$ is maximal with this property (i.e., each $Y \in \mathcal{F} \setminus \mathcal{G}$ properly intersects with some set in $\mathcal{G}$).

Lemma 9. *Let $\mathcal{F}$ be a finite nonempty family of sets such that $l = \max\{|X| \mid X \in \mathcal{F}\}$ and d is the cardinality of a smallest selection in $\mathcal{F}$. If $\mathcal{G} \subseteq \mathcal{F}$ is maximally disjoint w.r.t. $\mathcal{F}$, then $|\mathcal{G}| \geq \frac{d}{l}$.*

Proof. Let $\mathcal{G} \subseteq \mathcal{F}$ be maximally disjoint w.r.t. $\mathcal{F}$. Hence for each $Y \in \mathcal{F} \setminus \mathcal{G}$, $(\cup_{X \in \mathcal{G}} X) \cap Y \neq \varnothing$. Also, for each $Y \in \mathcal{G}$, $(\cup_{X \in \mathcal{G}} X) \cap Y \neq \varnothing$. Therefore there is a selection S in $\mathcal{F}$ with $S \subseteq \cup_{X \in \mathcal{G}} X$. Hence $d \leq |S| \leq |\cup_{X \in \mathcal{G}} X| \leq l|\mathcal{G}|$. Consequently, $|\mathcal{G}| \geq \frac{d}{l}$. □

We now apply Lemma 9 to families of pairwise incomparable sets.

Corollary 10. *Let $\mathcal{F}$ be a finite nonempty family of pairwise incomparable sets, and let T represent $\mathcal{F}$. Let $v \in V$ be such that $O(v)$ is a smallest selection in $\mathcal{F}_T[v]$. Then there exists a subfamily $\mathcal{G}$ of $\mathcal{F}_T[v]$ consisting of disjoint sets such that $|\mathcal{G}| \geq \frac{\deg(v)}{\mathrm{hgt}(v)}$.*

Proof. Since $|O(v)| = \deg(u)$ is the cardinality of a smallest selection in $\mathcal{F}_T[v]$, and $|X| \leq \mathrm{hgt}(v)$ for all $X \in \mathcal{F}_T[v]$, by Lemma 9 we obtain $|\mathcal{G}| \geq \frac{\deg(v)}{\mathrm{hgt}(v)}$. □

4 State Spaces in Trees

We now consider results concerning state spaces of reaction systems in relation to trees. As these results are quite generic and possibly applicable to other domains, we choose to first focus on the essential properties needed to obtain our results (without yet defining reaction systems), and then, in Section 6, relate the results in a precise way to the domain of reaction systems.

Let S be a finite set. We let $\bar{S} = \{\bar{x} \mid x \in S\}$ be a disjoint copy of S, i.e., $S \cap \bar{S} = \varnothing$. Moreover, we let $\bar{\bar{x}} = x$ for $x \in S$. For any subset $X \subseteq (S \cup \bar{S})$, we write $\bar{X} = \{\bar{x} \mid x \in X\}$.

Intuitively (this is made precise in Section 6), we consider S to be the set of all entities of a reaction system. A state W can be considered as a set $Q \cup \overline{(S \setminus Q)}$, where the entities of Q are present and the entities of $S \setminus Q$ are absent (in this state). A substate U is a subset of a state; for the entities in S that do not appear (with or without bar) in a substate it is not known whether or not they are present — hence we deal with incomplete knowledge here. We say that a state W is *compatible* with a substate U if $U \subseteq W$.

Define the *state space* (of S) as $\mathrm{sspace}_S = \{Q \cup \overline{(S \setminus Q)} \mid Q \subseteq S\}$. The elements of sspace_S are called *states* (of S). We define the function $st : 2^S \to \mathrm{sspace}_S$ as follows: for $Q \subseteq S$, $st(Q) = Q \cup \overline{(S \setminus Q)}$. Note that st is a bijection.

A *family of substates* $\mathcal{F}$ (of S) is a subset of $\{Z \mid Z \subseteq W \text{ and } W \in \mathrm{sspace}_S\}$ such that the sets in $\mathcal{F}$ are pairwise incomparable. By Theorem 7 it is possible to represent $\mathcal{F}$ by an optimally selected tree T.

Example 11. Let $S = \{a, b, c\}$. Then $\mathrm{sspace}_S = \{\{a, b, c\}, \{a, b, \bar{c}\}, \{a, \bar{b}, c\}, \{\bar{a}, b, c\}, \{a, \bar{b}, \bar{c}\}, \{\bar{a}, b, \bar{c}\}, \{\bar{a}, \bar{b}, c\}, \{\bar{a}, \bar{b}, \bar{c}\}\}$, and $\mathcal{F} = \{\{a, \bar{b}\}, \{\bar{b}, c\}, \{a, b, c\}\}$ is a family of substates (of S). On the other hand, e.g., $\mathcal{F} = \{\{a, \bar{b}\}, \{a, \bar{b}, c\}\}$ is *not* a family of substates (as $\{a, \bar{b}\} \subset \{a, \bar{b}, c\}$).

Let $\mathcal{F}$ be a family of substates and let T represent $\mathcal{F}$. For each vertex $v \in V$, $\sup(v)$ is a substate of S. The set of states compatible with $\sup(v)$ is denoted by $\mathcal{L}_v$, i.e., $\mathcal{L}_v = \{Q \in \mathrm{sspace}_S \mid \sup(v) \subseteq Q\}$. The set $\mathcal{L}_v$ can be partitioned into the sets $\mathcal{L}_v^+$ and $\mathcal{L}_v^-$, where $\mathcal{L}_v^+$ consists of those states that are compatible with $\sup(u)$ where u is a leaf of $T[v]$. Formally, $\mathcal{L}_v^+ = \{Q \in \mathcal{L}_v \mid \sup(u) \subseteq Q \text{ for some } u \in \mathrm{leav}(T[v])\}$, and $\mathcal{L}_v^- = \mathcal{L}_v \setminus \mathcal{L}_v^+$.

We will need the following technical lemma.

Lemma 12. *Let $\mathcal{F}$ be a family of substates, let T represent $\mathcal{F}$, and let $v \in V$. Let moreover $\mathcal{G} \subseteq \mathcal{F}_T[v]$ be such that the elements of $\mathcal{G}$ are pairwise disjoint, and let $\mathcal{L}_{\mathcal{G}}^- = \{Q \in \mathcal{L}_v \mid Z \not\subseteq Q \text{ for all } Z \in \mathcal{G}\}$. Then*

$$\frac{|\mathcal{L}_{\mathcal{G}}^-|}{|\mathcal{L}_v|} = \prod_{Z \in \mathcal{G}} \left(1 - \frac{1}{2^{|Z|}}\right).$$

Proof. For any $Z \in \mathcal{G}$, the ratio of all $Q \in \mathcal{L}_v$ such that $Z \subseteq Q$ to all $Q \in \mathcal{L}_v$ is $\frac{1}{2^{|Z|}}$. Hence the ratio of all $Q \in \mathcal{L}_v$ such that $Z \not\subseteq Q$ to all $Q \in \mathcal{L}_v$ is $1 - \frac{1}{2^{|Z|}}$.

Now, $\frac{|\mathcal{L}_{\mathcal{G}}^-|}{|\mathcal{L}_v|}$ is the ratio of all $Q \in \mathcal{L}_v$ such that $Z \not\subseteq Q$ for *all* $Z \in \mathcal{G}$ to all $Q \in \mathcal{L}_v$. Consequently, $\frac{|\mathcal{L}_{\mathcal{G}}^-|}{|\mathcal{L}_v|} = \prod_{Z \in \mathcal{G}} \left(1 - \frac{1}{2^{|Z|}}\right)$. □

We now consider the ratio of $|\mathcal{L}_v^-|$ to $|\mathcal{L}_v|$, i.e., the fraction of states that are not compatible with $\sup(u)$ for any leaf u of $T[v]$ among all states that *are* compatible with $\sup(v)$.

Theorem 13. *Let $\mathcal{F}$ be a family of substates, let T represent $\mathcal{F}$, and let $v \in V$ be such that $O(v)$ is a smallest selection in $\mathcal{F}_T[v]$. Then*

$$\frac{|\mathcal{L}_v^-|}{|\mathcal{L}_v|} \leq \left(1 - \frac{1}{2^{\mathrm{hgt}(v)}}\right)^{\frac{\deg(v)}{\mathrm{hgt}(v)}}.$$

Proof. By Corollary 10 there is a subfamily $\mathcal{G}$ of $\mathcal{F}_T[v]$ consisting of disjoint sets such that $|\mathcal{G}| \geq \frac{\deg(v)}{\mathrm{hgt}(v)}$. Let again $\mathcal{L}_{\mathcal{G}}^- = \{Q \in \mathcal{L}_v \mid Z \not\subseteq Q \text{ for all } Z \in \mathcal{G}\}$. Then $\mathcal{L}_v^- \subseteq \mathcal{L}_{\mathcal{G}}^-$. We have now by Lemma 12

$$\frac{|\mathcal{L}_v^-|}{|\mathcal{L}_v|} \leq \frac{|\mathcal{L}_{\mathcal{G}}^-|}{|\mathcal{L}_v|} = \prod_{Z \in \mathcal{G}} \left(1 - \frac{1}{2^{|Z|}}\right).$$

Since $|Z| \leq \mathrm{hgt}(v)$ for all $Z \in \mathcal{G}$, we obtain

$$\frac{|\mathcal{L}_v^-|}{|\mathcal{L}_v|} \leq \prod_{Z \in \mathcal{G}} \left(1 - \frac{1}{2^{\mathrm{hgt}(v)}}\right).$$

Finally, $|\mathcal{G}| \geq \frac{\deg(v)}{\mathrm{hgt}(v)}$ and thus we obtain

$$\frac{|\mathcal{L}_v^-|}{|\mathcal{L}_v|} \leq \left(1 - \frac{1}{2^{\mathrm{hgt}(v)}}\right)^{\frac{\deg(v)}{\mathrm{hgt}(v)}}.$$

Consequently, the theorem holds. □

5 Reaction Systems

In this section we recall some basic notions related to reaction systems, see, e.g., [6] and [1].

Reaction systems is a formal model of the functioning of the living cell. The underlying idea is that this functioning is determined by the interactions between biochemical reactions and these interactions are driven by two mechanisms: facilitation and inhibition.

The formalization of a biochemical reaction follows the basic intuition that a biochemical reaction will take place if all of its reactants are present (in the current state of a biochemical system) and none of its inhibitors is present. When a reaction takes place, it creates its products. This leads to the following definition.

Definition 14. *A* reaction *is a triplet* $a = (R, I, P)$*, where* R, I, P *are finite sets such that* $R \cap I = \varnothing$*.*

The sets R, I, P are also denoted by R_a, I_a, P_a, and called the *reactant set of* a, the *inhibitor set of* a, and the *product set of* a, respectively. If S is a set such that $R, I, P \subseteq S$, then a is a *reaction in* S, and $\mathrm{rac}(S)$ denotes the set of all reactions in S.

Usually (see, e.g., [1]) one requires that, for each reaction (R, I, P), both R and I are nonempty. However, in this paper we use a tree representation of a set of reactions, and moving from the root to the leaf representing a reaction $a = (R_a, I_a, P_a)$ corresponds to gaining knowledge of a (from zero knowledge in the root to the full knowledge in the leaf). On the way along this path we represent the current knowledge by the currently known part of R_a and the currently known part of I_a, and either of these parts may be empty before we arrive at the leaf representing a. Thus, for the sake of simplicity, rather than to introduce the notion of a "pseudo reaction", in this paper we do allow the empty reactant set and the empty inhibitor set.

Example 15. Let $S = \{s_1, s_2, s_3, s_4\}$, $a_1 = (\{s_2, s_3\}, \{s_4\}, \{s_1\})$, and $a_2 = (\{s_2\}, \{s_1\}, \{s_1, s_4\})$. Then $a_1, a_2 \in \mathrm{rac}(S)$ and, e.g., $P_{a_2} = \{s_1, s_4\}$.

A reaction system is a basic construct of the whole framework of reaction systems (see, e.g., [1]). It is essentially a finite set of reactions, however one also specifies the (background) set of all entities which are needed for specifying the reactions and for reasoning about the system.

Definition 16. *A* reaction system *is an ordered pair* $\mathcal{A} = (S, A)$ *such that* S *is a finite set, and* $A \subseteq \mathrm{rac}(S)$*.*

The set S is called the *background set of* $\mathcal{A}$, its elements are called *entities*, and A is called the *set of reactions of* $\mathcal{A}$ — note that since S is finite, so is A.

Definition 17. *Let* W *be a finite set, and let* a *be a reaction. Then* a *is* enabled by W*, denoted by* a en W*, if* $R_a \subseteq W$ *and* $I_a \cap W = \varnothing$*. The* result of a on W*, denoted by* $\mathrm{res}_a(W)$*, is defined by:* $\mathrm{res}_a(W) = P_a$ *if* a en W*, and* $\mathrm{res}_a(W) = \varnothing$ *otherwise.*

A *state* of a reaction system is a subset $W \subseteq S$ of the background set. A reaction a is enabled in state W if *all* of its reactants are present in W while *none* of its inhibitors are in W. This is the reason that we assume in Definition 14 that, for each reaction a, $R_a \cap I_a = \varnothing$, as otherwise a is never enabled. When a takes place it produces entities from P_a.

The effect of a *set* of reactions A is cumulative: the *result of* A *on* W, denoted by $\mathrm{res}_A(W)$, is defined by: $\mathrm{res}_A(W) = \bigcup_{a \in A} \mathrm{res}_a(W)$. For a reaction system $\mathcal{A} = (S, A)$, we write $\mathrm{res}_{\mathcal{A}}(W) = \mathrm{res}_A(W)$.

Example 18. Consider the reaction system $\mathcal{A} = (S, A)$ with S from Example 15 and $A = \{a_1, a_2\}$ with a_1 and a_2 from Example 15. Then for state $W = \{s_2, s_3\}$, we have $\mathrm{res}_{\mathcal{A}}(W) = \{s_1\} \cup \{s_1, s_4\} = \{s_1, s_4\}$, and for this successor state $W' = \{s_1, s_4\}$ we have $\mathrm{res}_{\mathcal{A}}(W') = \varnothing$.

In this paper we assume that the processes of reaction systems are so-called context-independent (see, e.g., [1]), i.e., we assume the behaviour of the reaction systems as a closed system (when there is no interference by the environment). Therefore the state transitions are determined only by the reactions of a reaction system (hence by the result function res_A).

The definition of the result function implies that the successor state consists of only the entities produced by the reactions in the current state. Thus there is *no permanency of entities*: an entity from a current state vanishes (in the transition to the successor state) unless it is produced/sustained by a reaction. This reflects the basic bioenergetics of the living cell, and forms a major difference with models of computation in computer science.

We also notice that the result of the set of reactions is cumulative and so we do not have the notion of conflict between reactions (if they share reactants). Thus we assume the *threshold nature of resources*: either an entity is available and then there is "enough of it" or it is not available. This reflects the level of abstraction adopted in the reaction systems model.

In this paper we consider the notion of enabling. In particular, we are interested, given a reaction system $\mathcal{A}$ and some partial knowledge of a state W, in the ratio of the states of $\mathcal{K}$ for which some reaction of $\mathcal{A}$ is enabled to all states of $\mathcal{K}$, where $\mathcal{K}$ is the set of states "compatible" with the partial knowledge of W.

Two reaction systems are called equivalent if they have the same behavior w.r.t. the $\mathrm{res}_{\mathcal{A}}$ function. This is formalized as follows.

Definition 19. *Reaction systems $\mathcal{A}$ and $\mathcal{A}'$ with common background set S are called* equivalent *if for all $W \subseteq S$, $\mathrm{res}_{\mathcal{A}}(W) = \mathrm{res}_{\mathcal{A}'}(W)$.*

Let a be a reaction, $\{P_1, P_2\}$ be a partition of P_a, $a_1 = (R_a, I_a, P_1)$, and $a_2 = (R_a, I_a, P_2)$. Then, as the result of a set A of reactions is cumulative, we have for all $W \subseteq S$, $\mathrm{res}_a(W) = \mathrm{res}_{\{a_1,a_2\}}(W)$. Hence, the reaction systems $(S, \{a\})$ and $(S, \{a_1, a_2\})$ are equivalent. Consequently, we say that $\mathcal{A}$ is in *singleton product normal form* if $|P_a| = 1$ for all $a \in A$.

Moreover, if $a, a' \in A$ with $P_a = P_{a'}$, $R_a \subseteq R_{a'}$ and $I_a \subseteq I_{a'}$, then for all $W \subseteq S$ $\mathrm{res}_{\{a,a'\}}(W) = \mathrm{res}_{\{a\}}(W)$. Therefore, reaction systems $(S, \{a\})$ and $(S, \{a, a'\})$ are equivalent. Hence, we may delete superfluous reactions: we say that $\mathcal{A}$ (or A) is *reduced* if for all different $a, a' \in A$ with $P_a = P_{a'}$ we cannot have both $R_a \subseteq R_{a'}$ and $I_a \subseteq I_{a'}$.

Example 20. Reaction system $\mathcal{A} = (S, A)$ with $A = \{a_1, a_2, a_3\}$, $a_1 = (\{s_2, s_3\}, \{s_4\}, \{s_1\})$, $a_2 = (\{s_2\}, \{s_1\}, \{s_1\})\}$, and $a_3 = (\{s_2\}, \{s_1\}, \{s_4\})\}$ is both in singleton product normal form and reduced. As a matter of fact, $\mathcal{A}$ is equivalent to the reaction system from Example 18.

We assume from now on that each reaction system is both reduced and in singleton product normal form.

6 State Space of Reaction Systems

In this section we describe how reactions and states of reaction systems may be translated to states and pairwise incomparable sets as discussed in Sections 3 and 4. In this way, we can apply results of these sections to reaction systems — see Theorem 25.

Let $\mathcal{A} = (S, A)$ be a reaction system. As $\mathcal{A}$ is in singleton product normal form, the reactions of $\mathcal{A}$ can be partitioned according to the (unique) product p of each reaction. We fix now a $p \in S$, and consider the reaction system $\mathcal{A}_p = (S, A_p)$ with $A_p = \{a \in A \mid P_a = \{p\}\}$.

Note that A_p can be uniquely represented as the family of substates $\mathcal{F}_{A_p} = \{R_a \cup \bar{I}_a \mid a \in A_p\}$ of S. Indeed, as A is reduced, A_p is reduced as well, and so the sets in $\mathcal{F}_{A_p}$ are pairwise incomparable.

Example 21. Consider the reaction system $\mathcal{A} = (S, A)$ from Example 20. Then $\mathcal{A}_{s_1} = (S, A_{s_1})$ and $A_{s_1} = \{a_1, a_2\}$ with $a_1 = (\{s_2, s_3\}, \{s_4\}, \{s_1\})$ and $a_2 = (\{s_2\}, \{s_1\}, \{s_1\})\}$. We may represent $\mathcal{A}_{s_1}$ by the family $\mathcal{F}_{A_{s_1}} = \{\{s_2, s_3, \bar{s}_4\}, \{s_2, \bar{s}_1\}\}$ of substates. Note that the sets in $\mathcal{F}_{A_{s_1}}$ are indeed pairwise incomparable.

We may also move the other way around. Let $\mathcal{F}$ be a family of substates of S and fix a $p \in S$. For $Y \in \mathcal{F}$, we define $R_Y = Y \cap S$, $I_Y = \bar{Y} \cap S$, and $a_Y = (R_Y, I_Y, \{p\})$. Hence $A_\mathcal{F} = \{a_Y \mid Y \in \mathcal{F}\}$ is a set of reactions corresponding to $\mathcal{F}$. We say that $\mathcal{A}_{p,\mathcal{F}} = (S, A_\mathcal{F})$ is the *reaction system of* $\mathcal{F}$ *w.r.t.* $p \in S$.

Example 22. Consider again the family $\mathcal{F} = \{\{a, \bar{b}\}, \{\bar{b}, c\}, \{a, b, c\}\}$ of substates of $S = \{a, b, c\}$ from Example 11. The reaction system $\mathcal{A}_{p,\mathcal{F}}$ of $\mathcal{F}$ w.r.t. b is defined by: $\mathcal{A}_{p,\mathcal{F}} = (S, \{a_1, a_2, a_3\})$ with $a_1 = (\{a\}, \{b\}, \{b\})$, $a_2 = (\{c\}, \{b\}, \{b\})$, and $a_3 = (\{a, b, c\}, \varnothing, \{b\})$.

If a tree T represents $\mathcal{F}_{A_p}$, then we will also simply say that T *represents* A_p. For a tree T representing A_p and vertex v of T, we define $R_v = R_{\sup(v)}$, $I_v = I_{\sup(v)}$, $a_v = a_{\sup(v)}$, and $A_v = \{a_w \mid w \in \mathrm{leav}(T[v])\}$. Note that $\sup(v) = R_v \cup \overline{I_v}$. We define $\mathcal{K}_v = \{W \subseteq S \mid a_v \text{ en } W\}$, and then we let $\mathcal{K}_v^+ = \{W \in \mathcal{K}_v \mid a \text{ en } W, \text{ for some } a \in A_v\}$ and $\mathcal{K}_v^- = \mathcal{K}_v \setminus \mathcal{K}_v^+$.

The following result holds (recall the function st from Section 4).

Lemma 23. *Let a be a reaction in S and $W \subseteq S$. Then a* en *W iff $R_a \cup \bar{I}_a \subseteq st(W)$.*

Proof. We have a en W iff both $R_a \subseteq W$ and $I_a \cap W = \varnothing$ iff both $R_a \subseteq W$ and $\bar{I}_a \subseteq \overline{(S \setminus W)}$ iff $R_a \cup \bar{I}_a \subseteq W \cup \overline{(S \setminus W)} = st(W)$. □

The following lemma will be used to transfer the results of Section 4 to reaction systems.

Lemma 24. *Let $\mathcal{A} = (S, A)$ be a reaction system, and $p \in S$. Let T be a tree representing A_p, and let $v \in V$. Then $st(\mathcal{K}_v) = \mathcal{L}_v$, $st(\mathcal{K}_v^+) = \mathcal{L}_v^+$, and $st(\mathcal{K}_v^-) = \mathcal{L}_v^-$.*

Proof. We have $\mathcal{K}_v = \{W \subseteq S \mid a_v \text{ en } W\}$. Hence by Lemma 23, $\mathcal{K}_v = \{W \subseteq S \mid R_v \cup \bar{I}_v \subseteq st(W)\} = \{W \subseteq S \mid \sup(v) \subseteq st(W)\}$. Thus, $st(\mathcal{K}_v) = \{Q \in \text{sspace}_S \mid \sup(v) \subseteq Q\} = \mathcal{L}_v$.

Next, we have $\mathcal{K}_v^+ = \{W \in \mathcal{K}_v \mid a \text{ en } W, \text{ for some } a \in A_v\}$. We obtain similarly, $\mathcal{K}_v^+ = \{W \in \mathcal{K}_v \mid \sup(w) \subseteq st(W) \text{ for some } w \in \text{leav}(T[v])\}$, and so $st(\mathcal{K}_v^+) = \{Q \in \mathcal{L}_v \mid \sup(w) \subseteq Q \text{ for some } w \in \text{leav}(T[v])\} = \mathcal{L}_v^+$.

Finally, $st(\mathcal{K}_v^-) = st(\mathcal{K}_v \setminus \mathcal{K}_v^+)$. As st is a bijection, $st(\mathcal{K}_v \setminus \mathcal{K}_v^+) = st(\mathcal{K}_v) \setminus st(\mathcal{K}_v^+) = \mathcal{L}_v \setminus \mathcal{L}_v^+ = \mathcal{L}_v^-$. □

Given a set of reactions A_p and a vertex v of tree T representing A_p (such that T is optimally selected at v), the following result gives an upper bound on the fraction of states $W \subseteq S$ for which no reaction of A_p is enabled among the states that are "compatible" with v (i.e., the states in $\mathcal{K}_v$). In other words, this result gives an upper bound on the fraction of *dead states* (i.e., the states for which there is no reaction enabled) within the subspace of the state space determined by v.

As a consequence of Lemma 24 we obtain that $|\mathcal{K}_v| = |\mathcal{L}_v|$, $|\mathcal{K}_v^+| = |\mathcal{L}_v^+|$, and $|\mathcal{K}_v^-| = |\mathcal{L}_v^-|$. The following result follows then from Theorem 13.

Theorem 25. *Let $\mathcal{A} = (S, A)$ be a reaction system, and $p \in S$. Let T be a tree representing A_p, and let $v \in V$ such that $O(v)$ is a smallest selection in $\mathcal{F}_T[v]$. Then*

$$\frac{|\mathcal{K}_v^-|}{|\mathcal{K}_v|} \leq \left(1 - \frac{1}{2^{\text{hgt}(v)}}\right)^{\frac{\deg(v)}{\text{hgt}(v)}}.$$

Note that the upper bound in Theorem 25 is determined *only* by the length $\text{hgt}(v)$ and the out-degree $\deg(v)$ of v.

Theorem 25 yields the following corollary.

Corollary 26. *Let $\mathcal{A} = (S, A)$ be a reaction system, and $p \in S$. Let T be a tree representing A_p, and let $v \in V$ such that $O(v)$ is a smallest selection in $\mathcal{F}_T[v]$. Then*

$$\frac{|\mathcal{K}_v^-|}{|\mathcal{K}_v|} \leq e^{-\frac{\deg(v)}{\text{hgt}(v) \cdot 2^{\text{hgt}(v)}}}.$$

Proof. Recall that $\left(1 - \frac{1}{m}\right)^m \leq e^{-1}$ for all positive integers m. By Theorem 25, $\frac{|\mathcal{K}_v^-|}{|\mathcal{K}_v|} \leq \left(1 - \frac{1}{2^{\text{hgt}(v)}}\right)^{\frac{\deg(v)}{\text{hgt}(v)}} = \left(\left(1 - \frac{1}{2^{\text{hgt}(v)}}\right)^{2^{\text{hgt}(v)}}\right)^{\frac{\deg(v)}{\text{hgt}(v) \cdot 2^{\text{hgt}(v)}}} \leq e^{-\frac{\deg(v)}{\text{hgt}(v) \cdot 2^{\text{hgt}(v)}}}$. □

Acknowledgements. We thank Hendrik Jan Hoogeboom and Kai Salomaa for valuable comments on this paper.

References

1. Brijder, R., Ehrenfeucht, A., Main, M.G., Rozenberg, G.: A tour of reaction systems. To appear in Fundamenta Informaticae (2011)
2. Brijder, R., Ehrenfeucht, A., Rozenberg, G.: A note on causalities in reaction systems. Electronic Communications of the EASST 30 (2010)
3. Ehrenfeucht, A., Main, M.G., Rozenberg, G.: Combinatorics of life and death for reaction systems. International Journal of Foundations of Computer Science 21(3), 345–356 (2010)
4. Ehrenfeucht, A., Main, M.G., Rozenberg, G.: Functions defined by reaction systems. International Journal of Foundations of Computer Science 22(1), 167–178 (2011)
5. Ehrenfeucht, A., Rozenberg, G.: Events and modules in reaction systems. Theoretical Computer Science 376(1-2), 3–16 (2007)
6. Ehrenfeucht, A., Rozenberg, G.: Reaction systems. Fundamenta Informaticae 75(1-4), 263–280 (2007)
7. Ehrenfeucht, A., Rozenberg, G.: Introducing time in reaction systems. Theoretical Computer Science 410(4-5), 310–322 (2009)

Derivatives of Regular Expressions and an Application*

Haiming Chen[1] and Sheng Yu[2]

[1] State Key Laboratory of Computer Science
Institute of Software, Chinese Academy of Sciences
Beijing 100190, China
chm@ios.ac.cn
[2] Department of Computer Science
University of Western Ontario, London, Ontario, N6A 5B7, Canada
syu@csd.uwo.ca

Abstract. In this paper, we propose a characterization of the structure of derivatives and prove several new properties of derivatives for regular expressions. The above work can be used to solve an issue in using Berry and Sethi's result, i.e., finding the unique representatives of derivatives. As an application, an improvement of Ilie and Yu's proof of the relation between the partial derivative and Glushkov automata is presented.

1 Introduction

The construction of finite automata from regular expressions is an important issue and has been studied for a long time. Note that finite automata have always been one of Calude's research interests [5]. An elegant construction of deterministic finite automata, based on the derivatives of regular expressions, was proposed by Brzozowski [4]. Among the well-known constructions of ϵ-free nondeterministic finite automata (NFA), the Glushkov automaton was proposed separately by Glushkov [8] and McNaughton and Yamada [10]. Berry and Sethi [2] showed that the Glushkov automaton has a natural connection with the notion of derivative [4], and related the above two different approaches.

The notion of derivative was generalized to partial derivatives by Antimirov [1], which yields the partial derivative automaton, introduced in [1]. Champarnaud and Ziadi [6] proved that the partial derivative automaton is a quotient of the Glushkov automaton. Therefore the partial derivative automaton is smaller than or equal to the Glushkov automaton. The latter has size at most quadratic and can be computed in quadratic time [3,7,11]. They also proposed a quadratic algorithm [6] for computing the partial derivative automata which improved very much the original Antimirov's algorithm. It appears that the partial derivative automaton is among the very small automata converted from a regular expression. Follow automata were introduced in [9]. For a given regular expression E,

* The work of the first author was supported by the National Natural Science Foundation of China under Grants 61070038, 60573013, and that of the second author was supported by the NSERC Discovery Grant 41630.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 343–356, 2012.

the size of the follow automaton constructed from E is at most $\frac{3}{2}|E| + \frac{5}{2}$, which is very close to a lower bound, where $|E|$ is the size of E.

The paper continues the investigation of derivatives along the line of Berry and Sethi. It gives a characterization of the structure of derivatives of an expression E with distinct symbols, showing that each non-null derivative of E is composed of one or more identical expressions (called repeating terms), which implies Berry and Sethi's result [2]. The paper proves several facts, including computation of repeating terms, and several properties of repeating terms. The above work provides new and deeper insight into the nature of derivatives.

Berry and Sethi showed that the derivatives in a certain class of derivatives of an expression E with distinct symbols correspond to the same state of the Glushkov automaton of E. This means that the derivatives that correspond to a state are not unique. In many cases, however, one needs a unique representative for that class of derivatives to correspond to a state. This, however, turns out to be non-trivial as discussed in Section 4. By the work on derivatives in the paper, the representatives can be obtained immediately.

As an application, an improved Ilie and Yu's proof of the relation between the partial derivative and Glushkov automata [9] is presented in this paper.

Section 2 introduces notations and notions required in the paper. Section 3 proposes a characterization of derivatives and several properties of derivatives. Section 4 presents a proof of the relation between the partial derivative and Glushkov automata. Section 5 concludes the paper.

2 Preliminaries

We assume that the reader is familiar with basic regular language and automata theory, e.g., from [12], so that we introduce here only some notations and notions used in the paper.

2.1 Regular Expressions and Finite Automata

Let Σ be an alphabet of symbols. The set of all words over Σ is denoted by Σ^*. The empty word is denoted by ε. A regular expression over Σ is $\emptyset$, ε or a for any $a \in \Sigma$, or is the union $E_1 + E_2$, the concatenation E_1E_2, or the star E_1^* for regular expressions E_1 and E_2. For a regular expression E, the language specified by E is denoted by $L(E)$. The size of E is denoted by $|E|$ and is the length of E when written in postfix (parentheses are not counted). The number of symbol occurrences in E, or the alphabetic width of E, is denoted by $\|E\|$. The symbols that occur in E, which is the smallest alphabet of E, is denoted by Σ_E.

Two regular expressions E_1 and E_2 which reduce to the same expression using associativity, commutativity, and idempotence of $+$ are called *ACI-similar* or *similar* [4], which is denoted by $E_1 \sim_{aci} E_2$.

We assume that the rules $E+\emptyset = \emptyset+E = E$, $E\emptyset = \emptyset E = \emptyset$, and $E\varepsilon = \varepsilon E = E$ ($\emptyset\varepsilon$-rules) hold in the paper.

For a regular expression E over Σ, we define the following sets:

$$first(E) = \{a \mid aw \in L(E), a \in \Sigma, w \in \Sigma^*\},$$
$$last(E) = \{a \mid wa \in L(E), w \in \Sigma^*, a \in \Sigma\},$$
$$follow(E, a) = \{b \mid uabv \in L(E), u, v \in \Sigma^*, b \in \Sigma\}, \text{ for } a \in \Sigma.$$

One can easily write equivalent inductive definitions of the above sets on E, which is omitted here.

For a regular expression we can mark symbols with subscripts so that in the marked expression each marked symbol occurs only once. For example $(a_1 + b_1)^* a_2 b_2 (a_3 + b_3)$ is a marking of the expression $(a + b)^* ab(a + b)$. A marking of an expression E is denoted by $\overline{E}$. If E is a marked expression, then $\overline{E}$ means dropping of subscripts from E. It will be clear from the context whether $\overline{\cdot}$ adds or drops subscripts. We extend the notation for words and automata in the obvious way.

In this way the subscripted symbols are called *positions* of the expression. In the literature, positions are sometimes defined as the subscripts. This definition of positions, however, has drawbacks because it separates subscripts from symbols. When both subscripts and related symbols are required, this presentation is rather awkward. Here we use symbols in $\Sigma_{\overline{E}}$ as the positions, which makes related definitions concise and more flexible (subscripts can be the same, as in the above example).

A finite automaton is a quintuple $M = (Q, \Sigma, \delta, q_0, F)$, where Q is the finite set of states, Σ is the alphabet, $\delta \subseteq Q \times \Sigma \times Q$ is the transition mapping, q_0 is the start state, and $F \subseteq Q$ is the set of accepting states. Denote the language accepted by the automaton M by $L(M)$.

Let $\equiv \subseteq Q \times Q$ be an equivalence relation. We say that $\equiv$ is right invariant w.r.t. M iff (1) $\equiv \subseteq (Q - F)^2 \cup F^2$ and (2) for any $p, q \in Q, a \in \Sigma$, if $p \equiv q$, then $p_1 \equiv q_1$ for $p_1 \in \delta(p, a), q_1 \in \delta(q, a)$. If $\equiv$ is right invariant, then we can define a quotient automaton $M/_{\equiv}$ in the usual way. One can prove that $L(M/_{\equiv}) = L(M)$.

2.2 Derivatives

Given a language L and a finite word w, the derivative (or left quotient set) of L w.r.t. w is $w^{-1}(L) = \{u \mid wu \in L\}$.

Derivatives of regular expressions were introduced by Brzozowski [4].

Definition 1. (Brzozowski [4]) *Given a regular expression E and a symbol a, the derivative of E with respect to a, $a^{-1}(E)$, is defined inductively as follows:*

$$\begin{aligned}
a^{-1}(\emptyset) &= a^{-1}(\varepsilon) = \emptyset \\
a^{-1}(b) &= \begin{cases} \varepsilon, \text{ if } b = a \\ \emptyset, \text{ otherwise} \end{cases} \\
a^{-1}(F + G) &= a^{-1}(F) + a^{-1}(G) \\
a^{-1}(FG) &= \begin{cases} a^{-1}(F)G + a^{-1}(G), & \text{if } \varepsilon \in L(F) \\ a^{-1}(F)G, & \text{otherwise} \end{cases} \\
a^{-1}(F^*) &= a^{-1}(F)F^*
\end{aligned}$$

Derivative with respective to a word is computed by $\varepsilon^{-1}(E) = E$, $(wa)^{-1}(E) = a^{-1}(w^{-1}(E))$.

It is known that $L(w^{-1}(E)) = w^{-1}(L(E))$. Brzozowski showed that an expression E has a finite number of dissimilar derivatives [4], which were used as states to construct a deterministic finite automaton of E.

Partial derivatives were introduced by Antimirov [1].

Definition 2. (Antimirov [1]) *Given a regular expression* E *and a symbol* a*, the set of partial derivatives of* E *with respect to* a*,* $\partial_a(E)$*, is defined as follows*[1]*:*

$$\begin{aligned}
\partial_a(\emptyset) &= \partial_a(\varepsilon) = \emptyset \\
\partial_a(b) &= \begin{cases} \{\varepsilon\}, \text{ if } b = a \\ \emptyset, \quad \text{otherwise} \end{cases} \\
\partial_a(F+G) &= \partial_a(F) \cup \partial_a(G) \\
\partial_a(FG) &= \begin{cases} \partial_a(F)G \cup \partial_a(G), \text{ if } \varepsilon \in L(F) \\ \partial_a(F)G, \qquad\quad \text{otherwise} \end{cases} \\
\partial_a(F^*) &= \partial_a(F)F^*
\end{aligned}$$

Partial derivative with respect to a word is computed by $\partial_\varepsilon(E) = \{E\}$, $\partial_{wa}(E) = \bigcup_{p \in \partial_w(E)} \partial_a(p)$*. The language denoted by* $\partial_w(E)$ *is*

$$L(\partial_w(E)) = \bigcup_{p \in \partial_w(E)} L(p).$$

It is proved in [1] that the cardinality of the set $PD(E) = \cup_{w \in \Sigma^*} \partial_w(E)$ of all partial derivatives of a regular expression E is less than or equal to $\|E\| + 1$.

2.3 Glushkov and Partial Derivative Automata

The Glushkov or position automaton was introduced independently by Glushkov [8] and McNaughton and Yamada [10].

Definition 3. *The Glushkov automaton of* E *is*

$$M_{\mathrm{g}}(E) = (Q_{\mathrm{g}}, \Sigma, \delta_{\mathrm{g}}, q_E, F_{\mathrm{g}}),$$

where

1. $Q_{\mathrm{g}} = \Sigma_{\overline{E}} \cup \{q_E\}$*,* q_E *is a new state not in* $\Sigma_{\overline{E}}$
2. $\delta_{\mathrm{g}}(q_E, a) = \{x \mid x \in first(\overline{E}), \overline{x} = a\}$ *for* $a \in \Sigma$
3. $\delta_{\mathrm{g}}(x, a) = \{y \mid y \in follow(\overline{E}, x), \overline{y} = a\}$ *for* $x \in \Sigma_{\overline{E}}$ *and* $a \in \Sigma$
4. $F_{\mathrm{g}} = \begin{cases} last(\overline{E}) \cup \{q_E\}, \text{ if } \varepsilon \in L(E), \\ last(\overline{E}), \qquad\quad \text{otherwise} \end{cases}$

[1] In the definition $RF = \{EF | E \in R\}$ for a set R of regular expressions and a regular expression F.

As shown by Glushkov [8] and McNaughton and Yamada [10], $L(M_{\mathrm{g}}(E)) = L(E)$. $M_{\mathrm{g}}(E)$ can be computed in quadratic time [3,7,11].

The partial derivative or equation automaton [1] is constructed by partial derivatives.

Definition 4. *The partial derivative automaton of a regular expression E is*

$$M_{\mathrm{pd}}(E) = (PD(E), \Sigma, \delta_{\mathrm{pd}}, E, \{q \in PD(E) \mid \varepsilon \in L(q)\}),$$

where $\delta_{\mathrm{pd}}(q, a) = \partial_a(q)$, for any $q \in PD(E), a \in \Sigma$.

Note that $PD(E)$ has been defined in the previous subsection. It is proved [6] that $M_{\mathrm{pd}}(E)$ is a quotient of $M_{\mathrm{g}}(E)$.

3 Regular Expressions with Distinct Symbols

From Brzozowski [4] and Berry and Sethi [2] the following two facts are easily derived.

Proposition 1. *Let all symbols in E be distinct. Given $a \in \Sigma_E$, for all words w,*

1. If $E = E_1 + E_2$, then

$$(wa)^{-1}(E_1 + E_2) = \begin{cases} (wa)^{-1}(E_1) & \text{if } a \in \Sigma_{E_1}, w \in \Sigma_{E_1}^* \\ (wx)^{-1}(E_2) & \text{if } a \in \Sigma_{E_2}, w \in \Sigma_{E_2}^* \\ \emptyset & \text{otherwise} \end{cases} \tag{1}$$

2. If $E = E_1 E_2$, then

$$(wa)^{-1}(E_1 E_2) = \begin{cases} (wa)^{-1}(E_1)E_2 & \text{if } a \in \Sigma_{E_1}, w \in \Sigma_{E_1}^* \\ (va)^{-1}(E_2) & \text{if } w = uv, \varepsilon \in L(u^{-1}(E_1)), a \in \Sigma_{E_2}, \\ & u \in \Sigma_{E_1}^*, v \in \Sigma_{E_2}^* \\ \emptyset & \text{otherwise} \end{cases} \tag{2}$$

Proof. 1. It is directly from Berry and Sethi [2].

2. From Berry and Sethi [2] it is already known

$$(wa)^{-1}(E_1 E_2) = \begin{cases} (wa)^{-1}(E_1)E_2 & \text{if } a \in \Sigma_{E_1}, w \in \Sigma_{E_1}^* \text{ (a)} \\ \Sigma_{w=uv, \varepsilon \in L(u^{-1}(E_1))}(va)^{-1}(E_2) & \text{otherwise (b)} \end{cases}$$

Let us consider (b) and set $wa = a_1 a_2 \ldots a_t$. For a concrete sequence of $a_1 \ldots a_t$, a subterm $(a_{r+1} \ldots a_t)^{-1}(E_2)$ in (b) can exist only if $a_1, \ldots, a_r$ in E_1 and $a_{r+1}, \ldots, a_t$ in E_2. Since $a_n, 1 \leq n \leq t$ is either in E_1 or in E_2, there is at most one such subterm in (b). If such condition is not satisfied, then $(wa)^{-1}(E_1 E_2) = \emptyset$.

Proposition 2. *Given $a \in \Sigma_E$, for all words w, $(wa)^{-1}(E^*)$ is equivalent to a sum of subterms chosen from the set $\{(va)^{-1}(E)E^* \mid wa = uva\}$.*

Proof. It is directly from Brzozowski [4] or Berry and Sethi [2].

Berry and Sethi [2] proved that

Proposition 3. (Berry and Sethi [2]) *Let all symbols in E be distinct. Given a fixed $a \in \Sigma_E$, $(wa)^{-1}(E)$ is either $\emptyset$ or unique modulo $\sim_{aci}$ for all words w.*

This is a very important property which was used to connect the class of non-null $(wa)^{-1}(\overline{E})$ to the state a of $M_{\rm g}(E)$ for an expression E.

We further investigate the structure of non-null $(wa)^{-1}(E)$ here.

Theorem 1. *Let all symbols in E be distinct. Given a fixed $a \in \Sigma_E$, for all words w, each non-null $(wa)^{-1}(E)$ must be of one of the following forms: F or $F + \ldots + F$, where F is a non-null regular expression, called the repeating term of $(wa)^{-1}(E)$, which does not contain $+$ at the top level.*

Proof. We prove it by induction on the structure of E. If $E = \emptyset$ or ε, then no symbol is in E, and no non-null derivative exists. Thus no repeating term exists. If $E = b, b \in \Sigma_E$, then $a^{-1}(b) = \varepsilon$ for $a = b$, $(wa)^{-1}(E) = \emptyset$ for $w \neq \varepsilon$ or $a \neq b$. Thus ε is the repeating term of $a^{-1}(a)$, in which no $+$ appears.

1. $E = E_1 + E_2$. By equation (1), a non-null $(wa)^{-1}(E)$ is either $(wa)^{-1}(E_1)$ or $(wa)^{-1}(E_2)$. Suppose the first, then $(wa)^{-1}(E_1)$ is non-null, and the inductive hypothesis applies to it. The repeating term of $(wa)^{-1}(E)$ is the same as $(wa)^{-1}(E_1)$, and no top-level $+$ will be added. The same is for the second.

2. $E = E_1E_2$. By equation (2), a non-null $(wa)^{-1}(E)$ is either $(wa)^{-1}(E_1)E_2$ or $(va)^{-1}(E_2)$ for some v such that $w = uv$. If $(wa)^{-1}(E) = (wa)^{-1}(E_1)E_2$, by the inductive hypothesis, $(wa)^{-1}(E_1)$ is F or $F + \ldots F$ where F does not contain $+$ at the top level. Then FE_2 is the repeating term of $(wa)^{-1}(E)$, which does not contain top-level $+$. If $(wa)^{-1}(E) = (wa)^{-1}(E_2)$, the proof is the same as in the above case 1.

3. $E = E_1^*$. From Proposition 2 it is known that $(wa)^{-1}(E)$ is the sum of subterms of the form $(va)^{-1}(E_1)E_1^*$ where $wa = uva$. From the inductive hypothesis, each non-null $(va)^{-1}(E_1)$ is F or $F + \ldots + F$ where F does not contain $+$ at the top level, so $(va)^{-1}(E_1)E_1^*$ is FE_1^* or $FE_1^* + \ldots + FE_1^*$. If $(wa)^{-1}(E)$ is non-null, it is a sum of one or more FE_1^*, which does not contain $+$ at the top level.

Therefore each $(wa)^{-1}(E)$ is either $\emptyset$ or a sum of one or more repeating terms of $(wa)^{-1}(E)$.

In the following examples the expression is taken from [9].

Example 1. Let $E = (a + b)(a^* + ba^* + b^*)^*$, then
$\overline{E} = (a_1 + b_2)(a_3^* + b_4a_5^* + b_6^*)^*$,
$a_1^{-1}(\overline{E}) = (a_3^* + b_4a_5^* + b_6^*)^* = \tau_1$,
$(a_1a_3)^{-1}(\overline{E}) = a_3^{-1}(\tau_1) = a_3^*\tau_1 = \tau_2$,
$(a_1a_3a_3)^{-1}(\overline{E}) = a_3^{-1}(\tau_2) = \tau_2 + \tau_2$,
$\ldots$
The repeating term for $(wa_1)^{-1}(\overline{E})$ is τ_1, the repeating term for $(wa_3)^{-1}(\overline{E})$ is τ_2.

Denote by $rt_a(E)$ the repeating term of $(wa)^{-1}(E)$. From Theorem 1 we have

Corollary 1. *Let all symbols in E be distinct. If $(wa)^{-1}(E)$ is non-null, then $(wa)^{-1}(E) \sim_{aci} rt_a(E)$.*

Corollary 1 is a more precise version of Berry and Sethi's result (i. e., Proposition 3), that is, Theorem 1 implies Berry and Sethi's result, but not vice versa.

Below we consider the question: *For each $a \in \Sigma_E$, whether there is a non-null $(wa)^{-1}(E)$ containing one and only one $rt_a(E)$, that is, $rt_a(E)$ is a derivative of E.* The answer is positive. We show it by a construction, the first appearance.

Let all symbols in E be distinct. We associate symbols in Σ_E with an order. This is achieved by setting up a one-to-one function $ind : \Sigma_E \to \{1, \ldots, \|E\|\}$: $ind(a) = d$ if a is the dth occurrence of symbols from left to right in E (Note that each symbol in E occurs only once). For $a, b \in \Sigma_E$, define $a < b$ iff $ind(a) < ind(b)$. For any words $w_1, w_2 \in \Sigma_E^*$, define the graded lexicographical order by $w_1 \prec w_2$ if either $|w_1| < |w_2|$, or $|w_1| = |w_2|$ and the condition is satisfied: let $w_1 = a_1 \ldots a_n, w_2 = a'_1 \ldots a'_n$, there exists an integer $k, 1 \leq k \leq n$, such that $a_t = a'_t$ for $t = 1, \ldots, k-1$, and $a_k < a'_k$. A non-null $(wa)^{-1}(E)$ is called the *first appearance* of derivative of E w.r.t. a, denoted by $F_a(E)$, if for any other non-null $(w_1a)^{-1}(E)$ it has $w \prec w_1$. From Berry and Sethi [2] a non-null $(wa)^{-1}(E)$ exists for all $a \in \Sigma_E$, which ensures the existence of $F_a(E)$.

Example 2. For $E = (a + b)(a^* + ba^* + b^*)^*$, $\overline{E} = (a_1 + b_2)(a_3^* + b_4a_5^* + b_6^*)^*$. The first appearances of derivatives w.r.t. symbols in $\overline{E}$, in which the symbols are underlined, are computed as follows.

$$
\begin{array}{ll}
\underline{a_1}^{-1}(\overline{E}) = (a_3^* + b_4a_5^* + b_6^*)^* = \tau_1, & \underline{b_2}^{-1}(\overline{E}) = (a_3^* + b_4a_5^* + b_6^*)^* = \tau_1, \\
(a_1\underline{a_3})^{-1}(\overline{E}) = a_3^{-1}(\tau_1) = a_3^*\tau_1 = \tau_2, & (a_1\underline{b_4})^{-1}(\overline{E}) = b_4^{-1}(\tau_1) = a_5^*\tau_1 = \tau_3, \\
(a_1\underline{b_6})^{-1}(\overline{E}) = b_6^{-1}(\tau_1) = b_6^*\tau_1 = \tau_4, & (b_2a_3)^{-1}(\overline{E}) = a_3^{-1}(\tau_1) = \tau_2, \\
(b_2b_4)^{-1}(\overline{E}) = b_4^{-1}(\tau_1) = \tau_3, & (b_2b_6)^{-1}(\overline{E}) = b_6^{-1}(\tau_1) = \tau_4, \\
(a_1a_3a_3)^{-1}(\overline{E}) = a_3^{-1}(\tau_2) = \tau_2 + \tau_2, & (a_1a_3b_4)^{-1}(\overline{E}) = b_4^{-1}(\tau_2) = \tau_3, \\
(a_1a_3b_6)^{-1}(\overline{E}) = b_6^{-1}(\tau_2) = \tau_4, & (a_1b_4a_3)^{-1}(\overline{E}) = a_3^{-1}(\tau_3) = \tau_2, \\
(a_1b_4b_4)^{-1}(\overline{E}) = b_4^{-1}(\tau_3) = \tau_3, & (a_1b_4\underline{a_5})^{-1}(\overline{E}) = a_5^{-1}(\tau_3) = \tau_3.
\end{array}
$$

From Example 2 we can see that no first appearance has duplicated repeating terms while other derivatives may have. Generally we have

Proposition 4. *Let all symbols in E be distinct. Given a fixed $a \in \Sigma_E$, the first appearance $F_a(E)$ consists of only one repeating term.*

Proof. We prove it by induction on the structure of E. The cases for $E = \varepsilon, \emptyset, b$, $b \in \Sigma_E$ are obvious. Suppose wa is chosen such that $F_a(E)$ is $(wa)^{-1}(E)$.

1. $E = E_1 + E_2$. Consider equation (1). If $(wa)^{-1}(E) = (wa)^{-1}(E_1)$, we show that $F_a(E_1)$ is $(wa)^{-1}(E_1)$. If this is not true, there is a word $w_1 \prec w$ such that $(w_1a)^{-1}(E_1) \neq \emptyset$. So $(w_1a)^{-1}(E) \neq \emptyset$, which is a contradiction. Therefore $(wa)^{-1}(E_1)$ is the first appearance and the inductive hypothesis applies to it. The same is for $(wa)^{-1}(E) = (wa)^{-1}(E_2)$.

2. $E = E_1E_2$. Consider equation (2). If $(wa)^{-1}(E) = (wa)^{-1}(E_1)E_2$, similarly as above we can prove that $(wa)^{-1}(E_1)$ is the first appearance, and the inductive hypothesis applies to it. If $(wa)^{-1}(E) = (v_1a)^{-1}(E_2)$ for some v_1 such that $wa = uv_1a$, we show that this subterm is $F_a(E_2)$. Suppose the converse. Then there is a word $v \prec v_1$ such that $(va)^{-1}(E_2) \neq \emptyset$. So it is easy to see that $(uva)^{-1}(E) \neq \emptyset$. But $uva \prec wa$, which is a contradiction. Therefore $(v_1a)^{-1}(E_2)$ is the first appearance and the inductive hypothesis applies to it.

3. $E = E_1^*$. From Proposition 2 $(wa)^{-1}(E)$ is the sum of subterms of the form $(va)^{-1}(E_1)E_1^*$ where $wa = uva$. We show that when $(wa)^{-1}(E)$ is $F_a(E)$ the above becomes $(wa)^{-1}(E) = (wa)^{-1}(E_1)E_1^*$. Suppose $(wa)^{-1}(E)$ contains another non-null subterm $(va)^{-1}(E_1)E_1^*, w = uv, w \neq v$. Then $(va)^{-1}(E)$ is not $\emptyset$ since it contains $(va)^{-1}(E_1)E_1^*$ as a summand. However $v \prec w$, which is a contradiction. Similarly we can prove that $(wa)^{-1}(E_1)$ is $F_a(E_1)$, so the inductive hypothesis applies to it.

The choice of the order is not significant. Actually for different *ind* the resulting $F_a(E)$ is the same.

Proposition 5. *Let all symbols in E be distinct. Given any words $w_1, w_2 \in \Sigma_E^*$ and $a \in \Sigma_E$, if $|w_1| = |w_2|$ and $(w_1a)^{-1}(E), (w_2a)^{-1}(E) \neq \emptyset$, and there is no $w \in \Sigma_E^*$, such that $|w| < |w_1|$ and $(wa)^{-1}(E) \neq \emptyset$, then $(w_1a)^{-1}(E) = (w_2a)^{-1}(E)$.*

Proof. We prove it by induction on the structure of E. If $E = \emptyset$ or ε, no non-null derivative exists. If $E = b$ for a symbol b, the only non-null derivative is ε, in which case $w_1 = w_2 = \varepsilon$ and $a = b$. So $(w_1a)^{-1}(E) = (w_2a)^{-1}(E)$.

1. $E = E_1 + E_2$. If $a \in \Sigma_{E_1}$, from equation (1), we have $(w_1a)^{-1}(E) = (w_1a)^{-1}(E_1)$ and $(w_2a)^{-1}(E) = (w_2a)^{-1}(E_1)$. We can see that there is no w, such that $|w| < |w_1|$ and $(wa)^{-1}(E_1) \neq \emptyset$. Otherwise $(wa)^{-1}(E) = (wa)^{-1}(E_1) \neq \emptyset$ which is a contradiction. So the inductive hypothesis applies to E_1. The proof is the same for $a \in \Sigma_{E_2}$.

2. $E = E_1E_2$. If $a \in \Sigma_{E_1}$, from equation (2), we have $(w_1a)^{-1}(E) = (w_1a)^{-1}(E_1)E_2$ and $(w_2a)^{-1}(E) = (w_2a)^{-1}(E_1)E_2$. Similar as in case 1 we can prove $(w_1a)^{-1}(E_1) = (w_2a)^{-1}(E_1)$. Thus $(w_1a)^{-1}(E) = (w_2a)^{-1}(E)$.

If $a \in \Sigma_{E_2}$, from equation (2), we have $(w_1a)^{-1}(E) = (v_1a)^{-1}(E_2)$ and $(w_2a)^{-1}(E) = (v_2a)^{-1}(E_2)$ for some v_1, v_2 such that $w_1 = u_1v_1, w_2 = u_2v_2, \varepsilon \in L(u_1^{-1}(E_1)), \varepsilon \in L(u_2^{-1}(E_1)), u_1, u_2 \in \Sigma_{E_1}^*, v_1, v_2 \in \Sigma_{E_2}^*$. We show $|v_1| = |v_2|$. Suppose the converse. Without losing generality suppose $|v_1| < |v_2|$. Notice $|w_1| = |w_2|$, then $|u_1| > |u_2|$. Since $\varepsilon \in L(u_2^{-1}(E_1))$, $u_2 \in \Sigma_{E_1}^*$, and $v_1 \in \Sigma_{E_2}^*$, by equation (2) $(u_2v_1a)^{-1}(E) = (v_1a)^{-1}(E_2) \neq \emptyset$. But $|u_2v_1| < |w_1|$ which is a contradiction. So $|v_1| = |v_2|$. Similarly we can prove there is no $v \in \Sigma_{E_2}^*$, such that $|v| < |v_1|$ and $(va)^{-1}(E_2) \neq \emptyset$. So the inductive hypothesis applies to E_2.

3. $E = E_1^*$. Similar as the proof of case 3 in the proof of Proposition 4, we can prove $(w_1a)^{-1}(E) = (w_1a)^{-1}(E_1)E_1^*$ and $(w_2a)^{-1}(E) = (w_2a)^{-1}(E_1)E_1^*$ and there is no $w \in \Sigma_{E_1}^*$ such that $|w| < |w_1|$ and $(wa)^{-1}(E_1) \neq \emptyset$. Then by the inductive hypothesis $(w_1a)^{-1}(E_1) = (w_2a)^{-1}(E_1)$, thus $(w_1a)^{-1}(E) = (w_2a)^{-1}(E)$.

In the above proposition we easily see that $(w_1a)^{-1}(E) = (w_2a)^{-1}(E) = F_a(E)$. Therefore $F_a(E)$ is the same for varying *ind*.

Then

Proposition 6. *Let all symbols in E be distinct. There exists a word $w \in \Sigma_E^*$ for each $a \in \Sigma_E$, such that $(wa)^{-1}(E) = rt_a(E)$.*

Proof. The first appearance $F_a(E)$ is one such $(wa)^{-1}(E)$ satisfying $F_a(E) = rt_a(E)$.

Thus repeating terms are derivatives of E, and any non-null derivative of E is built from one of them. Next we present other properties for $rt_a(E)$.

Proposition 7. *Let all symbols in E be distinct. For each $a \in \Sigma_E$,*
(1) $rt_a(E)$ exists,
(2) $rt_a(E)$ is unique.

Proof. (1) From Berry and Sethi [2] it is known that a non-null $(wa)^{-1}(E)$ exists for each $a \in \Sigma_E$. Then from Theorem 1 $rt_a(E)$ exists and $rt_a(E) \neq \emptyset$.

(2) Suppose $rt_a(E)$ is not unique. That is, for some $a \in \Sigma_E$, there are two repeating terms F and F_1, such that $F \neq F_1$. From Theorem 1 and Proposition 6 it implies $F = F_1 + \ldots + F_1$ and $F_1 = F + \ldots + F$, which is a contradiction. Therefore $rt_a(E)$ is unique.

If $E = \emptyset$ or ε, no symbol is in E, so $rt_a(E)$ is undefined. We let $rt_a(\emptyset) = rt_a(\varepsilon) = \emptyset$ for any $a \in \Sigma$ for the sake of completeness.

The following lemma will be used in the proof of Proposition 8.

Lemma 1. *Let all symbols in E be distinct. If $(wa)^{-1}(E) \sim_{aci} E$ for some $w \in \Sigma_E^*$, then $rt_a(E) = E$.*

Proof. We prove by induction on the structure of E. If $E = \emptyset$, then $(wa)^{-1}(E) = \emptyset$. By assumption, $rt_a(E) = \emptyset$. So $rt_a(E) = E$. If $E = \varepsilon$, then $(wa)^{-1}(E) = \emptyset$, $(wa)^{-1}(E) \not\sim_{aci} E$. If $E = a$, then $a^{-1}(E) = \varepsilon$, $(wb)^{-1}(E) = \emptyset$ for $w \neq \varepsilon$ or $b \neq a$. So $(wa)^{-1}(E) \not\sim_{aci} E$.

By induction: 1. $E = F + G$. By the rules ($\emptyset\varepsilon$-rules), $F, G \neq \emptyset$, then $(wa)^{-1}(E) \sim_{aci} E \neq \emptyset$. By equation (1), $(wa)^{-1}(E)$ is either $(wa)^{-1}(F)$ or $(wa)^{-1}(G)$. If $(wa)^{-1}(E) = (wa)^{-1}(F)$, then $(wa)^{-1}(F) \sim_{aci} F + G$. Since $(wa)^{-1}(F)$ does not contain symbols in G, we have $G = \emptyset$, which is a contradiction. Similarly, if $(wa)^{-1}(E) = (wa)^{-1}(G)$, we also have a contradiction.

2. $E = FG$. By the rules ($\emptyset\varepsilon$-rules), $F, G \neq \emptyset$ or ε, then since $(wa)^{-1}(E) \sim_{aci} E \neq \emptyset$, by equation (2) $wa^{-1}(E)$ is either $(wa)^{-1}(F)G$ or $(va)^{-1}(G)$ for some v such that $w = uv$. If $wa^{-1}(E) = (wa)^{-1}(F)G$, then $(wa)^{-1}(F)G \sim_{aci} FG$. So $(wa)^{-1}(F) \sim_{aci} F$. By the inductive hypothesis, we have $rt_a(F) = F$. By equation (2) $wa^{-1}(E) = (wa)^{-1}(F)G$ implies $a \in \Sigma_F$. Hence, from the proof of Theorem 1 we know $rt_a(E) = rt_a(F)G = FG$. If $wa^{-1}(E) = (va)^{-1}(G)$, then $(va)^{-1}(G) \sim_{aci} FG$. Since $(va)^{-1}(G)$ does not contain symbols in F, we have $F = \varepsilon$, which is a contradiction.

3. $E = F^*$. If $E = \emptyset$, then $rt_a(E) = E$. Otherwise $E \neq \emptyset$, then $(wa)^{-1}(E) \neq \emptyset$. From the proof of Theorem 1 we have $rt_a(E) = rt_a(F)F^*$. Thus $(wa)^{-1}(E)$ is a sum of one or more $rt_a(F)F^*$. Since $(wa)^{-1}(E) \sim_{aci} F^*$, we have $rt_a(F) = \varepsilon$. Hence $rt_a(E) = rt_a(F)F^* = F^* = E$.

This means if $w^{-1}(E) \sim_{aci} E$ then E does not contain any top-level $+$, or, equivalently, if E contains any top-level $+$, then $w^{-1}(E) \not\sim_{aci} E$ for any $w \in \Sigma_E^*$.

Remark. If the rules ($\emptyset\varepsilon$-rules) do not hold, the above lemma can also be proved without difficulty.

Proposition 8. *Let all symbols in E be distinct. If there are non-null $(w_1a_1)^{-1}(E)$ and $(w_2a_2)^{-1}(E)$, such that $(w_1a_1)^{-1}(E) \sim_{aci} (w_2a_2)^{-1}(E)$, then $rt_{a_1}(E) = rt_{a_2}(E)$, and vice versa.*

Proof. ($\Rightarrow$) We prove it by induction on the structure of E. The cases for $E = \varepsilon, \emptyset, a$, $a \in \Sigma_E$ are obvious.

1. $E = F+G$. From equation (1), the non-null $(w_1a_1)^{-1}(E)$ is either $(w_1a_1)^{-1}(F)$ or $(w_1a_1)^{-1}(G)$. Likewise, the non-null $(w_2a_2)^{-1}(E)$ is either $(w_2a_2)^{-1}(F)$ or $(w_2a_2)^{-1}(G)$.

If

$$(w_1a_1)^{-1}(E) = (w_1a_1)^{-1}(F), (w_2a_2)^{-1}(E) = (w_2a_2)^{-1}(F) \quad (a),$$

then $(w_1a_1)^{-1}(F) \sim_{aci} (w_2a_2)^{-1}(F)$. By the inductive hypothesis, we have $rt_{a_1}(F) = rt_{a_2}(F)$. In addition, (a) implies $a_1, a_2 \in \Sigma_F$. Then from the proof of Theorem 1 we know $rt_{a_1}(E) = rt_{a_1}(F)$, and $rt_{a_2}(E) = rt_{a_2}(F)$. Hence $rt_{a_1}(E) = rt_{a_2}(E)$.

If

$$(w_1a_1)^{-1}(E) = (w_1a_1)^{-1}(F), (w_2a_2)^{-1}(E) = (w_2a_2)^{-1}(G) \quad (b),$$

then $(w_1a_1)^{-1}(F) \sim_{aci} (w_2a_2)^{-1}(G)$. Since symbols in F and G are distinct, we have $(w_1a_1)^{-1}(F) = (w_2a_2)^{-1}(G) = \varepsilon$. Then from Theorem 1 we have $rt_{a_1}(F) = rt_{a_2}(G) = \varepsilon$. In addition, (b) implies $a_1 \in \Sigma_F$ and $a_2 \in \Sigma_G$. Hence similarly from the proof of Theorem 1 we know $rt_{a_1}(E) = rt_{a_1}(F)$ and $rt_{a_2}(E) = rt_{a_2}(G)$, So $rt_{a_1}(E) = rt_{a_2}(E)$.

Proofs for the remaining two cases are similar to the above cases.

2. $E = FG$. From equation (2), the non-null $(w_1a_1)^{-1}(E)$ is either $(w_1a_1)^{-1}(F)G$ or $(v_1a_1)^{-1}(G)$ for some v_1 such that $w_1 = u_1v_1$. Likewise, the non-null $(w_2a_2)^{-1}(E)$ is either $(w_2a_2)^{-1}(F)G$ or $(v_2a_2)^{-1}(G)$.

If

$$(w_1a_1)^{-1}(E) = (w_1a_1)^{-1}(F)G, (w_2a_2)^{-1}(E) = (w_2a_2)^{-1}(F)G \quad (a),$$

then $(w_1a_1)^{-1}(F)G \sim_{aci} (w_2a_2)^{-1}(F)G$, which then implies $(w_1a_1)^{-1}(F) \sim_{aci} (w_2a_2)^{-1}(F)$. By the inductive hypothesis, we have $rt_{a_1}(F) = rt_{a_2}(F)$. In addition, (a) implies $a_1, a_2 \in \Sigma_F$. Then from the proof of Theorem 1 we know $rt_{a_1}(E) = rt_{a_1}(F)G$, and $rt_{a_2}(E) = rt_{a_2}(F)G$. Hence $rt_{a_1}(E) = rt_{a_2}(E)$.

If

$$(w_1a_1)^{-1}(E) = (w_1a_1)^{-1}(F)G, (w_2a_2)^{-1}(E) = (v_2a_2)^{-1}(G) \quad (b),$$

then $(w_1a_1)^{-1}(F)G \sim_{aci} (v_2a_2)^{-1}(G)$. Since $(v_2a_2)^{-1}(G)$ does not contain symbols in F, we have $(w_1a_1)^{-1}(F) = \varepsilon$, and $G \sim_{aci} (v_2a_2)^{-1}(G)$. Since $(w_1a_1)^{-1}(F) = \varepsilon$, from Theorem 1 we have $rt_{a_1}(F) = \varepsilon$. By Lemma 1 $G \sim_{aci} (v_2a_2)^{-1}(G)$ implies $rt_{a_2}(G) = G$. In addition, (b) implies $a_1 \in \Sigma_F$ and $a_2 \in \Sigma_G$. Hence $rt_{a_1}(E) = rt_{a_1}(F)G = G = rt_{a_2}(G) = rt_{a_2}(E)$.

Proofs for the remaining two cases are similar to the above cases.

3. $E = F^*$. Since $(w_1a_1)^{-1}(E) \sim_{aci} (w_2a_2)^{-1}(E)$, by Corollary 1 we have $rt_{a_1}(E) \sim_{aci} rt_{a_2}(E)$. From the proof of Theorem 1 we know

$$rt_{a_1}(E) = rt_{a_1}(F)F^*, rt_{a_2}(E) = rt_{a_2}(F)F^* \quad .$$

So $rt_{a_1}(F) \sim_{aci} rt_{a_2}(F)$, which implies there are $(u_1a_1)^{-1}(F), (u_2a_2)^{-1}(F) \neq \emptyset$, such that $(u_1a_1)^{-1}(F) \sim_{aci} (u_2a_2)^{-1}(F)$. Then from the inductive hypothesis, we have $rt_{a_1}(F) = rt_{a_2}(F)$. Hence $rt_{a_1}(E) = rt_{a_1}(F)F^* = rt_{a_2}(E)$.

($\Leftarrow$) This is obvious from Corollary 1.

Corollary 2. *Let all symbols in E be distinct. If $rt_{a_1}(E) \sim_{aci} rt_{a_2}(E)$, then $rt_{a_1}(E) = rt_{a_2}(E)$.*

Remark 1. From the previous discussions, it is clear that $rt_a(E)$'s are 'atomic' building blocks, in the following meanings. (1) Each non-null $(wa)^{-1}(E)$ is uniquely decomposed into a sum of $rt_a(E)$, that is, $(wa)^{-1}(E) = \Sigma\, rt_a(E)$. (2) $rt_a(E)$ and $rt_b(E)$ are either identical, or not equivalent modulo $\sim_{aci}$, if $a \neq b$.

4 An Application

The above results solve an issue in using Berry and Sethi's result. Berry and Sethi showed that an arbitrary derivative in the class $\{(wx)^{-1}(\overline{E}) \mid w \in \Sigma^*_{\overline{E}}\}$ corresponds to the state x of the Glushkov automaton of E. This means that the derivatives that correspond to a state are not unique. In many cases, however, one needs a unique representative for that class of derivatives to correspond to a state. This, however, turns out to be non-trivial as is discussed later in this section. By the theoretical work on derivatives in the paper, the representatives can be obtained immediately. As an application this section gives an improvement of Ilie and Yu's proof [9].

4.1 Background

Champarnaud and Ziadi's proof of the fact that the partial derivative automaton is a quotient of the Glushkov automaton resorts to their newly defined notion of c-derivative [6]. It is thus an interesting and attractive issue whether a proof can

directly use only the notions of derivative and partial derivative. Ilie and Yu [9] presented such a proof, which is much simplified compared with Champarnaud and Ziadi's proof. The central issue to use Ilie and Yu's approach is to find a unique representative for a class of derivatives mentioned above. As we show shortly, the proof in [9] actually fails to find the correct representatives. See next subsection for details. The difficulty of finding the representatives may also be partly reflected by the fact that the first proof (Champarnaud and Ziadi [6]) has to use an indirect approach.

Since a correct proof directly using only derivatives and partial derivatives may provide insight and helpful techniques for related researches, for example research of algorithms for partial derivative automata, it is valuable to give such a proof.

In the following, based on our work on derivatives, and in the spirit of Ilie and Yu, an improved proof which directly uses only the notions of derivative and partial derivative is presented.

4.2 Ilie and Yu's Proof

It is claimed in the proof [9] that, by using the rules ($\emptyset\varepsilon$-rules), for a fixed $x \in \Sigma_{\overline{E}}$ and for all words w, $(wx)^{-1}(\overline{E})$ is either $\emptyset$ or unique. However this result is incorrect, which can be seen from the following example.

Example 3. In Example 1, $(a_1a_3)^{-1}(\overline{E})$ and $(a_1a_3a_3)^{-1}(\overline{E})$ are distinct.

The whole proof is based on this uniqueness assumption.

4.3 An Improved Proof

From Theorem 1 and the definitions of derivatives and partial derivatives it is easy to see that for an expression E if $\partial_{ux}(\overline{E}) \neq \emptyset$ then $\partial_{ux}(\overline{E}) = \{rt_x(\overline{E})\}$.

For a letter $x \in \Sigma_{\overline{E}}$, recall that Berry and Sethi's continuation, denoted $C_x(\overline{E})$, is any expression $(wx)^{-1}(\overline{E}) \neq \emptyset$. We use $rt_x(\overline{E})$ instead of arbitrary $(wx)^{-1}(\overline{E}) \neq \emptyset$ to represent $C_x(\overline{E})$, i. e., $C_x(\overline{E}) = rt_x(\overline{E})$. Now the continuation $C_x(\overline{E})$ is unique. Denote also $C_{q_E}(\overline{E}) = \overline{E}$ (q_E is the start state of the Glushkov automaton of E). Berry and Sethi's continuation automaton of E is

$$M_{\mathrm{cont}}(E) = (Q, \Sigma, \delta, q, F),$$

where $Q = \{(x, C_x(\overline{E})) \,|\, x \in \Sigma_{\overline{E}} \cup \{q_E\}\}$, $\delta((x, C_x(\overline{E})), a) = \{(y, C_y(\overline{E})) \mid \overline{y} = a \text{ and } y \in first(C_x(\overline{E}))\}$ for $x \in \Sigma_{\overline{E}} \cup \{q_E\}$ and $a \in \Sigma$, $q = (q_E, \overline{E})$, $F = \{(x, C_x(\overline{E})) \mid \varepsilon \in L(C_x(\overline{E}))\}$.

Define $M_1 \simeq M_2$ if two automata M_1 and M_2 are isomorphic. It is proved [2] that $M_{\mathrm{cont}}(E) \simeq M_{\mathrm{g}}(E)$.

By definition, $M_{\mathrm{pd}}(\overline{E})$ takes elements in $\partial_{wx}(\overline{E})$ and $\{\overline{E}\}$ as states for $x \in \Sigma_{\overline{E}}$, with transitions labeled by letters in $\Sigma_{\overline{E}}$. Then $\overline{M_{\mathrm{pd}}(\overline{E})}$ is the automaton obtained from $M_{\mathrm{pd}}(\overline{E})$ by unmarking labels of transitions. From the correspondence between $\partial_{wx}(\overline{E})$ and $C_x(\overline{E})$ it is easy to see that the difference between

$\overline{M_{\mathrm{pd}}(\overline{E})}$ and $M_{\mathrm{cont}}(E)$, hence between $\overline{M_{\mathrm{pd}}(\overline{E})}$ and $M_{\mathrm{g}}(E)$, is whenever $C_x(\overline{E})$ and $C_y(\overline{E})$ are identical, they correspond to the same state in $\overline{M_{\mathrm{pd}}(\overline{E})}$ and correspond to different states in $M_{\mathrm{cont}}(E)$. This leads to the following proposition.

Define the equivalence $=_{c'} \subseteq Q^2$ by $(x_1, C_{x_1}(\overline{E})) =_{c'} (x_2, C_{x_2}(\overline{E}))$ iff $C_{x_1}(\overline{E}) = C_{x_2}(\overline{E})$. The equivalence is right invariant w.r.t. $M_{\mathrm{cont}}(E)$. Define the equivalence $=_c \subseteq (\Sigma_{\overline{E}} \cup \{q_E\})^2$ by $x_1 =_c x_2$ iff $C_{x_1}(\overline{E}) = C_{x_2}(\overline{E})$. The equivalence is right invariant w.r.t. $M_{\mathrm{g}}(E)$.

Proposition 9. $\overline{M_{\mathrm{pd}}(\overline{E})} \simeq M_{\mathrm{cont}}(E)/_{=_{c'}} \simeq M_{\mathrm{g}}(E)/_{=_c}$.

Define $\equiv_c \subseteq (\Sigma_{\overline{E}} \cup \{q_E\})^2$ by $x_1 \equiv_c x_2$ iff $\overline{C_{x_1}(\overline{E})} = \overline{C_{x_2}(\overline{E})}$. The equivalence is right invariant w.r.t. $M_{\mathrm{g}}(E)$. It is easy to see that $=_c \subseteq \equiv_c$. That is, $M_{\mathrm{g}}(E)/_{\equiv_c}$ is a quotient of $M_{\mathrm{g}}(E)/_{=_c}$. Let us compute the quotient. Suppose $M_{\mathrm{g}}(E) = (Q, \Sigma, \delta, q, F)$, then $M_{\mathrm{g}}(E)/_{=_c} = (Q/_{=_c}, \Sigma, \delta_{=_c}, [q]_{=_c}, F/_{=_c})$, $M_{\mathrm{g}}(E)/_{\equiv_c} = (Q/_{\equiv_c}, \Sigma, \delta_{\equiv_c}, [q]_{\equiv_c}, F/_{\equiv_c})$. Define the equivalence $\equiv \subseteq (Q/_{=_c})^2$ by $[x_1]_{=_c} \equiv [x_2]_{=_c}$ iff $\overline{C_{x_1}(\overline{E})} = \overline{C_{x_2}(\overline{E})}$. The equivalence is right invariant w.r.t. $M_{\mathrm{g}}(E)/_{=_c}$. Then $M_{\mathrm{g}}(E)/_{\equiv_c} \simeq M_{\mathrm{g}}(E)/_{=_c}/_{\equiv}$. From Proposition 9, $M_{\mathrm{g}}(E)/_{=_c}$ and $\overline{M_{\mathrm{pd}}(\overline{E})}$ are isomorphic. The difference between $\overline{M_{\mathrm{pd}}(\overline{E})}$ and $M_{\mathrm{pd}}(E)$ is that, for $C_x(\overline{E})$ and $C_y(\overline{E})$, $C_x(\overline{E}) \neq C_y(\overline{E})$, whenever $\overline{C_{x_1}(\overline{E})} = \overline{C_{x_2}(\overline{E})}$, they correspond to different states in $\overline{M_{\mathrm{pd}}(\overline{E})}$ and correspond to the same state in $M_{\mathrm{pd}}(E)$. Therefore it is easy to further see that $M_{\mathrm{g}}(E)/_{=_c}/_{\equiv}$ and $M_{\mathrm{pd}}(E)$ are isomorphic. Therefore we have

Theorem 2. $M_{\mathrm{pd}}(E) \simeq M_{\mathrm{g}}(E)/_{\equiv_c}$.

Remark 2. The above proof is possible mainly due to (1) $C_x(\overline{E})$ is unique, which is enabled by selecting a representative, $rt_x(\overline{E})$, for it, and (2) $C_x(\overline{E})$ is still a derivative of $\overline{E}$.

Remark 3. After setting $C_x(\overline{E}) = rt_x(\overline{E})$, the remaining part of the proof for Theorem 2 is in the spirit of Ilie and Yu [9], but reformulated in a more rigorous form and corrects several flaws contained in the original proof.

5 Conclusion

The paper proposed a characterization of the structure of derivatives and proved several properties of derivatives for an expression with distinct symbols. Base on this, it gave a representative of derivatives and presented an improved proof of Ilie and Yu [9] of the fact that the partial derivative automaton is a quotient of the Glushkov automaton.

We believe that the characterization of derivatives given in the paper is a useful technique for relevant researches.

References

1. Antimirov, V.: Partial derivatives of regular expressions and finite automaton constructions. Theoretical Computer Science 155, 291–319 (1996)

2. Berry, G., Sethi, R.: From regular expressions to deterministic automata. Theoretical Computer Science 48, 117–126 (1986)
3. Bruggemann-Klein, A.: Regular expressions into finite automata. Theoretical Computer Science 120, 197–213 (1993)
4. Brzozowski, J.A.: Derivatives of regular expressions. J. ACM 11(4), 481–494 (1964)
5. Calude, C., Calude, E., Khoussainov, B.: Deterministic Automata: Simulation, Universality and Minimality. Developments in Language Theory, 519–537 (1997)
6. Champarnaud, J.-M., Ziadi, D.: Canonical derivatives, partial derivatives and finite automaton constructions. Theoretical Computer Science 289, 137–163 (2002)
7. Chang, C.-H., Page, R.: From regular expressions to DFAs using compressed NFA's. Theoretical Computer Science 178(1-2), 1–36 (1997)
8. Glushkov, V.M.: The abstract theory of automata. Russian Math. Surveys 16, 1–53 (1961)
9. Ilie, L., Yu, S.: Follow automata. Information and Computation 186(1), 146–162 (2003)
10. McNaughton, R., Yamada, H.: Regular expressions and state graphs for automata. IRE Trans. on Electronic Computers 9(1), 39–47 (1960)
11. Ponty, J.-L., Ziadi, D., Champarnaud, J.-M.: A New Quadratic Algorithm to Convert a Regular Expression into an Automaton. In: Raymond, D.R., Yu, S., Wood, D. (eds.) WIA 1996. LNCS, vol. 1260, pp. 109–119. Springer, Heidelberg (1997)
12. Yu, S.: Regular Languages. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. I, pp. 41–110. Springer, Berlin (1997)

Triangular and Hexagonal Tile Self-assembly Systems

Lila Kari, Shinnosuke Seki, and Zhi Xu

Department of Computer Science, University of Western Ontario,
London, Ontario, N6A 5B7 Canada

Abstract. We discuss theoretical aspects of the self-assembly of triangular tiles, in particular, right triangular tiles and equilateral triangular tiles, and the self-assembly of hexagonal tiles. We show that triangular tile assembly systems and square tile assembly systems cannot be simulated by each other in a non-trivial way. More precisely, there exists a deterministic square (hexagonal) tile assembly system S such that no deterministic triangular tile assembly system that is a division of S produces an equivalent supertile (of the same shape and same border glues). There also exists a deterministic triangular tile assembly system T such that no deterministic square (hexagonal) tile assembly system produces the same final supertile while preserving border glues.

1 Introduction

A basic model of DNA computation by self-assembly was proposed by Adleman [1] and Winfree [2], based on the theory of Wang tiles [3]. In this model, the basic components are *square tiles* with sides painted with "glues", that can stick together to form supertiles if the glues at abutting edges match, and attach with sufficient strength.

A regular tiling of the plane is a highly symmetric tiling made up of congruent regular polygons. Only three such regular tilings exist: those made up of equilateral triangles, squares, or hexagons. This paper departs from the existing model of self-assembly by investigating, instead of square tiles, the case of *triangular tiles* and *hexagonal tiles*. We namely discuss the self-assembly by equilateral-triangular, right-triangular, and hexagonal tile systems.

Our line of investigation follows that started by Winfree [4], who showed how the formation of large structures made out of the aggregation of rectangular DNA complexes can simulate Blocked Cellular Automata (BCA), which have the computational power of Turing machines. Winfree, Liu, Wenzler, and Seeman [2] designed and experimentally produced two-dimensional DNA crystals by self-assembly. A systematic study of self-assembly as a computational process was initiated by Adleman [1], who studied the time complexity of a particular case of linear self-assembly via "step counting" and raised the question of the construction of large squares via self-assembly. Rothemund and Winfree [5] studied the self-assembly of squares at fixed temperature (the threshold that the sum of the strengths of glues of a tile have to surpass, in order for it to "stick" to

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 357–375, 2012.

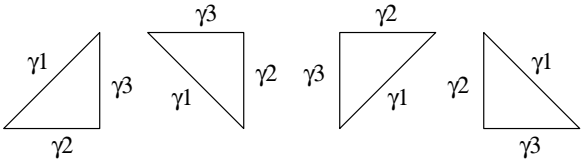


Fig. 1. Four kinds of isosceles right triangular tiles $(\gamma_1, \gamma_2, \gamma_3, \texttt{se})$, $(\gamma_1, \gamma_2, \gamma_3, \texttt{ne})$, $(\gamma_1, \gamma_2, \gamma_3, \texttt{nw})$, and $(\gamma_1, \gamma_2, \gamma_3, \texttt{sw})$

an existing assembled shape), and showed that in order to deterministically self-assemble an $N \times N$ full square (the square, N tiles on a side), N^2 different tile types are required at temperature $\tau = 1$ and $O(\log N)$ different tiles suffice at fixed temperature $\tau \geq 2$. Adleman, Cheng, Goel, and Huang [6] improved the latter result to $\Theta(\log N / \log \log N)$ different tiles. Kao and Schweller [7] showed that if the temperature τ is allowed to change systematically, then a constant number of tiles is enough for the self-assembly of an arbitrary $N \times N$ full square, with a temperature sequence of length $O(\log N)$.

In this paper, we follow a similar line of inquiry for triangular tiles and hexagonal tiles. Besides a natural theoretical interest, this study is motivated by the fact that triangular DNA tiles have been experimentally produced. For example, Liu, Wang, Deng, Walulu and Mao [8] reported the construction of a DNA triangle tile composed of three four-arm junctions, while Ding, Sha and Seeman [9] reported obtaining a triangular DNA tile formed from DX DNA molecules, and He, Chen, Liu, Ribbe and Mao [10] built a 3-point DNA star tile.

In this paper, in Sect. 2 we introduce the definition of triangular, respectively hexagonal, tile assembly systems. In Sect. 3 we compare the square tile assembly systems and triangular tile assembly systems from the point of view of shapes of the final supertiles they generate and show that the two types of systems cannot be simulated by each other in a straightforward way; we also compare the triangular tile assembly systems and hexagonal tile assembly systems.

2 Preliminaries

Our discussion of the triangular, respectively hexagonal, tile assembly systems will make use of the following definitions.

A *triangular tile* is a tile with three edges, each of which is "colored" with elements from a finite set Γ, called a *glue set*, whose elements dictate the interactions between the tiles. For all tiles discussed in this paper, we assume that the shortest side of the tile is of unit length, and that tiles cannot be rotated or flipped over.

An isosceles right triangular tile is a triangular tile in the shape of an isosceles right triangle, with each of its three edges colored by a glue from the glue set, and with the right angle pointing to the four possible directions: South-East, North-East, North-West, South-West as illustrated in Fig. 1. More formally, an

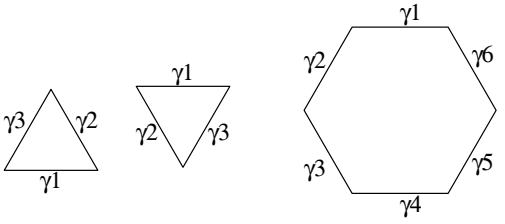


Fig. 2. Two equilateral triangular tiles $(\gamma_1, \gamma_2, \gamma_3, \mathtt{u})$, $(\gamma_1, \gamma_2, \gamma_3, \mathtt{d})$ and a hexagonal tile $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6)$

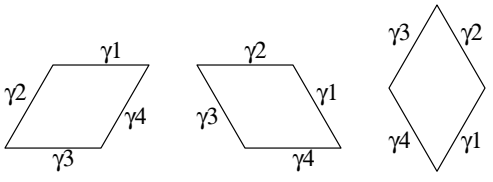


Fig. 3. Three diamond tiles $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \mathtt{II})$, $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \mathtt{IV})$, and $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \mathtt{VI})$

isosceles right triangular tile t is represented as a quadruple $(\gamma_1, \gamma_2, \gamma_3, k)$, where $\gamma_1, \gamma_2, \gamma_3 \in \Gamma$ are the glues on the sides of the tile in the counter-clockwise order starting from the longest side, and $k \in \{\mathtt{se}, \mathtt{ne}, \mathtt{nw}, \mathtt{sw}\}$ presents the direction pointed to by the right angle. In the rest of this paper, we denote the glues $\gamma_1, \gamma_2, \gamma_3$ of the 3 edges of a tile t by $\gamma_1(t)$, $\gamma_2(t)$, and $\gamma_3(t)$. Throughout this paper we will call isosceles right triangular tiles simply right triangular tiles.

An equilateral triangular tile is a triangular tile in the shape of an equilateral triangle, with its edges colored by glues from the glue set, and that is either in an upward position or in a downward position as illustrated in Fig. 2. An equilateral triangular tile is formally represented as a quadruple $(\gamma_1, \gamma_2, \gamma_3, k)$, where $\gamma_1, \gamma_2, \gamma_3 \in \Gamma$ are the glues on the sides of the tile in the counter-clockwise order starting from the horizontal side and $k \in \{\mathtt{u}, \mathtt{d}\}$ presents the upward, respectively downward, orientation of the "arrow" represented by the triangle. The notations $\gamma_1(t)$, $\gamma_2(t)$, and $\gamma_3(t)$ are defined in the same way as for right triangular tiles.

A *regular hexagonal tile* is a tile in the shape of a regular hexagon, with each of the six edges being colored with glues from the set Γ. Unlike triangular tiles, two geometrically adjacent regular hexagonal tiles must be of the same orientation. Without loss of generality, we assume that all regular hexagonal tiles are positioned as illustrated in Fig. 2. More formally, a regular hexagonal tile t is represented as a tuple $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6)$, where $\gamma_i \in \Gamma$ are the glues on the sides of the tile in the counter-clockwise order starting from the top-most side. The notations $\gamma_i(t)$ for $i = 1, \ldots, 6$ are defined in the same way as for triangular tiles. In this paper we will only investigate regular hexagonal tiles, and simply call them hexagonal tiles.

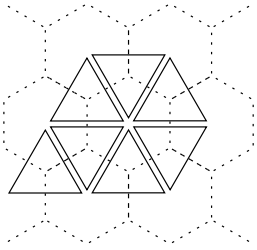


Fig. 4. A hexagonal grid graph

A *diamond tile* is a tile in the shape of a diamond (rhombus), one angle of which is $\pi/3$, with each of the four edges being colored with glues from the set Γ. We assume each diamond tile is in one of the three possible positions illustrated in Fig. 3. More formally, a diamond tile t is represented as a tuple $(\gamma_1, \gamma_2, \gamma_3, \gamma_4, k)$, where $\gamma_i \in \Gamma$ are the glues on the sides of the tile and $k \in \{\mathtt{II}, \mathtt{IV}, \mathtt{VI}\}$ as specified by the three typical examples in Fig. 3. The notations $\gamma_i(t)$ for $i = 1, \ldots, 4$ are defined in the same way as for triangular tiles.

Let us define $\mathcal{T}_{\mathrm{sq}} = \Gamma^4$, $\mathcal{T}_{\mathrm{R}} = \Gamma^3 \times \{\mathtt{se}, \mathtt{ne}, \mathtt{nw}, \mathtt{sw}\}$, $\mathcal{T}_{\triangle} = \Gamma^3 \times \{\mathtt{u}, \mathtt{d}\}$, $\mathcal{T}_{\mathrm{H}} = \Gamma^6$, and $\mathcal{T}_{\mathrm{D}} = \Gamma^4 \times \{II, IV, VI\}$ as the sets of all possible square tiles, right triangular tiles, equilateral triangular tiles, hexagonal tiles, and diamond tiles respectively, given the glue set Γ. We can further split $\mathcal{T}_{\mathrm{R}}$ into four disjoint subsets $\mathcal{T}_{\mathrm{R},\mathtt{se}}, \mathcal{T}_{\mathrm{R},\mathtt{ne}}, \mathcal{T}_{\mathrm{R},\mathtt{nw}}, \mathcal{T}_{\mathrm{R},\mathtt{sw}}$ depending on the fourth element of tiles, defined as: $\mathcal{T}_{\mathrm{R},x} = \{(\gamma_1, \gamma_2, \gamma_3, x) \in \mathcal{T}_{\mathrm{R}}\}$ for $x \in \{\mathtt{se}, \mathtt{ne}, \mathtt{nw}, \mathtt{sw}\}$. In a similar manner, $\mathcal{T}_{\triangle}$ can be split into the two disjoint subsets $\mathcal{T}_{\triangle,\mathtt{u}}, \mathcal{T}_{\triangle,\mathtt{d}}$ and $\mathcal{T}_{\mathrm{D}}$ can be split into three disjoint subsets $\mathcal{T}_{\mathrm{D},\mathtt{II}}, \mathcal{T}_{\mathrm{D},\mathtt{IV}}, \mathcal{T}_{\mathrm{D},\mathtt{VI}}$.

Let us proceed now to augment the notion of glue by associating to every glue a numerical "glue strength" as follows. Let $\mathcal{R}$ be the set of non-negative real numbers. Let $\Gamma = \{(\ell_1, n_1), (\ell_2, n_2), \ldots, (\ell_k, n_k) \mid n_1, \ldots, n_k \in \mathcal{R}$, where $\ell_1, \ldots, \ell_k$ are unique labels, i.e., $\ell_i = \ell_j$ iff $i = j\}$ for some $k \geq 1$. The set Γ dictates interactions between tiles, where for each $1 \leq i \leq k$, ℓ_i is the label of the i-th glue and n_i is the glue strength associated with it. (In the figures of this paper, the strength associated with the glue on a side will be represented by the number of parallel edges along that side.) A particular glue $\phi \in \Gamma$, defined as $\phi = (\phi, 0)$, denotes the non-interactive glue. Tiles can stick to each other by the glues on their adjacent edges to form *supertiles*.

Let T be a set of tiles of the same kind (square, equilateral triangle, right triangle, hexagon, or diamond). Conventionally, tiling the plane by tiles in T is modeled as a partial function from either the set of coordinates on the plane, or from the corresponding undirected lattice graph, to the set of tiles T. This partial function is called a *supertile* of T. Tiles assigned to adjacent vertices of the lattice graph are considered to be adjacent in the supertile.

For square supertiles, the coordinate system should be orthogonal, and hence, the corresponding graph is the grid graph (the two-dimensional integer lattice). In contrast, the lattice graph for a supertile made of triangular tiles should

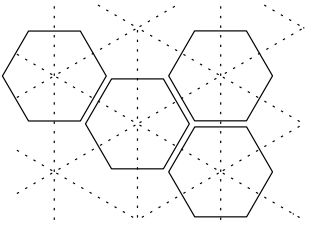


Fig. 5. A triangular grid graph

be *3-regular* (each vertex of the underlying graph has 3 neighbours) because a triangular tile can abut to at most 3 other tiles. Thus, the most appropriate lattice graph for tiling by equilateral triangular tiles is a *hexagonal grid graph* $H = (V, E)$ (see Fig. 4). In order to enforce the condition that two upward equilateral triangular tiles are never adjacent to each other, and neither are two downward ones, H has to be bipartite as: $V = V_{\mathtt{u}} \cup V_{\mathtt{d}}$ and $E \subseteq V_{\mathtt{u}} \times V_{\mathtt{d}}$. For $T \subseteq \mathcal{T}_{\triangle}$, a supertile C of T is defined as a partial function from V to T such that

1. for any $t_{\mathtt{u}} \in \mathcal{T}_{\triangle,\mathtt{u}}$, if $C(v) = t_{\mathtt{u}}$, then $v \in V_{\mathtt{u}}$, and
2. for any $t_{\mathtt{d}} \in \mathcal{T}_{\triangle,\mathtt{d}}$, if $C(v') = t_{\mathtt{d}}$, then $v' \in V_{\mathtt{d}}$.

For defining a supertile of a set of right triangular tiles, the underlying 3-regular lattice graph $G = (V', E')$ should be a 4-partite graph because there are the four kinds of right triangular tiles. Hence, let $V' = V_{\mathtt{se}} \cup V_{\mathtt{ne}} \cup V_{\mathtt{nw}} \cup V_{\mathtt{sw}}$ and E' satisfy

1. for $v_{\mathtt{se}} \in V_{\mathtt{se}}$, $\{(v_{\mathtt{se}}, v_1), (v_{\mathtt{se}}, v_2), (v_{\mathtt{se}}, v_3)\} \subseteq E'$ such that $v_1 \in V_{\mathtt{nw}}$, $v_2 \in V_{\mathtt{ne}} \cup V_{\mathtt{nw}}$, and $v_3 \in V_{\mathtt{nw}} \cup V_{\mathtt{sw}}$;
2. for $v_{\mathtt{ne}} \in V_{\mathtt{ne}}$, $\{(v_{\mathtt{ne}}, v'_1), (v_{\mathtt{ne}}, v'_2), (v_{\mathtt{ne}}, v'_3)\} \subseteq E'$ such that $v'_1 \in V_{\mathtt{sw}}$, $v'_2 \in V_{\mathtt{nw}} \cup V_{\mathtt{sw}}$, and $v'_3 \in V_{\mathtt{se}} \cup V_{\mathtt{sw}}$;
3. for $v_{\mathtt{nw}} \in V_{\mathtt{nw}}$, $\{(v_{\mathtt{nw}}, v''_1), (v_{\mathtt{nw}}, v''_2), (v_{\mathtt{nw}}, v''_3)\} \subseteq E'$ such that $v''_1 \in V_{\mathtt{se}}$, $v''_2 \in V_{\mathtt{se}} \cup V_{\mathtt{sw}}$, and $v''_3 \in V_{\mathtt{se}} \cup V_{\mathtt{ne}}$;
4. for $v_{\mathtt{sw}} \in V_{\mathtt{sw}}$, $\{(v_{\mathtt{sw}}, v'''_1), (v_{\mathtt{sw}}, v'''_2), (v_{\mathtt{sw}}, v'''_3)\} \subseteq E'$ such that $v'''_1 \in V_{\mathtt{ne}}$, $v'''_2 \in V_{\mathtt{se}} \cup V_{\mathtt{ne}}$, and $v'''_3 \in V_{\mathtt{ne}} \cup V_{\mathtt{nw}}$.

For $T' \subseteq \mathcal{T}_{\mathrm{R}}$, a supertile C' of T' is defined as a partial function from V' of G to T' such that for any $t_x \in \mathcal{T}_{\mathrm{R},x}$, if $C'(v) = t_x$, then $v \in V_x$, where $x \in \{\mathtt{se}, \mathtt{ne}, \mathtt{nw}, \mathtt{sw}\}$. For defining the supertile of a set of hexagonal tiles, a *6-regular* lattice graph, *triangular grid graph* $Tr = (V'', E'')$, is adopted (see Fig. 5). For $T'' \subseteq \mathcal{T}_{\mathrm{H}}$, a supertile C'' of T'' is defined as a partial function from V'' of Tr to T''. For defining the supertile of a set of diamond tiles, a *4-regular* lattice graph $G' = (V''', E''')$ is needed and the lattice graph is a 3-partite graph. Let $V''' = V_{\mathtt{II}} \cup V_{\mathtt{IV}} \cup V_{\mathtt{VI}}$ and E''' satisfy:

1. for $v_{\mathtt{II}} \in V_{\mathtt{II}}$, $\{(v_{\mathtt{II}}, v_1), (v_{\mathtt{II}}, v_2), (v_{\mathtt{II}}, v_3), (v_{\mathtt{II}}, v_4)\} \subseteq E'''$ such that $v_1, v_3 \in V_{\mathtt{II}} \cup V_{\mathtt{IV}}$ and $v_2, v_4 \in V_{\mathtt{II}} \cup V_{\mathtt{VI}}$;
2. for $v_{\mathtt{IV}} \in V_{\mathtt{IV}}$, $\{(v_{\mathtt{IV}}, v_1), (v_{\mathtt{IV}}, v_2), (v_{\mathtt{IV}}, v_3), (v_{\mathtt{IV}}, v_4)\} \subseteq E'''$ such that $v_2, v_4 \in V_{\mathtt{II}} \cup V_{\mathtt{IV}}$ and $v_1, v_3 \in V_{\mathtt{IV}} \cup V_{\mathtt{VI}}$;
3. for $v_{\mathtt{VI}} \in V_{\mathtt{VI}}$, $\{(v_{\mathtt{VI}}, v_1), (v_{\mathtt{VI}}, v_2), (v_{\mathtt{VI}}, v_3), (v_{\mathtt{VI}}, v_4)\} \subseteq E'''$ such that $v_1, v_3 \in V_{\mathtt{II}} \cup V_{\mathtt{II}} \cup V_{\mathtt{VI}}$ and $v_2, v_4 \in V_{\mathtt{IV}} \cup V_{\mathtt{VI}}$.

For $T''' \subseteq \mathcal{T}_{\mathrm{D}}$, a supertile C''' of right triangular tiles from T''' is defined as a partial function from V''' of G' to T'''. In the definition of a supertile, both the hexagonal grid graph H and the triangular grid graph Tr are unique, but there are more than one valid lattice graphs to present supertiles for right triangular tiles and for diamond tiles.

Let us now formally define the interaction between tiles which depends on the match between the glues on the tiles' adjacent edges and also on a threshold parameter called *temperature* $\tau \in \mathcal{R}$ that determines whether or not the "sticking" is strong enough for the new tile to attach to an existing supertile. In general, the *strength function* $g : \Gamma \times \Gamma \to \mathcal{R}$ is defined such that $g(\gamma, \gamma') = g(\gamma', \gamma)$ and $g(\phi, \gamma) = 0$ for all $\gamma, \gamma' \in \Gamma$. In particular, we are interested in the discrete case where τ is an integer and $g((\ell, n), (\ell', n')) = n$ if $\ell = \ell'$ and $n = n'$; $g((\ell, n), (\ell', n')) = 0$ otherwise. We call a supertile D *full* if the strength $g(\gamma_i(D(v)), \gamma_i(D(v_i)))$ of common edges of every two adjacent tiles $D(v)$ and $D(v_i)$ in the supertile is strictly positive.

In order to model the growth of tile assemblies, we need to define the notion of *attachability*. Let $T \subseteq \mathcal{T}_{\triangle}$ be a set of equilateral triangular tiles and C, D be two supertiles of T. We say that t *attaches* to C at vertex v, to derive D, and we write $C \to_{T,g,\tau} D$, if the following conditions hold. Firstly, $C(u) = D(u)$ for all $u \in \mathrm{dom}(C)$. Secondly, there exist some $t \in T$ and $v \in V$ such that $C(v)$ is undefined, $\mathrm{dom}(C) = \mathrm{dom}(D) \setminus \{v\}$, $D(v) = t$, and for every $\{(v, v_1), (v, v_2), (v, v_3)\} \subseteq E$ we have,

$$\sum_{i \in \{1,2,3\}} g(\gamma_i(t), \gamma_i(D(v_i))) \geq \tau.$$

Informally, the supertile D is derived from the supertile C by the attachment of t to C iff the sum of the glue strengths on those edges of t that are adjacent to C is greater than or equal to the threshold τ. Note that in the definition of attachability, we do not require either C or D be full.

We can define the notions of attachability and transition for right triangular tiles, hexagonal tiles and diamond tiles in a similar manner, and those notions for square tiles can be found in the literature [6]. The reflexive and transitive closure of $\to_{T,g,\tau}$ is denoted by $\to^*_{T,g,\tau}$.

A *tile assembly system* (TAS) is a tuple $S = (T, s, g, \tau)$. T is a finite set of tiles of the *same kind*; so $T \subseteq \mathcal{T}_{\mathrm{sq}}$ ($T \subseteq \mathcal{T}_{\mathrm{R}}$, $T \subseteq \mathcal{T}_{\triangle}$, $T \subseteq \mathcal{T}_{\mathrm{H}}$, $T \subseteq \mathcal{T}_{\mathrm{D}}$) implies that all tiles of S are square (respectively, right triangular, equilateral triangular, hexagonal, diamond). The other parameters of S mean that $s \in T$ is a special supertile called the *seed*, g is a strength function, and τ is the temperature.

We now define the notion of *derived supertile* of a given TAS S as follows. The seed tile s, when placed in an a priori chosen "reference position" on the

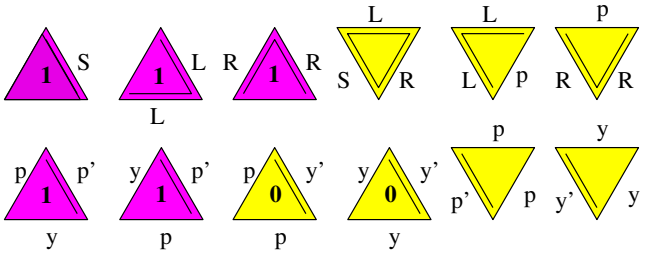


Fig. 6. A set of 12 tiles from which the Sierpinski triangle self-assembles deterministically. S, L, R, p, p', y, and y' are the glue labels and the number of parallel lines along each edge denotes the glue strength. The labels `1` and `0` specify the digits used in the XOR operation.

plane or on the grid graph, is a partial function called the *seed derived supertile* or simply *seed supertile*. For example, in the case of square TASs, we may choose to always place the seed supertile on the plane as the square with corners at coordinates $(0,0),(0,1),(1,0),(1,1)$. A *derived supertile* of S is a supertile C such that $s \rightarrow^*_{T,g,\tau} C$. A *final supertile* of S is a derived supertile C such that $C \rightarrow^*_{T,g,\tau} D$ implies $C = D$ for any supertile D, that is, no tile is attachable at any vertex in C. The number of tile types of S is called the *program size complexity* of S, and is denoted by $|S|$ [5].

A TAS is said to be *deterministic* if its final supertile is unique regardless of how the self-assembly proceeds starting from the seed. Otherwise, the TAS is said to be *non-deterministic*. A non-deterministic TAS can have many different final supertiles possibly with different shapes. *In this paper, unless explicitely stated otherwise, all tile systems are assumed to be deterministic.* When $T \subseteq \mathcal{T}_{\mathrm{R}}$ (respectively, $T \subseteq \mathcal{T}_{\triangle}$, $T \subseteq \mathcal{T}_{\mathrm{H}}$, $T \subseteq \mathcal{T}_{\mathrm{D}}$), S is explicitly called a *right triangular TAS* (respectively, an *equilateral triangular TAS*, a *hexagonal TAS*, a *diamond TAS*).

Before we discuss shapes generated by non-square self-assembly systems, let us first see an example of how a Sierpinski triangle can self-assemble using tiles from an equilateral triangular TAS.

Proposition 1. *There exists an equilateral triangular tile system which deterministically self-assembles the Sierpinski triangle at temperature* $\tau = 2$.

Proof. Let us recall the square tile system which deterministically self-assembles the Sierpinski triangle [11], whose tile set contains seven tile types in two colour categories, dark and light. Using the technique of "division", formally defined in Section 3, we transform each of the square tiles by flattening it into a parallelogram, and dividing this parallelogram to obtain one upward and one downward equilateral triangular tile, see Fig. 9. Furthermore, by reusing some of the triangular tiles, the number of triangular tiles needed to assemble an arbitrarily large Sierpinski triangle is reduced to twelve. The tile set of an equilateral triangular tile system thus designed is illustrated in Fig. 6.

The seed tile is $(\phi, S, \phi, \mathtt{u})$ and together with the other five tiles on the top row in Fig. 6 can form the L-shaped supertile, which composes the outmost left and bottom boundary of the Sierpinski triangle (See Fig. 7). Then each parallelogram-shaped space is filled by a pair of triangular tiles that together simulate a flattened square tile that implements an XOR-like rule. This XOR rule takes as input the left and bottom glues of the first triangular tile of the pair (corresponding to the left and bottom neighbours of that tile), and outputs the result as the right and top glues of the second triangular tile of the pair. The tiles of the pair are held together by a glue of strength 2. Four tile types are enough to implement the XOR operation. Two more tile types deliver the information and fill the triangle, as illustrated on the bottom row in Fig. 6. The result is a Sierpinski triangle as illustrated in Fig. 7.

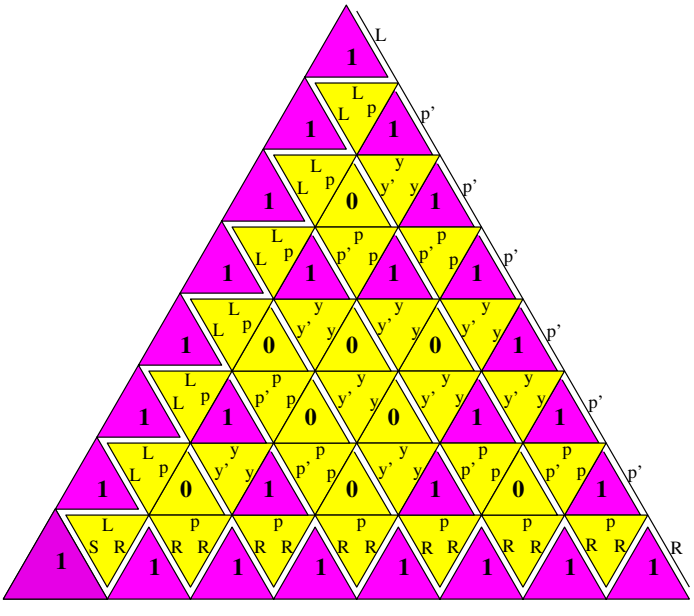


Fig. 7. The Sierpinski-triangle which consists of 64 equilateral triangular tiles with 12 different tile types

3 Comparing Supertiles

We now proceed to compare the final supertiles produced by various tile assembly systems in terms of their shape and boundary glues.

The tiles we considered are well defined geometrically by their shape and the fact that their shortest edge is of unit length. Thus, given a TAS with tiles of shape α, where $\alpha \in \{\text{square, triangle, right triangle, hexagon}\}$, we can now associate to every supertile a corresponding region in $\mathcal{R}^2$ as follows.

We associate to a supertile of size 1, i.e., consisting of one tile only, the region in $\mathcal{R}^2$ enclosed by the edges of that tile, assuming the tile is placed on the two-dimensional plane at an *a priori* chosen reference position. For example if $\alpha =$ square, then the corresponding region is the square (including its interior) with corners $(0,0),(0,1),(1,0),(1,1)$. Let us assume we have associated to the seed tile a region in $\mathcal{R}^2$ in this fashion. We can now associate to a supertile of size 2, obtained by attaching a single tile to the seed tile, a region in $\mathcal{R}^2$. This is obtained by taking the union between the region in $\mathcal{R}^2$ coresponding to the seed tile, and the region in $\mathcal{R}^2$ resulting by translating the region in $\mathcal{R}^2$ corresponding to the second tile to the position where it attaches to the seed. By iterating the process, we can thus associate to each supertile that is derived from the seed a corresponding region in $\mathcal{R}^2$.

Two supertiles are said to have *the same shape* if their corresponding regions of $\mathcal{R}^2$ are identical. If, in addition, the glues on the boundaries of the two supertiles (but not necessarily the internal glues) are the same, the two supertiles are said to be *equivalent*. The fact that, in order to be considered equivalent, final supertiles have the same boundary glues in addition to covering the same region in $\mathcal{R}^2$, reflects the fact that supertiles are often used as components for further assemblies, and thus have to have the same "sticking properties" if they are to be used interchangeably.

A region $Y \subseteq \mathcal{R}^2$ is called α-compatible, where α is an element of {square, right triangle, equilateral triangle, hexagon, diamond}, if Y can be geometrically "covered" by tiles from an α-TAS, i.e., if Y can be written as the set union of regions, overlapping at most on their edges, that are obtained by translating the regions corresponding to single tiles from an α-TAS. We call a supertile α-compatible if its corresponding region in $\mathcal{R}^2$ is α-compatible.

For example, the region corresponding to the final supertile of any hexagonal TAS is equilateral-triangle-compatible, and none of the triangular regions of $\mathcal{R}^2$ is square-compatible. For a given α-TAS, only the assembly of final supertiles of α-compatible shapes is meaningful. Hence, in the remaining discussion, we only consider the assembly of final supertiles of α-compatible shapes.

First, note that for any α-compatible supertile, there is a trivial α-TAS (deterministic or non-deterministic) that produces a final supertile of the same shape.

Proposition 2. *Let $\alpha \in$ {square, right triangle, equilateral triangle, hexagon, diamond} and let Y be an α-compatible supertile. There exists a non-deterministic α-TAS of a constant number of tile types whose final supertile has the same shape as Y. If Y is finite, then there exists a deterministic α-TAS with n tile types whose final supertile has the same shape as Y, where n is the total number of tiles needed to mosaic Y geometrically.*

Proof. Let us consider α being equilateral triangle. The other cases are similar.

Consider first the non-deterministic case. Let T be the set of tiles $T = \{(a,b,c,k) \mid a,b,c \in \{\phi, g\}, k \in \{\mathtt{u}, \mathtt{d}\}\}$. All glues of g are of strength 1 and temperature is $\tau = 1$. The seed and assembly process are as follows: The supertile Y is assembled according to the geometrical division of the region enclosed

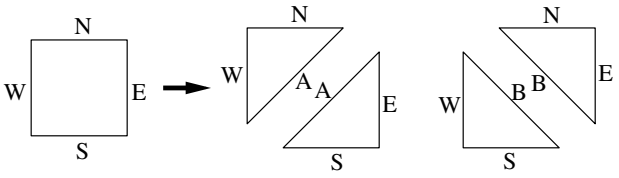


Fig. 8. Two ways to divide a square tile into right triangular tiles

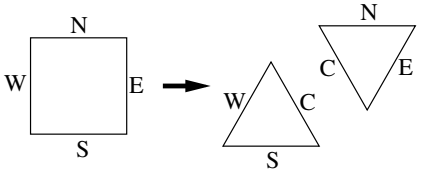


Fig. 9. Even with the help of affine transformations, squares can be divided into two equilateral triangles in only one way

by Y into triangular tiles of T. This can be done since Y is equilateral-triangle-compatible. At each step, a tile sticks to the supertile in such a way that if the tile is surrounded by other tiles in the completed region, then every edge of that tile is of glue g; otherwise, the edges that compose the boundary of that region are of empty glue ϕ. Then a final supertile of the same shape as Y can be produced by the given TAS with at most 16 tile types.

For the deterministic case, we mosaic T geometrically with equilateral triangular tiles. If n tiles are needed, we define a TAS consisting of n tiles, where the glues between each two tiles that stick to each other in the final supertile are unique.

Proposition 2 shows that, if we are interested only in shape, and we either *a)* do not care that a unique supertile is assembled, or *b)* we care about uniqueness of the final supertile but we do not care about the program complexity of the tile system (how many tile types are needed), then α-compatible tile systems can essentially produce final supertiles of the same shape. This is because every α-compatible supertile can be produced by a trivial but huge deterministic α-TAS of type α, where $\alpha \in \{$square, right triangle, equilateral triangle, hexagon, diamond$\}$, or by a non-deterministic one. In what follows, we will discuss natural restrictions on the comparison of TASs, that avoid these trivial cases.

We first compare the triangular TASs and square TASs from the point of view of the shapes of the final supertiles they generate. A right triangular TAS $S_{\mathrm{R}} = (T_{\mathrm{R}}, s_{\mathrm{R}}, g, \tau)$ is called a *triangular division* of a square TAS $S = (T, s, g, \tau)$, if:

1. For any square tile $t \in T$ of S, there is a pair of triangular tiles $t', t'' \in T_{\mathrm{R}}$ of S_{R} whose hypotenuses are colored with the same glue (l_t, n_t) with $n_t \geq \tau$

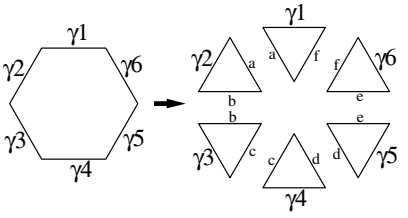


Fig. 10. A hexagonal tile is divided into six equilateral triangular tiles

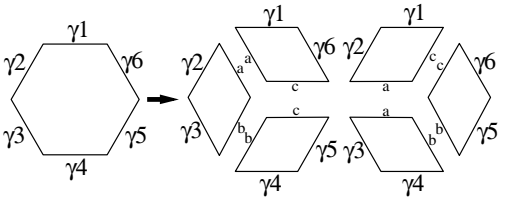


Fig. 11. A hexagonal tile is divided into three diamond tiles in two different ways

so that at temperature $\tau \geq 1$, these tiles can stick to each other via their hypotenuses and result in a two-tile supertile equivalent to the square tile t;

2. For any triangular tile $t \in T_{\mathrm{R}}$, there exists another triangular tile $t' \in T_{\mathrm{R}}$ of S_{R} such that these two tiles can stick to each other via their hypotenuses and produce a supertile that is equivalent to a square tile in T at temperature $\tau \geq 1$.

Note that the "hypothenuse glues" may or may not be distinct for different triangular tile pairs. Note also that the numbers of tiles in the two systems, the square tile system S and its division S_R, above satisfy the inequality $\sqrt{|T|} \leq |T_R|$. By definition, the division of a square TAS may not be unique. This is mainly because a square tile can be divided into two right triangular tiles in two different ways (see Fig. 8). In addition, two different square TASs can have the same right triangular TAS as a division. Finally, note that even if a square TAS is deterministic, its triangular division may not be so. A triangular division of a square TAS S is called a *deterministic triangular division* if it is a triangular division of S and, in addition, it is deterministic.

Let us define the *flattening function* $f : \mathcal{T}_{\triangle} \rightarrow \mathcal{T}_{\mathrm{R}}$ as $f((\gamma_1, \gamma_2, \gamma_3, \mathtt{u})) = (\gamma_2, \gamma_3, \gamma_1, \mathtt{sw})$ and $f((\gamma_1, \gamma_2, \gamma_3, \mathtt{d})) = (\gamma_2, \gamma_3, \gamma_1, \mathtt{ne})$. This function has the effect of "flattening" an equilateral triangular TAS $S = (T, s, g, \tau)$ into a right triangular TAS $\mathcal{F}(S) = (U, f(s), g, \tau)$, where $U = \{f(t) \mid t \in T\}$. Informally, a flattened right-triangular TAS is obtained from an equilateral triangular one by morphing each of the equilateral triangular tiles into either a South-West pointing, or respectively North-East pointing right triangular one.

An equilateral triangular TAS T is called a *division of a square TAS* S, Fig. 9, if the flattened TAS $\mathcal{F}(T)$ obtained from it is a division of S. T is called a

deterministic division of S if it is a division of S which is deterministic. The numbers of tiles in the two systems, the square tile system S with a tile set T_S and its equilateral triangle division T with the tile set T_T, satisfy the inequality $\sqrt{|T_S|} \leq |T_T|$.

We now ask the question of whether or not any square TAS can be converted, by division, into a triangular TAS that produces an equivalent final supertile. In general, the answer is "no", as shown by the following lemma.

Lemma 1. *There exists a deterministic square TAS, none of whose deterministic triangular divisions produces an equivalent final supertile.*

Two examples proving this lemma are illustrated in Fig. 13, one for $\tau = 2$, (left), and one for $\tau = 3$, (center). In the figure, each tile is numbered in the order of a possible assembly process.

For the example in Fig. 13 (left), each of the square tiles $s, 1, \ldots, 6$ can be simulated by a pair of right triangular tiles. There are two sticky edges for tile 7, which are on parallel sides of the square tile, each of which is of strength 1. So under $\tau = 2$ the attachment of tile 7 cannot be simulated by successive attachments of two right triangular tiles to assemble the same final supertile. This is because the edges necessary for tile 7 to attach are its North and South edges, both of them of strength one. No matter how we divide this square tile into two triangles, the North and South edges will belong to different triangular tiles and, because they have only strength $1 \leq \tau = 2$, neither of them can attach to the existing supertile.

The next example is the 4×4 square in Fig. 13 (center). By a similar reasoning, the attachment of tile 11 cannot be simulated by successive attachments of two right triangular tiles, and thus, the assembly stops and fails to grow into the 4×4 square.

The supertile in Fig. 13 (left) has a missing tile in the middle, and we say that it has a *"hole"*. In general, a derived supertile S is hole-free if it is full and for any closed tile-path, all the positions of the grid subgraph corresponding to S that are inside the closed path are filled with tiles.

Lemma 2. *For any deterministic square TAS S at $\tau = 1$, and any square TAS at $\tau = 2$ whose final supertile is hole-free, there is a triangular division of S that can produce an equivalent final supertile.*

Proof. For $\tau = 1$, the proof is straightforward. Any square tile s_i with glues $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ (on East, North, West, South sides, respectively) in S is replaced with a pair of right triangular tiles $(i, \gamma_1, \gamma_2, \texttt{ne})$ and $(i, \gamma_3, \gamma_4, \texttt{sw})$, where i is a new glue added, and when s_i is the seed, we let $(i, \gamma_1, \gamma_2, \texttt{ne})$ be the new seed. Then the new right triangular TAS is a division of S and produces an equivalent final supertile.

Now we assume $\tau = 2$ and assume that there is no hole in the final supertile of S.

First we prove that for any hole-free derived supertile st, there is an assembly process such that every derived supertile in the process is hole-free. For an

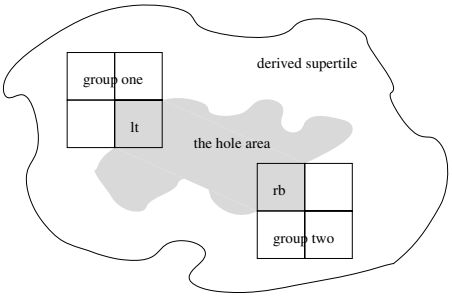


Fig. 12. Proof of Lemma 2: Tiles lt and rb in a hole area

assembly process $p : st_0 \to st_1 \to st_2 \to \ldots \to st_n = st$, let $f(p)$ be the number such that st_i is hole-free for $i < f(p)$ but $st_{f(p)}$ has a hole. For the case that none of si_i has a hole, we write $f(p) = +\infty$. Now choose a p such that $f(p)$ is the largest among all assembly process of st. We prove by contradiction and assume $f(p) \neq +\infty$. Let t be the new attached tile in the step $st_{f(p)-1} \to st_{f(p)}$. Since st_n is full, every st_i is also full. So $st_{f(p)}$ has a hole for the reason that there are missing tiles in a hole region. Those missing tiles will eventually be filled up in st_n since st_n is hole-free. Now we consider the following two tiles, not necessarily distinct, among all missing tiles in the hole of $st_{f(p)}$: the left-most tiles lt among the top-most tiles, and the right-most tiles rb among the bottom-most tiles. Then the closest positions to the North of lt, to the West of lt and to the North-West of lt have tiles on them, called group one, and similarly the closest positions to the South of rb, to the East of rb and to the South-East of rb have tiles on them, called group two. (See Fig. 12) Tile t cannot be in both group one and group two. Without loss of generality, we assume t is not in group one. So both West and North of lt are tiles in $st_{f(p)-1}$, and thus there are two adjacent edges in the hole area that can stick to lt due to the fullness. Then there is a valid assembly process p' such that the first few steps up to st_{i-1} are the same and then tile lt instead of t sticks to the supertile st_{i-1}. The process p' will finally assemble st since the TAS S is deterministic. In this case, we have $f(p') \geq f(p) + 1$, which contradicts the property that $f(p)$ is the largest among all assembly of st. Therefore $f(p) = +\infty$.

Now we prove, for the TAS S, that for the assembly process wherein all the intermediate supertiles are hole-free, a new tile can stick to the supertile at each step either by two adjacent edges or by an edge of strength at least 2. If the new tile sticks by more than two edges, then we can pick two adjacent edges. The only remaining case is when the new tile sticks by exactly two parallel edges. We show it is impossible. Without loss of generality, suppose st_{i-1} becomes st_i by sticking t to st_{i-1} by North and South sides. There is no tile on the East and on the West sides, or t can stick by two adjacent edges due to the fullness of st_i. But in this case, since st_{i-1} is a derived supertile, there is a tile-path between the two tiles to the North and to the South of t in st_{i-1}. So there is a closed

tile-path in st_i which encloses either the position to the East or the position to the West of t. In other words, st_i contains a hole, which contradicts the fact that st_i is hole-free.

We construct the following right triangular TAS: any square tile s_i with glues $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ (on East, North, West, South sides, respectively) in S is replaced with four right triangular tiles $(i, \gamma_2, \gamma_3, \texttt{nw})$, $(i, \gamma_4, \gamma_1, \texttt{se})$, $(i, \gamma_1, \gamma_2, \texttt{ne})$ and $(i, \gamma_3, \gamma_4, \texttt{sw})$, where i is a new glue added with strength $\geq \tau = 2$, and when s_i is the seed, we let $(i, \gamma_1, \gamma_2, \texttt{ne})$ be the new seed. Then the new right triangular TAS is a division of S. Since new tiles can stick to the supertile at each step by either two adjacent edges or by an edge of strength at least 2, the assembly of the square TAS can be simulated by the constructed right triangular TAS. So the constructed right triangular TAS produces a final supertile equivalent to the final supertile of the square TAS.

By Lemmas 1 and 2, we see that square TASs can be simulated by their right triangular divisions only under certain conditions. Now we discuss the other direction: whether every right triangular TAS can be simulated by a square TAS, assuming that the final supertile is square-compatible. For $\tau = 1$, the answer is a qualified "yes", as shown by the following lemma.

Lemma 3. *For any deterministic right triangular TAS S_R, at $\tau = 1$, if its final supertile is square-compatible and the strength of the glue of the hypothenuse of any of its tiles is 1, then there exists a deterministic square TAS S such that S_R is a division of S.*

Proof. Let t be a tile of S_R. Without loss of generality, assume that this tile is used at least once in the (unique) final supertile of S_R. Since the final supertile is square-compatible, there must be a (unique) tile that abuts t in the final supertile, by its hypothenuse. In addition, these two tiles attach to each other via a glue of strength 1. For each tile t in S_R, we add to S the square tile obtained by thus binding these two triangular tiles. Being thus contructed, S is deterministic, and S_R is its division.

For $\tau = 2$, the situation is different, as shown by the following lemma. Note that, in the following lemmas, when comparing two TASs, we ask the final supertiles generated by them to be equivalent, that is, to have the same shape and the same border glues.

Lemma 4. *There exists a deterministic right triangular TAS at $\tau = 2$, whose final supertile is square-compatible, but no deterministic square TAS produces an equivalent final supertile.*

An example of a right triangular TAS at $\tau = 2$ as postulated in Lemma 4 is illustrated in Fig. 13 (right), where each tile is numbered in the order of a possible assembly process. Note that any square TAS that produces a supertile of the same shape as the rightmost supertile depicted in Fig. 13 must include a square tile with West side glue `a`, and South glue `b`. However, if a tile system

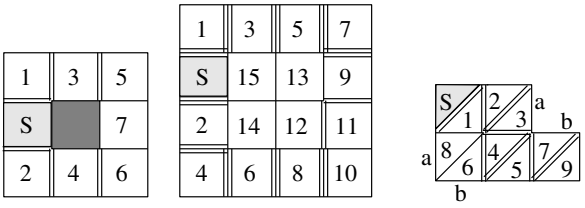


Fig. 13. Examples that show that square TASs and triangular TASs are, in some sense, not comparable from the point of view of the shapes of final supertiles they generate. The left and center figures depict two final supertiles of square systems at $\tau = 2$, and $\tau = 3$, respectively, that illustrate Lemma 1. The right figure ($\tau = 2$), illustrates Lemma 4. Each glue, unless mentioned, is unique, and thus, the label is omitted.

contained such a tile, its assembly would grow at its North-East corner and produce the 3×2 rectangle instead.

Lemmas 1 and 4 indicate that square TASs and triangular TASs are, in some sense, not comparable from the point of view of the shapes of final supertiles they generate.

The following lemma compares square TASs with hexagonal TASs from the point of view of the shapes of the final supertiles they generate.

Lemma 5. *No supertile is both hexagon-compatible and square-compatible, even under possible affine transformation on* $\mathcal{R}^2$.

Proof. Suppose there is a supertile st assembled by hexagonal tiles that is of the same shape as a supertile st' assembled by square tiles under affine transformation F. Let t be the left-most tile on the top-most row of tiles in st. Assume the vertices of t are $v_1, \ldots, v_6$, starting from the top-right vertex in the counterclockwise order. Then v_1, v_2, v_3, v_6 are also vertices of the supertile st by the position of t. Assume v'_1, v'_2, v'_3, v'_6 be the corresponding vertices in st'. Since an affine transformation transforms lines to lines, the angles v'_1, v'_2, v'_3 and v'_2, v'_1, v'_6 are of degree $\pi/4$ or $3\pi/4$. In other words, the two lines $v'_2v'_3$ and $v'_1v'_6$ are parallel. Then v_2v_3 and v_1v_6 should also be parallel, since an affine transformation preserves parallel relationship of lines. But v_2v_3 and v_1v_6 are not parallel, a contradiction. So no hexagon-compatible shape is square-compatible even under possible affine transformations on $\mathcal{R}^2$.

We now compare triangular TASs with hexagonal TASs from the point of view of the shapes of the final supertiles they generate.

An equilateral triangular TAS T is called a *triangular division* of a hexagonal TAS H, if *(i)* for any hexagonal tile h in H, there are six triangular tiles in T that can attach to each other to produce a supertile equivalent to h at temperature $\tau \geq 1$, and *(ii)* for any triangular tile t in T, there are five other triangular tiles in T, that can attach to each other to produce a supertile equivalent to a hexagonal tile in H (see Fig. 10). An equilateral triangular TAS T is called

a *deterministic triangular division* of a hexagonal TAS H if it is a triangular division of H that is a deterministic TAS. A right triangular TAS T is called a division of a hexagonal TAS H, if T is the flattening $\mathcal{F}(T')$ of an equilateral triangle TAS T' that is a division of H. A right triangular TAS T is called a *deterministic division* of a hexagonal TAS H if it is a division of H that is a deterministic TAS.

It is obvious that for any hexagonal TAS H at $\tau = 1$, there is a triangular division of H that produces a final supertile of the same shape. For $\tau = 2$ the situation is different, as shown by the following lemma.

Lemma 6. *There exists a deterministic hexagonal TAS H at $\tau = 2$, such that no deterministic triangular division of H produces an equivalent final supertile.*

An example is illustrated in Fig. 14 (left), for $\tau = 2$, where S is the seed and tile 1 is attached to the supertile before tile 2 is. Tiles s and 1 can be simulated by their triangular divisions, but tile 2 cannot, since the cooperation between edges, even for the case when the cooperative edges abut in the hexagonal tile, cannot be preserved when replacing a hexagonal tile by equilateral triangular tiles. Indeed, no triangular tile from the division of the hexagonal tile can attach to the supertile formed by the seed and tile 1, since the edges that would be needed for any of them to attach all have strength 1. This simple example shows that most hexagonal TASs cannot be simulated by their triangular divisions.

It is obvious that for any equilateral triangular TAS T at $\tau = 1$ whose final supertile is hexagon-compatible, there is a hexagonal TAS H that produces an equivalent final supertile. For $\tau = 2$, the situation is different, as shown by the following lemma.

Lemma 7. *There exists a deterministic equilateral triangular TAS T at $\tau = 2$ whose final supertile is hexagon-compatible, but no deterministic hexagonal TAS produces an equivalent final supertile.*

An example is illustrated in Fig. 14 (right), where tiles are numbered in a possible order of assembly. Note that any hexagonal TAS that produces an equivalent final supertile as that of Fig. 14 (right), has to contain a hexagonal tile t such that $\gamma_1(t) = b$ and $\gamma_2(t) = a$. Then the process of assembly of such a hexagonal TAS can grow further in the right-bottom direction, and thus cannot produce the unique required final supertile.

By replacing pairs of tiles $(s, 1), (2, 3), \ldots, (10, 11)$ by diamond tiles in the example in Fig. 14, one can also show that there exists a diamond TAS whose final supertile is hexagon-compatible but no hexagonal TAS produces an equivalent final supertile.

Lemmas 6 and 7 together indicate that hexagonal TASs and triangular TASs are, in some sense, not comparable from the point of view of the shapes of the final supertiles they generate.

Let us now compare square with diamond tile assembly systems. For every square TAS S, there is a diamond TAS D such that the final supertiles produced by the systems are equivalent up to an affine transformation on the two

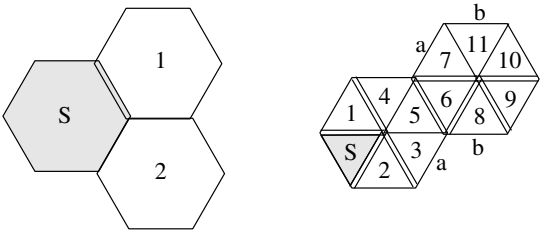


Fig. 14. Examples that show that hexagonal TASs and triangular TASs are, in some sense, not comparable from the point of view of the final supertiles they can generate. The left figure is a hexagonal system at $\tau = 2$ that illustrates Lemma 6. The right figure ($\tau = 2$) illustrates Lemma 7. Each glue, unless mentioned, is unique, and thus, the label is omitted.

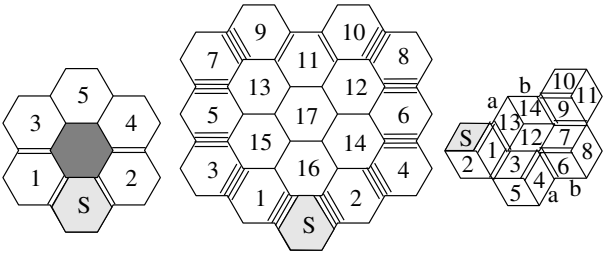


Fig. 15. Examples that show that hexagonal TASs and diamond TASs are, in some sense, not comparable from the point of view of the final supertiles they can generate. The left and center figures depict two hexagonal systems, at $\tau = 2$ and $\tau = 4$ respectively, that illustrate Lemma 8. The right figure ($\tau = 2$) illustrates Lemma 9. Each glue, unless mentioned, is unique, and thus, the label is omitted.

dimensional plane $\mathcal{R}^2$. To see this, we use a single type, either `II`, `IV`, or `VI`, of diamond tile to simulate each square tile. On the other hand, by Lemma 5, every diamond TAS that produces a hexagon-compatible final supertile cannot be simulated by a square TAS.

The comparison of diamond TAS and equilateral TAS is similar to the comparison of square TAS and right triangular TAS: the examples given in Fig. 13, under affine transformations, indicate that diamond TASs and triangular TASs are, in some sense, not comparable from the point of view of the shapes of the final supertiles they generate.

Now we compare diamond TASs with hexagonal TASs. A diamond TAS D is called a *diamond division* of a hexagonal TAS H if *(i)* for any hexagonal tile h in H, there are three diamond tiles in D that can can assemble to produce a supertile equivalent to h at temperature $\tau \geq 1$, and *(ii)* for any diamond tile d in D, there are other two diamond tiles in D that can assemble to produce a supertile tile equivalent to a hexagonal in H (see Fig. 11). A diamond TAS D is

called a *deterministic diamond division* of a hexagonal TAS H if it is a diamond division of H that is a deterministic TAS.

Lemma 8. *There exists a deterministic hexagonal TAS H such that no deterministic diamond division of H produces an equivalent final supertile.*

Two examples that prove the statement of Lemma 8, are illustrated in Fig. 15, one for $\tau = 2$ (left), and one for $\tau = 4$ (centre), none of which can be simulated by their diamond division. Each tile is numbered in the order in which it would appear in a possible assembly process. The two examples are of the same flavor as those in Fig. 13: tile 5, respectively tile 11, cannot be simulated by successive attachments of diamond tiles, since the two cooperative sticky edges that attached the original hexagonal tile to the supertile are not adjacent, hence they will belong to different diamond tiles in the division.

Lemma 9. *There exists a deterministic diamond TAS T whose final supertile is hexagon-compatible, but no deterministic hexagonal TAS produces an equivalent final supertile.*

An example is given in Fig. 15 (right). The proof is similar to that of Lemma 7.

Lemmas 8 and 9 indicate that hexagonal TASs and diamond TASs are, in some sense, not comparable from the point of view of the shapes of the final supertiles they generate. In spite of this, under certain conditions, hexagonal TASs can be simulated by diamond TASs. It is, for example, obvious that for any hexagonal TAS H at $\tau = 1$, there is a diamond division of H that produces a final supertile of the same shape. Furthermore, we have the following result.

Proposition 3. *For any hexagonal TAS H at $\tau = 1$, and any hexagonal TAS at $\tau = 2$, whose final supertile has no hole, there is a diamond division of H that produces a final supertile of the same shape.*

The proof is similar to that of Lemma 2.

4 Conclusion

Square tile assembly systems have been widely studied in the literature as a model, in particular for DNA self-assembly. In this paper, we focus instead on triangular and hexagonal TASs and some of their properties. We show that, in some restricted sense, triangular TASs and square TASs, respectively triangular TASs and hexagonal TASs, are not comparable from the point of view of the shape of the final supertiles they generate. More precisely, there exists a deterministic square (respectively hexagonal) TAS S such that no deterministic triangular division of S produces an equivalent final supertile (of the same shape and same boundary glues). Also, there exists a deterministic triangular TAS T such that the final supertile is square (respectively hexagon) -compatible, but no deterministic square (respectively hexagonal) TAS produces an equivalent final supertile.

Acknowledgements. We thank Dr. David Doty for his valuable comments on the earlier versions of the paper. This research was supported by The Natural Sciences and Engineering Council of Canada Discovery Grant R2824A01 and Canada Research Chair Award to Lila Kari.

References

1. Adleman, L.: Toward a mathematical theory of self-assembly (manuscript, 2000), https://eprints.kfupm.edu.sa/72519/1/72519.pdf
2. Winfree, E., Liu, F., Wenzler, L.A., Seeman, N.C.: Design and self-assembly of two-dimensional DNA crystals. Nature 394, 539–544 (1998)
3. Wang, H.: Proving theorems by pattern recognition II. Bell System Technical Journal 40, 1–42 (1961)
4. Winfree, E.: On the computational power of DNA annealing and ligation. In: DNA Based Computers: DIMACS Workshop, vol. 27, pp. 199–221 (1996)
5. Rothemund, P.W.K., Winfree, E.: The program-size complexity of self-assembled squares. In: Proc. 32nd Ann. ACM Symp. Theor. of Comp. (STOC 2000), pp. 459–468 (2000)
6. Adleman, L., Cheng, Q., Goel, A., Huang, M.: Running time and program size for self-assembled. In: Proc. 33rd Ann. ACM Symp. Theor. of Comp. (STOC 2001), pp. 740–748 (2001)
7. Kao, M., Schweller, R.: Reducing tile complexity for self-assembly through temperature programming. In: Proc. 7th Ann. ACM-SIAM Symp. Discrete Algorithms (SODA), pp. 571–580 (2006)
8. Liu, D., Wang, M., Deng, Z., Walulu, R., Mao, C.: Tensegrity: Construction of rigid DNA triangles with flexible four-arm DNA junctions. J. Am. Chem. Soc. 126, 2324–2325 (2004)
9. Ding, B., Sha, R., Seeman, N.C.: Pseudohexagonal 2D DNA crystals from double crossover cohesion. J. Am. Chem. Soc. 126, 10230–10231 (2004)
10. He, Y., Chen, Y., Liu, H., Ribbe, A.E., Mao, C.: Self-assembly of hexagonal DNA two-dimensional (2D) arrays. J. Am. Chem. Soc. 127, 12202–12203 (2005)
11. Rothemund, P.W.K., Papadakis, N., Winfree, E.: Algorithmic self-assembly of DNA Sierpinski triangles. PLoS Biol. 2, e424 (2004), http://dx.doi.org/10.1371/journal.pbio.0020424

dP Automata versus Right-Linear Simple Matrix Grammars

Gheorghe Păun[1,2] and Mario J. Pérez-Jiménez[2]

[1] Institute of Mathematics of the Romanian Academy
PO Box 1-764, 014700 Bucureşti, Romania
[2] Department of Computer Science and Artificial Intelligence
University of Sevilla
Avda. Reina Mercedes s/n, 41012 Sevilla, Spain
{gpaun,marper}@us.es

Abstract. We consider dP automata with the input string distributed in an arbitrary (hence not necessary balanced) way, and we investigate their language accepting power, both in the case when a bound there is on the number of objects present inside the system and in the general case. The relation with right-linear simple matrix grammars is useful in this respect. Some research topics and open problems are also formulated.

1 Introduction

dP automata are a class of computing devices considered in membrane computing area in order to have a distributed language accepting machinery, with the strings to recognize being split among the components of the system and with these components working in parallel on the input strings. In the general case, dP systems consist of a given number of components in the form of a usual symport/antiport P system, which can have their separate inputs and communicate from skin to skin membranes by means of antiport rules like in tissue-like P systems. Such devices were introduced in [7] and further investigated in [3], [8], [9], mainly comparing their power with that of usual P automata and with families of languages in the Chomsky hierarchy. In the basic definition and in all these papers, following the style of the communication complexity area (see, [4]), the so-called *balanced* mode of introducing the input string is considered: the string is split in equal parts, modulo one symbol, and distributed among components.

Here we consider the general case, with no restriction on the input string distribution; each component just takes symbols from the environment when it can do it, without any restriction on their number. This is a very natural and general set-up, which, however, was only incidentally investigated so far. Two cases are distinguished: with a bound on the size of the system (on the total number of objects present inside) and without such a bound. Both cases are naturally related to a classic family of regulated grammars, the simple matrix grammars of [5] (see also [2]). Actually, as expected, right-linear simple matrix grammars are closely related to dP automata, and we will examine below this

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 376–387, 2012.

connection (looking for mutual simulations among the two types of language identifying machineries). This connection was already pointed out in [8], where the conjecture was formulated that, in the same way as a usual finite automaton can be simulated by a P automaton, a right-linear simple matrix grammar can be simulated by a dP automaton. We confirm here this conjecture (in the general, not the balanced case).

2 Formal Language Theory Prerequisites

The reader is assumed to have some familiarity with basics of membrane computing, e.g., from [6], [10], and of formal language theory, e.g., from [2], [11], but we recall below all notions necessary in the subsequent sections.

In what follows, V^* is the free monoid generated by the alphabet V, λ is the empty word, $V^+ = V^* - \{\lambda\}$, and $|x|$ denotes the length of the string $x \in V^*$. REG, LIN, CF, CS, RE denote the families of regular, linear, context-free, context-sensitive, and recursively enumerable languages, respectively.

In the proof of the main result of the paper we will make an essential use of the right-linear simple matrix grammars introduced in [5]. Such a grammar of degree $n \geq 1$ is a construct of the form $G = (N_1, \ldots, N_n, T, S, M)$, where $N_1, N_2, \ldots, N_n, T$ are pairwise disjoint alphabets (we denote by N the union of $N_1, \ldots, N_n$), $S \notin T \cup N$, and M contains matrices of the following forms:

(i) $(S \to x), x \in T^*$,
(ii) $(S \to A_1 A_2 \ldots A_n),\ A_i \in N_i, 1 \leq i \leq n$,
(iii) $(A_1 \to x_1 B_1, \ldots, A_n \to x_n B_n),\ A_i, B_i \in N_i, x_i \in T^*, 1 \leq i \leq n$,
(iv) $(A_1 \to x_1, \ldots, A_n \to x_n),\ A_i \in N_i, x_i \in T^*, 1 \leq i \leq n$.

A derivation starting with a matrix of type (ii) continues with an arbitrary numbers of steps which use matrices of type (iii) and ends by applying a matrix of type (iv).

We denote by $L(G)$ the language generated in this way by G and by RSM_n the family of languages $L(G)$ for right-linear simple matrix grammars G of degree at most n, for $n \geq 1$. The union of all these families is denoted by RSM_*. The strict inclusions $RSM_n \subset RSM_{n+1}, n \geq 1$, are known. Moreover, $REG = RSM_1$, $RSM_* \subset CS$, RSM_* is incomparable with LIN and CF, all languages in RSM_* are semilinear, and this family is closed under union, intersection with regular languages, direct and inverse morphisms (but not under intersection, complement and Kleene +).

Clearly, a normal form can be easily found for these grammars: in matrices of type (iii) we can ask to have $x_i \in T \cup \{\lambda\}, 1 \leq i \leq n$, and in matrices of type (iv) we can have $x_i = \lambda$ for all $1 \leq i \leq n$.

3 dP Automata

We introduce now the computing devices we investigate in this paper, also giving a relevant example.

As usual in membrane computing, the multisets over an alphabet V are represented by strings in V^*; a string and all its permutations correspond to the same multiset, with the number of occurrences of a symbol in a string representing the multiplicity of that object in the multiset. (We work here only with multisets of finite multiplicity.) The terms "symbol" and "object" are used interchangeably, all objects are here represented by symbols.

A *dP automaton* (of degree $n \geq 1$) is a construct

$$\Delta = (O, E, \Pi_1, \ldots, \Pi_n, R),$$

where:

(1) O is an alphabet (of objects);
(2) $E \subseteq O$ (the objects available in arbitrarily many copies in the environment);
(3) $\Pi_i = (O, \mu_i, w_{i,1}, \ldots, w_{i,k_i}, E, R_{i,1}, \ldots, R_{i,k_i})$ is a symport/antiport P system of degree k_i (O is the alphabet of objects, μ_i is a membrane structure of degree k_i, $w_{i,1}, \ldots, w_{i,k_i}$ are the multisets of objects present in the membranes of μ_i in the beginning of the computation, E is the alphabet of objects present – in arbitrarily many copies – in the environment, and $R_{i,1}, \ldots, R_{i,k_i}$ are finite sets of symport/antiport rules associated with the membranes of μ_i; the symport rules are of the form $(u, in), (u, out)$, where $u \in O^*$, and the antiport rules are of the form $(u, out; v, in)$, where $u, v \in O^*$; note that we do not have an output membrane), with the skin membrane labeled with $(i, 1) = s_i$, for all $i = 1, 2, \ldots, n$;
(4) R is a finite set of rules of the form $(s_i, u/v, s_j)$, where $1 \leq i, j \leq n, i \neq j$, and $u, v \in O^*, uv \neq \lambda$.

The systems $\Pi_1, \ldots, \Pi_n$ are called *components* of Δ and the rules in R are called *communication rules*. For a rule $(s_i, u/v, s_j)$, $|uv|$ is the *weight* of this rule.

Using a rule $(u, in), (u, out)$ associated with a membrane i means to bring in the membrane, respectively to send out of it the multiset u; using a rule $(u, out; v, in)$ associated with a membrane i means to send out of the membrane the objects of multiset u and, simultaneously, to bring in the membrane, from the region surrounding membrane i, the objects of multiset v. A communication rule $(s_i, u/v, s_j)$ moves the objects of u from component Π_i to component Π_j, simultaneously with moving the objects in the multiset v in the opposite direction.

Each component Π_i can take symbols from the environment, work on them by using the rules in sets $R_{i,1}, \ldots, R_{i,k_i}$, and communicate with other components by means of rules in R.

A halting computation with respect to Δ accepts the string $x = x_1 x_2 \ldots x_n$ over O if the components $\Pi_1, \ldots, \Pi_n$, starting from their initial configurations, using the symport/antiport rules as well as the inter-components communication rules, in the non-deterministic maximally parallel way, bring from the environment the substrings $x_1, \ldots, x_n$, respectively, and eventually halts. A problem appears in the case when several objects are read at the same time from the environment, by several rules or by a single rule of the form $(u, out; v, in)$, with

$|v| \geq 2$; in this case any permutation of the symbols brought in the system in the same step are considered as a valid substring of the input string (thus, a computation can recognize several strings, differing to each other by permutations of certain substrings). Note that we impose here no condition on the relative lengths of strings $x_1, x_2, \ldots, x_n$ (as it is done in previous papers dealing with dP automata, under the influence of communication complexity area). We denote by $L(\Delta)$ the language of all strings recognized by Δ in this way, and by LdP_n the family of languages $L(\Delta)$, for Δ of degree at most $n \geq 1$. The union of all these families is denoted by LdP_*.

The dP automata are synchronized devices, a universal clock exists for all components, marking the time in the same way for the whole dP automaton. When the system has only one component, then we obtain the usual notion of a P automaton, as investigated in a series of papers (mainly in the extended version, with a terminal alphabet of objects – see the respective chapter in [10] and the references therein). We denote by LP the family of languages recognized by P automata. Hence, $LP = LdP_1$ and, from [3], it is known that $REG \subset LP \subset CS$ and LP is incomparable with CF.

We consider now a somewhat surprising example, of a dP automaton of degree 2, generating a complex language, $L_1 = \{ww \mid w \in \{a, b\}^*\}$. The automaton is given in Figure 1, in the standard way of representing a dP automaton. We have $O = \{a, b, c_1, c_2, d, \#\}$ and $E = \{a, b\}$.

All antiport rules which bring objects from the environment are of weight one, hence the number of objects present in the system is constant, four in each component. In the first step, objects d release c_2a in the skin region of the first component and c_1a in the second. Each symbol a can bring either an a or a b from the environment and, at the same time, the objects c_1, c_2 are interchanged between the two components (otherwise, they release the trap object $\#$, which will oscillate forever across membranes $(1, 1)$, respectively, $(2, 1)$, and the computation never stops). With $c_1\alpha, \alpha \in \{a, b\}$, in the first component and $c_2\beta, \beta \in \{a, b\}$, in the second one, the only continuation which does not release the trap object is possible when $\alpha = \beta$, by using the communication rule $(s_1, c_1\alpha/c_2\alpha, s_2)$ (if one of the symbols α, β brings new symbols from the environment, the corresponding c_1, c_2 should enter the membrane $(1, 2)$ or $(2, 2)$, bringing out the object $\#$). We obtain a configuration similar to the one we started with, hence the process can be iterated. If, at any moment when c_2 is in Π_1 and c_1 is in Π_2, one of the rules $(c_2\alpha, in), \alpha \in \{a, b\}$, is used in the first component, or $(c_1\beta, in), \beta \in \{a, b\}$, is used in the second component, then this should be done simultaneously in both components, otherwise again one of c_1, c_2 has to release the trap object. In conclusion, the strings read from the environment by the two components are identical, hence $L(\Delta) = L_1$.

Note the important facts that the system reads the input in a balanced way and that it is *bounded*, the total number of objects present inside is always bounded by a constant (8 in our case) given in advance. This last characteristic is important, so that we denote by $LdP_n^b, n \geq 1$, the family of languages recognized by bounded dP automata of degree at most n; when n is not specified, we replace it by $*$.

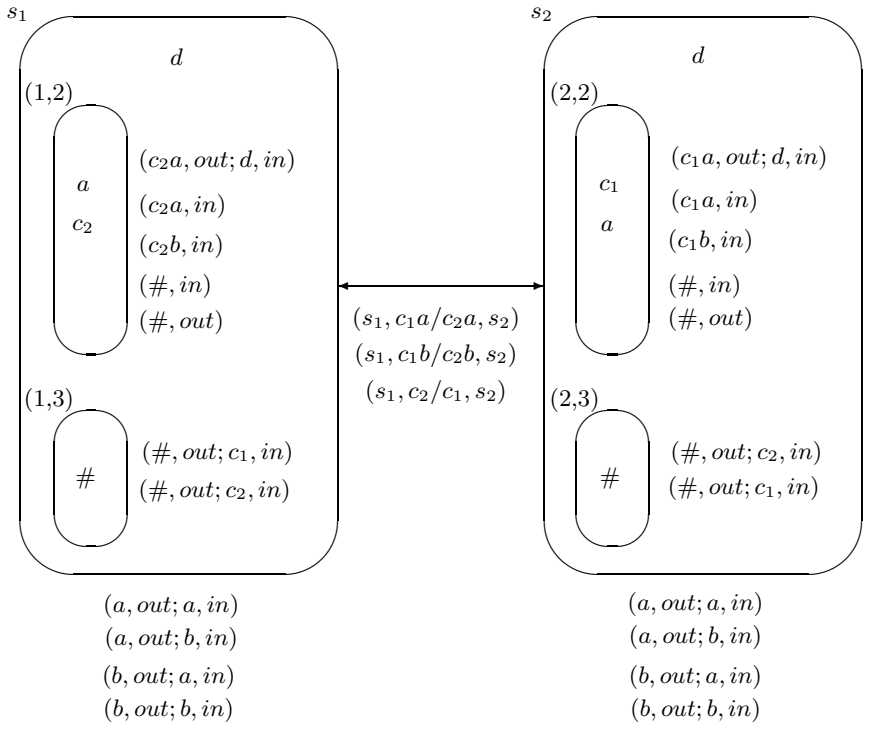


Fig. 1. A dP automaton recognizing the language L_1

4 The Power of dP Automata

We start by reformulating in a more general way a result already suggested by a proof in [9].

Theorem 1. *$LdP_n^b \subseteq RSM_n$, for all $n \geq 1$.*

Proof. Let Δ be a dP automaton of degree n (with the set of objects O) which is bounded. Then, the set of all its configurations is finite. Let $\{\sigma_0, \sigma_1, \ldots, \sigma_p\}, p \geq 0$, be this set, with σ_0 being the initial configuration. We construct the following right-linear simple matrix grammar:

$$
\begin{aligned}
&G = (N_1, \ldots, N_n, O, S, M), \text{ with}\\
&N_i = \{(\sigma_j)_i \mid 0 \leq j \leq p\},\ i = 1, 2, \ldots, n,\\
&M = \{(S \to (\sigma_0)_1(\sigma_0)_2 \ldots (\sigma_0)_n)\}\\
&\quad \cup \{(\sigma_i)_1 \to \alpha_1(\sigma_j)_1, \ldots, (\sigma_i)_n \to \alpha_n(\sigma_j)_n) \mid\\
&\qquad \text{from configuration } \sigma_i \text{ the dP automaton } \Delta \text{ can pass to}\\
&\qquad \text{the configuration } \sigma_j \text{ by a correct transition, taking from the}
\end{aligned}
$$

environment the objects $\alpha_1, \ldots, \alpha_n$ by its components, where $\alpha_s \in O \cup \{\lambda\}, 1 \leq s \leq n\}$
$\cup \{(\sigma_h)_1 \to \lambda, \ldots, (\sigma_h)_n \to \lambda) \mid \sigma_h$ is a halting configuration$\}$.

Note that all nonterminals in the rules of a matrix contain the same "core information", namely the current configuration of the system, hence the complete control of the system working is obtained in this way. The equality $L(\Delta) = L(G)$ is obvious. □

This result cannot be extended to arbitrary dP automata. Actually, we have:

Theorem 2. $LdP_2 - RSM_* \neq \emptyset$.

Proof. Let us consider the following dP automaton:

$$\begin{aligned}
\Delta &= (O, E, \Pi_1, \Pi_2, R), \text{ with} \\
O &= \{a, c, d, e, f, \#\}, \\
E &= \{a, c, d, e\}, \\
\Pi_1 &= (O, [\ \ ]_{s_1}, f, E, \{(f, out; a, in),\ (a, out; aa, in)\}), \\
\Pi_2 &= (O, [\ [\ \ ]_{(2,2)}\ ]_{s_2}, E, \{(f, out; d, in),\ (a, out; c, in),\ (d, out; e, in)\}, \\
&\qquad\qquad \{(f, out; f, in)\}), \\
R &= \{(s_1, a/\lambda, s_2)\}.
\end{aligned}$$

For an easier examination of the work of the system, we also represent it graphically, in Figure 2.

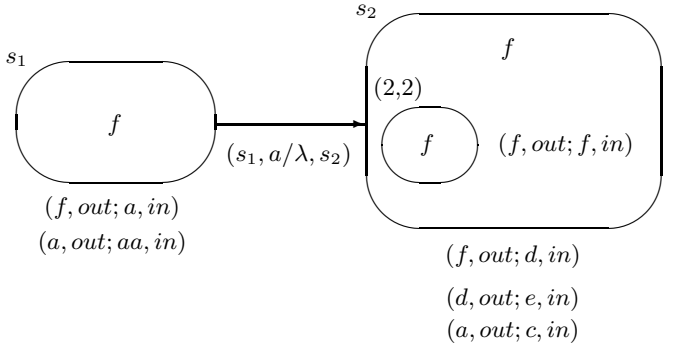


Fig. 2. A dP system recognizing a language not in RSM_*

Let us look for strings accepted by this dP automaton which are of the form $a^i dc^j e$, for some $i, j \geq 1$.

After introducing the symbol a in the first component, let us assume that for $n \geq 0$ steps we use here the rule $(a, out; aa, in)$, hence we produce 2^n copies of a in Π_1, while the second component uses the rule $(f, out; f, in) \in R_{(2,1)}$. Suppose now that $p \geq 0$ copies of a remains in the first component and the remaining $r = 2^n - p$ copies of a are moved to the second component. Here, all r copies of a must go out, in exchange of objects c, hence the string read by the second component starts with c^r. At the same time or one step before, the second component must introduce the symbol d. This object becomes immediately e, hence the exchange of a for c should be done either in the same step with reading d or at the same time with reading e in the second component (because any permutation of the objects is allowed in the string, either variant is possible). However, after e, we do not want to have any symbol, hence all copies of a were already moved to the second component, and thus the work of the first component stops. When introducing the symbol d in the second component, the p copies of a from the first component cannot use the rule $(a, out; aa, in)$, but they must come immediately in the second component, to introduce c here at the same time with introducing e. Therefore, if the string has the form $a^i dc^j e$, then $i = j = 2^n$ for some $n \geq 0$ ($n = 0$ is obtained if the unique a introduced in the first step in Π_1 is immediately sent to component Π_2).

Consequently, $L(\Delta) \cap a^*dc^*e = \{a^{2^n} dc^{2^n} e \mid n \geq 0\}$, which is not in RSM_*, hence also $L(\Delta)$ is not in RSM_*: this family is closed under intersection with regular languages and contains only semilinear languages. □

Note that the previous construction takes the input string in an almost balanced way, and, if in the first step, the first component uses a rule $(f, out; dea, in)$ instead of $(f, out; a, in)$, then we have a balanced input, hence the result in the previous theorem holds true also for the balanced way of defining the recognized string.

We pass now to the counterpart of Theorem 1 announced above.

Theorem 3. *$RSM_n \subseteq LdP^b_{n+1}$, for all $n \geq 1$.*

Proof. Let us consider a right-linear simple matrix grammar $G = (N_1, \ldots, N_n, T, S, M)$ as introduced in Section 2, with the alphabets $N_1, N_2, \ldots, N_n$ (their union is denoted by N) and T. Matrices of the form (i), $(S \to x), x \in T^*$, can be replaced by matrices of forms (ii), (iii) and (iv), in an obvious way, hence we assume that we do not have such matrices. We assume all matrices are labeled in a one-to-one manner; let $m_j : (A_1 \to x_1 B_1, \ldots, A_n \to x_n B_n)$, with $1 \leq j \leq k$, be all matrices of type (iii), with $A_i, B_i \in N_i, x_i \in T^*, 1 \leq i \leq n$. Similarly, let $m_j : (A_1 \to x_1, \ldots, A_n \to x_n)$, with $k+1 \leq j \leq p$, be all matrices of type (iv), with $A_i \in N_i, x_i \in T^*, 1 \leq i \leq n$. Without any loss of the generality we can assume that all strings x_i in these matrices are from $T \cup \{\lambda\}$.

For each matrix, of any form, $m_j : (A_1 \to u_1, \ldots, A_i \to u_i, \ldots, A_n \to u_n)$, let us consider the symbol $[m_j, A_i \to u_i]$ (thus identifying the matrix and its ith rule), and let $X_j(i)$ be a shorthand for it. Consider the alphabets

$$M_i = \{X_j(i) \mid 1 \leq j \leq p\}, \text{ for all } 1 \leq i \leq n.$$

We also denote by M_i' the alphabet of primed symbols in M_i.

For a matrix $m_j : (A_1 \to x_1 B_1, \ldots, A_n \to x_n B_n)$ of type (iii), let us denote $lhs_j = A_1 A_2 \ldots A_n$ and $rhs_j = B_1 B_2 \ldots B_n$. Similarly, for a matrix $m_j : (A_1 \to x_1, \ldots, A_n \to x_n)$ of type (iv), we denote $lhs_j = A_1 A_2 \ldots A_n$.

If $rhs_j = lhs_k$, then we write $m_j \rightsquigarrow m_k$. Similarly, we write $S \rightsquigarrow m_j$ if $(S \to A_1 A_2 \ldots A_n) \in M$ and $A_1 A_2 \ldots A_n = lhs_j$.

Any set Q can be also considered a multiset (denoted again by Q) consisting of the elements of Q with the multiplicity one for each of them.

We are now ready to construct the dP system we look for (a_0 is an arbitrary symbol of T fixed in advance):

$$
\begin{aligned}
\Delta &= (O, E, \Pi_1, \ldots, \Pi_{n+1}, R), \text{ with :}\\
O &= \bigcup_{i=1}^{n} (M_i \cup M_i') \cup T \cup \{c_i \mid 1 \le i \le n\} \cup \{d, f, \#\},\\
E &= T,\\
\Pi_i &= (O, [\,[\quad]_{(i,2)}[\quad]_{(i,3)}\,]_{s_i}, \lambda, M_i' T c_i, \#, R_{s_i}, R_{(i,1)}, R_{(i,2)}),\\
&\quad R_{s_i} = \{(a, out; b, in) \mid a, b \in T\},\\
&\quad R_{(i,2)} = \{(X_j'(i), out; X_j(i), in),\\
&\quad (X_j(i) c_i a, out; X_j'(i) c_i a, in) \mid 1 \le j \le p,\\
&\quad \text{if } X_j(i) = [m_j, A_i \to a B_i], a \in T\}\\
&\quad \cup \{(X_j'(i) a, out; X_j(i) a, in),\\
&\quad (X_j(i) c_i a, out; X_j'(i) c_i a, in) \mid 1 \le j \le p, a \in T,\\
&\quad \text{if } X_j(i) = [m_j, A_i \to B_i]\}\\
&\quad \cup \{(\#, in),\ (\#, out)\},\\
&\quad R_{(i,3)} = \{(\#, out; c_i, in)\}\\
&\quad \cup \{(\#, out; X_j(i), in) \mid 1 \le j \le p, \text{ if } X_j(i) = [m_j, A_i \to B_j]\},\\
&\quad \text{for all } 1 \le i \le n,\\
\Pi_{n+1} &= (O, [\,[\ ]_{(n+1,2)}\,]_{s_{n+1}}, c_1 \ldots c_n f, M_1^2 \ldots M_n^2 T^n a_0^n, \emptyset, R_{(n+1,1)}),\\
&\quad R_{(n+1,2)} = \{(X_j(1) \ldots X_j(n) a_0^n, out; f, in) \mid 1 \le j \le p \text{ if } S \rightsquigarrow m_j\}\\
&\quad \cup \{(X_k(1) a_1 \ldots X_k(n) a_n, out; X_j(1) a_1 \ldots X_j(n) a_n, in)\\
&\quad \mid 1 \le j, k \le p, a_i \in T, 1 \le i \le n, \text{ if } m_j \rightsquigarrow m_k\}\\
&\quad \cup \{(X_j(1) c_1 a_1 \ldots X_j(n) c_n a_n, in)\\
&\quad \mid a_i \in T, 1 \le i \le n, \text{ if } m_j \text{ is a terminal matrix}\},\\
R &= \{(s_i, \lambda / c_i, s_{n+1}),\\
&\quad (s_i, c_i / X_j(i) a, s_{n+1},\\
&\quad (s_i, X_j(i) c_i a / \lambda, s_{n+1}) \mid 1 \le j \le p, 1 \le i \le n, a \in T\}.
\end{aligned}
$$

This dP system, with one component Π_i and with Π_{n+1} given in full details, is represented in Figure 3.

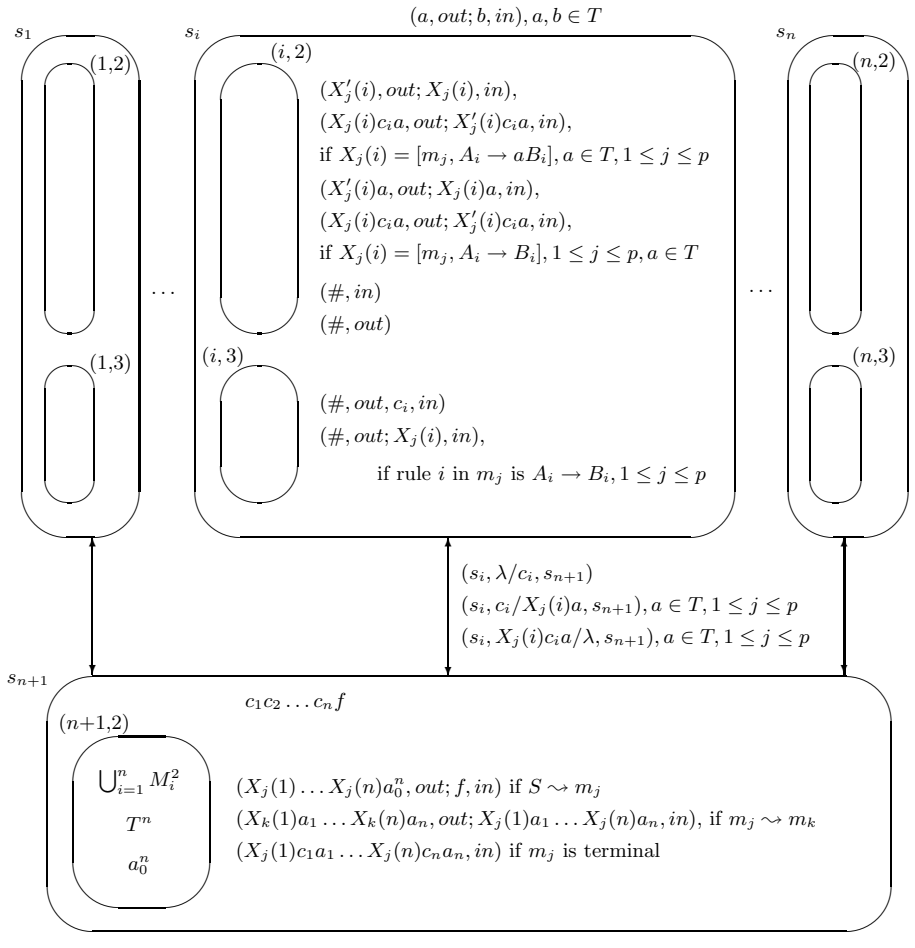


Fig. 3. The dP system in the proof of Theorem 3

The components $\Pi_i, 1 \le i \le n$, simulate the corresponding "component" of the grammar G, while Π_{n+1} is a "synchronizer" of the other components, it takes no objects from the environment. All rules which bring objects from the environment are uniport rules, hence the system is bounded, the number of objects inside it remains constant during the computation.

We start by sending objects c_i from Π_{n+1} to components Π_i, simultaneously releasing from membrane $(n+1, 2)$ some objects $X_j(i), 1 \le i \le n$, for a matrix m_j which can follow immediately after an initial matrix of G; each symbol $X_j(i)$ is accompanied by a copy of the symbol a_0, arbitrarily chosen from T.

In the next step, we have to exchange the symbol c_i from Π_i with $X_j(i)a_0$ from Π_{n+1} (if c_i remains unused in Π_i, then it will release the trap object $\#$ from membrane $(i, 2)$, and the computation will never halt).

In the next step, c_i comes back to Π_i, and in this component we have two possibilities:

(1) The rule i from m_j is of the form $A_i \to aB_i$, and then we use a rule $(a_0, out; b, in)$, for some $b \in T$, and $(X'_j(i), out; X_j(i), in)$.

Now, we check whether the simulation of the rule in G is correct (hence b was the correct symbol to take from the environment, i.e., $a = b$): c_i cannot return to Π_{n+1} alone and cannot stay unused in Π_i. The only continuation which does not lead to an infinite computation is to use the rule $(X_j(i)c_ia, out; X'_j(i)c_ia, in)$. These three objects, $X_j(i)c_ia$, can now move together to Π_{n+1}. The only continuation is to move again c_i in Π_i, for all i, and to exchange $X_j(1) \dots X_j(n)$ for some $X_k(1) \dots X_k(n)$ in Π_{n+1}, for $m_j \rightsquigarrow m_k$.

We return in this way to a situation similar to that we have started with: object c_i in Π_i and $X_k(i)$ in Π_{n+1}.

(2) If the rule i from m_j is of the form $A_i \to B_i$, and we use a rule $(a_0, out; b, in)$, for some $b \in T$, then the computation will never stop: we do not have a rule for introducing $X_j(i)$ alone in membrane $(i, 2)$, hence $X_j(i)$ must release the trap object from membrane $(i, 3)$. Therefore, we have to use the rule $(X'_j(i)a, out; X_j(i)a, in)$ from $R_{(i,2)}$ (at the same time, the object c_i comes to Π_i). As above, the three objects $X_j(i)c_ia$ can move together to Π_{n+1}, where, while c_i moves to Π_i, we exchange $X_j(1) \dots X_j(n)$ for some $X_k(1) \dots X_k(n)$ in Π_{n+1}, for $m_j \rightsquigarrow m_k$.

Also in this case we return to a situation similar to that we have started with: object c_i in Π_i and $X_k(i)$ in Π_{n+1}.

The process can be continued. Checking the correctness of the simulation of the rules in G is done in components Π_i, the fact that the rules which are simultaneously checked form a matrix of G is ensured by the component Π_{n+1}.

When a terminal matrix is simulated, component Π_{n+1} halts the computation by using the rule $(X_j(1)c_1a_1 \dots X_j(n)c_na_n, in)$ (if we do not "hide" also the objects c_i in membrane $(n+1, 2)$, then these objects have to go to components Π_i, where no rule can use them other than the trap-releasing ones).

We conclude that $L(G) = L(\Delta)$. □

5 Final Remarks

Let us first synthesize all previous results and remarks in a diagram – see Figure 4. The arrows indicate inclusions; if the arrow is marked with a dot, then that inclusion is known to be proper. The inclusions $RSM_n \subset RSM_{n+1}, n \geq 1$, are known to be proper, hence also the hierarchy $LdP^b_n, n \geq 1$, is infinite, but we do not know languages proving the strictness of inclusions $LdP^b_n \subseteq RSM_n \subseteq LdP^b_{n+1}, n \geq 1$, with the exception of the inclusion $RSM_1 \subset LdP^b_2$, because $RSM_1 = REG$ and LdP^b_2 contains non-regular languages (see, e.g., the example in Section 3). Similarly, we do not know whether the inclusions $LdP_n \subseteq LdP_{n+1}, n \geq 2$, are proper – but we *conjecture* that this is the case.

Further open problems and research topics about dP systems can be found in [1], [3], [7], [8], [9] – the study of dP automata is one of the recently introduced and most active branches of membrane computing.

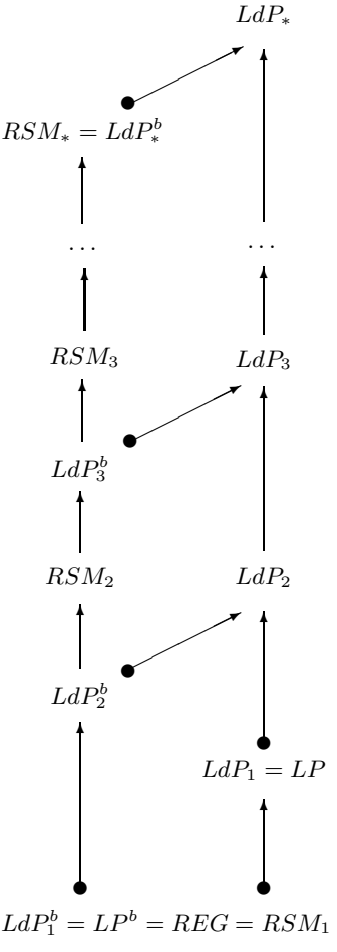


Fig. 4. The hierarchy of the families RSM_n, LdP_n^b, and LdP_n

Acknowledgements. Work supported by Proyecto de Excelencia con Investigador de Reconocida Valía, de la Junta de Andalucía, grant P08 – TIC 04200.

References

1. Csuhaj-Varju, E., Vaszil, G.: A connection between finite dP automata and multihead finite automata. In: Proc. Twelfth Intern. Conf. of Membrane Computing, CMC 12, Fontainebleau, France, pp. 109–126 (August 2011)
2. Dassow, J., Păun, G.: Regulated Rewriting in Formal Language Theory. Springer, Berlin (1989)
3. Freund, R., Kogler, M., Păun, G., Pérez-Jiménez, M.J.: On the power of P and dP automata. Annals of Bucharest University. Mathematics-Informatics Series 63, 5–22 (2009)

4. Hromkovic, J.: Communication Complexity and Parallel Computing: The Application of Communication Complexity in Parallel Computing. Springer, Berlin (1997)
5. Ibarra, O.: Simple matrix grammars. Information and Control 17, 359–394 (1970)
6. Păun, G.: Membrane Computing. An Introduction. Springer, Berlin (2002)
7. Păun, G., Pérez-Jiménez, M.J.: Solving problems in a distributed way in membrane computing: dP systems. Int. J. of Computers, Communication and Control 5(2), 238–252 (2010)
8. Păun, G., Pérez-Jiménez, M.J.: P and dP Automata: A Survey. In: Calude, C.S., Rozenberg, G., Salomaa, A. (eds.) Maurer Festschrift. LNCS, vol. 6570, pp. 102–115. Springer, Heidelberg (2011)
9. Păun, G., Pérez-Jiménez, M.J.: An infinite hierarchy of languages defined by dP Systems. Theoretical Computer Sci. (in press)
10. Păun, G., Rozenberg, G., Salomaa, A. (eds.): Handbook of Membrane Computing. Oxford University Press (2010)
11. Rozenberg, G., Salomaa, A. (eds.): Handbook of Formal Languages, vol. 3. Springer, Berlin (1998)
12. The P Systems Website, http://ppage.psystems.eu

State Complexity of Kleene-Star Operations on Trees*

Xiaoxue Piao and Kai Salomaa

School of Computing, Queen's University, Kingston, Ontario K7L 3N6, Canada
{piao,ksalomaa}@cs.queensu.ca

Abstract. The concatenation of trees can be defined either as a sequential or a parallel operation, and the corresponding iterated operation gives an extension of Kleene-star to tree languages. Since the sequential tree concatenation is not associative, we get two essentially different iterated sequential concatenation operations that we call the bottom-up star and top-down star operation, respectively. We establish that the worst-case state complexity of bottom-up star is $(n + \frac{3}{2}) \cdot 2^{n-1}$. The bound differs by an order of magnitude from the corresponding result for string languages. The state complexity of top-down star is similar as in the string case. The iteration of the parallel concatenation has to be defined slightly differently in order to yield a regularity preserving operation.

Keywords: tree automata, state complexity, iterated concatenation.

1 Introduction

The state complexity of regular languages has been studied for over half a century, and especially the last two decades have seen much fruitful work on descriptional complexity of finite automata and related computational models. For example, the state complexity of combined operations has been investigated by Domaratzki and Okhotin [5], Gao and Yu [8], and A. Salomaa et al. [21,22]. Descriptional complexity of regular expressions was studied by Ellul et al. [6] and Gruber and Holzer [10]. Calude et al. [1,2] have considered the state complexity of finite transducers from an algorithmic information theory perspective, while Jirásková and Okhotin [13] and Kapoutsis [14] have studied the state complexity of two-way finite automata. Good general references on descriptional complexity of finite automata include the recent survey by Holzer and Kutrib [11], the textbook by Shallit [24] and the handbook article by Yu [25].

The state complexity of tree automata has been considered by Marten and Niehren [15] and by the current authors [17,19] and, in particular, we gave tight state complexity bounds for the concatenation of regular tree languages in [16].

Concatenation of tree languages can be defined either as a sequential or a parallel operation. Here we consider iterated concatenation of trees, that is, an extension of the Kleene-star operation for tree languages. If defined in the usual

* In Honor of Cristian Calude's 60th Birthday.

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 388–402, 2012.

way, iterated parallel concatenation is not a regularity preserving operation and the Kleene-star of tree languages is defined slightly differently in [9]. Since sequential concatenation of tree languages is non-associative, there are two essentially different ways to define the corresponding iterated operation. We name these variants as the *bottom-up star* and the *top-down star* operations. It is easy to see that the top-down (sequential) star operation coincides with the iterated product (Kleene-star) based on parallel concatenation considered in [9].

We give tight state complexity bounds for both bottom-up and top-down Kleene-star operations. We show that the bottom-up star of a tree language recognized by a deterministic bottom-up automaton with n states can be recognized by an automaton with $(n + \frac{3}{2}) \cdot 2^{n-1}$ states and, furthermore, there exist worst-case examples where this number of states is needed. This bound is, roughly, n times the corresponding bound for regular string languages. On the other hand, the state complexity of the top-down star operation is shown to coincide with the state complexity of Kleene-star on string languages.

Much of the recent work on tree automata uses automata operating on unranked trees that are used in modern applications such as XML document processing [3,4,16,17,23]. The transitions of an unranked tree automaton A are defined in terms of regular languages, called *horizontal languages.* Each horizontal language is specified by a deterministic finite automaton (DFA) that processes strings of states of the bottom-up computation, or *vertical states.* The size of A is defined to be the sum of the number of vertical states and the numbers of states of the DFAs used to define the horizontal languages. Interestingly, the minimization of deterministic unranked tree automata is intractable [15], and the minimal automaton for a tree language is not necessarily the automaton with the smallest possible number of bottom-up (or, vertical) states [19]. An alternative syntactic definition of determinism for unranked tree automata that guarantees that the minimal automaton is unique was considered in [4].

In the case of the Kleene-star operations, the worst-case state complexity bounds for the numbers of vertical states can be reached using just binary trees, and for the sake of readability we restrict here consideration to automata operating on ranked trees. An early version of this paper contains the corresponding constructions for unranked tree automata [18]. The upper bound construction in the case of unranked trees relies on the same ideas as Lemma 4.1 below, however, the notations are considerably more involved in the case of unranked trees.

To conclude this section, we include some comments on our choice to use incomplete deterministic automata. In the case of DFAs operating on strings, it is common to give state complexity bounds in terms of complete DFAs, that is, all transitions of a DFA are required to be defined, see e.g. [11,25]. In order to keep our state complexity bounds consistent with corresponding results for tree automata operating on unranked trees [3,16,17], our definition allows a deterministic tree automaton to have undefined transitions.

Note that requiring a ranked tree automaton (or an ordinary DFA) to be complete, changes the number of states by at most one. On the other hand, for deterministic tree automata operating on unranked trees where the horizontal

languages are defined by DFAs [3,16,17], the sizes of an incomplete deterministic automaton and the corresponding completed version may be significantly different. In an unranked tree automaton, adding a dead state q_{sink} for the bottom-up computation, requires the addition, corresponding to an input symbol σ, a horizontal language $L_{\sigma,q_{\text{sink}}}$ that is the complement of a finite disjoint union $L_{\sigma,q_1} \cup \ldots \cup L_{\sigma,q_n}$, where $q_1, \ldots, q_n$ are the vertical states of the incomplete automaton. The size of the minimal DFA for $L_{\sigma,q_{\text{sink}}}$ may be considerably larger than the sum of the sizes of the DFAs for L_{σ,q_i}, $i = 1, \ldots, n$, [12].

2 Basic Definitions on Tree Automata

We assume that the reader is familiar with the basics of automata and formal languages, a good general reference is the handbook by Rozenberg and A. Salomaa [20]. Here we recall and introduce some definitions related to tree automata, for more information see the electronic book by Comon et al. [3] or the handbook article by Gécseg and Steinby [9].

The cardinality of a finite set S is $|S|$ and the power set of S is 2^S. The set of positive integers is $\mathbb{N}$. A ranked alphabet is a finite set Σ where each element is associated a nonnegative integer as its rank. The set of elements of rank m is Σ_m, $m \geq 0$. The set of trees over ranked alphbet Σ, or Σ-trees, F_Σ, is the smallest set S satisfying the condition: if $m \geq 0$, $\sigma \in \Sigma_m$ and $t_1, \ldots, t_m \in S$ then $\sigma(t_1, \ldots, t_m) \in S$.

A *tree domain* is a prefix-closed subset D of $\mathbb{N}^*$ such that if $ui \in D$, $u \in \mathbb{N}^*$, $i \in \mathbb{N}$ then $uj \in D$ for all $1 \leq j < i$. The set of nodes of a tree $t \in F_\Sigma$ can be represented in the well-known way as a tree domain $\text{dom}(t) \subseteq \{1, \ldots, M\}^*$ where M is the largest rank of any element of the ranked alphabet Σ. The tree t is then viewed as a mapping $t : \text{dom}(t) \to \Sigma$.

We assume that notions such as the *root,* a *leaf,* a *subtree* and the *height* of a tree are known. We use the convention that the height of a single node tree is zero. For $\sigma \in \Sigma$ and $t \in F_\Sigma$, $\text{leaf}(t, \sigma) \subseteq \text{dom}(t)$ denotes the set of leaves of t with label σ. Let t be a tree and u some node of t. The tree obtained from t by replacing the subtree at node u with a tree s is denoted $t(u \leftarrow s)$. The notation is extended in the natural way for a set of pairwise independent nodes U of t and $S \subseteq F_\Sigma$: $t(U \leftarrow S)$ is the set of trees obtained from t by replacing each node of U by some tree in S.

The set of Σ-trees where exactly one leaf is labelled by a special symbol x ($x \notin \Sigma$) is $F_\Sigma[x]$. For $t \in F_\Sigma[x]$ and $t' \in F_\Sigma$, $t(x \leftarrow t')$ denotes the tree obtained from t by replacing the unique occurrence of variable x by t'.

A *deterministic bottom-up tree automaton* (DTA) is a tuple $A = (\Sigma, Q, Q_F, g)$, where Σ is a ranked alphabet, Q is a finite set of states, $Q_F \subseteq Q$ is a set of accepting states and g associates to each $\sigma \in \Sigma_m$ a partial function $\sigma_g : Q^m \longrightarrow Q$, $m \geq 0$. In the usual way, we define the state $t_g \in Q$ reached by A at the root of a tree $t = \sigma(t_1, \ldots, t_m)$, $\sigma \in \Sigma_m$, $m \geq 0$, $t_i \in F_\Sigma$, $i = 1, \ldots, m$, inductively by setting $t_g = \sigma_g((t_1)_g, \ldots, (t_m)_g)$ if the right side is defined, and

t_g is undefined otherwise. The tree language recognized by A is $L(A) = \{t \in F_\Sigma \mid t_g \in Q_F\}$. Deterministic bottom-up tree automata recognize the family of regular tree languages.

The intermediate stages of a computation of A, called *configurations of A,* are Σ-trees where some leaves may be labeled by states of A. The set of configurations of A consists of Σ^A-trees where $\Sigma_0^A = \Sigma_0 \cup \{Q\}$ and $\Sigma_m^A = \Sigma_m$ when $m \geq 1$.

A bottom-up automaton begins processing the tree from the leaves because, following a common custom, we view trees to be drawn with the root at the top. As discussed in the previous section, our definition allows a DTA to have undefined transitions, that is, σ_g, $\sigma \in \Sigma_m$, is a partial function.

3 Concatenation and Iterated Concatenation of Trees

We extend the string concatenation operation to an operation where a leaf of a tree is replaced by another tree. Concatenation of trees can be defined also as a parallel operation, however, as will be observed below the iteration of parallel concatenation does not preserve recognizability.

For $\sigma \in \Sigma_0$ and $t_1, t_2 \in F_\Sigma$, we define the *sequential σ-concatenation* of t_1 and t_2 as

$$t_1 \cdot_\sigma^s t_2 = \{\, t_2(u \leftarrow t_1) \mid u \in \mathrm{leaf}(t_2, \sigma) \,\}. \tag{1}$$

That is, $t_1 \cdot_\sigma^s t_2$ is the set of trees obtained from t_2 by replacing one occurrence of a leaf labeled by σ with t_1. The definition is extended in the natural way for tree languages $T_1, T_2 \subseteq F_\Sigma$ by setting

$$T_1 \cdot_\sigma^s T_2 = \bigcup_{t_i \in T_i, i=1,2} t_1 \cdot_\sigma^s t_2.$$

Alternatively, we can consider a *parallel σ-concatenation* of tree languages $T_1, T_2 \subseteq F_\Sigma$ by setting

$$T_1 \cdot_\sigma^p T_2 = \{\, t_2(\mathrm{leaf}(t_2, \sigma) \leftarrow T_1) \mid t_2 \in T_2 \,\}.$$

The operation $T_1 \cdot_\sigma^p T_2$ is called the σ-product of T_1 and T_2 in [9]. Note that the parallel concatenation of tree languages could not be defined by defining first the concatenation of individual trees (as was done for sequential concatenation in (1)) and then taking union over sets of trees. For trees $t_1, t_2 \in F_\Sigma$, $t_1 \cdot_\sigma^p t_2$ is an individual tree while $t_1 \cdot_\sigma^s t_2$ is a set of trees. In the case where no leaf of t_2 is labeled by σ, $t_1 \cdot_\sigma^s t_2 = \emptyset$ and $t_1 \cdot_\sigma^p t_2 = t_2$.

When considering bottom-up tree automata operating on unary trees, both of the above definitions reduce to the usual concatenation of string languages: when processing $T_1 \circ T_2$, $\circ \in \{\cdot_\sigma^s, \cdot_\sigma^p\}$, the automaton reads first an element of T_1 and then an element of T_2.

The parallel concatenation operation is associative, however, sequential concatenation is nonassociative, as observed below in Example 3.1. The nonassociativity of sequential concatenation means, in particular, that there are two variants of the iteration of the operation.

For $\sigma \in \Sigma$ and $T \subseteq F_\Sigma$, we define the *kth sequential top-down σ-power of T*, $k \geq 0$, by setting $T_\sigma^{s,t,0} = \{\sigma\}$, and $T_\sigma^{s,t,k} = T \cdot_\sigma^s T_\sigma^{s,t,k-1}$, when $k \geq 1$. The *sequential top-down σ-star of T* is then

$$T_\sigma^{s,t,*} = \bigcup_{k \geq 0} T_\sigma^{s,t,k}.$$

Similarly, the *kth sequential bottom-up σ-power of T*, is defined by setting $T_\sigma^{s,b,0} = \{\sigma\}$, $T_\sigma^{s,b,1} = T$ and $T_\sigma^{s,b,k} = T_\sigma^{s,b,k-1} \cdot_\sigma^s T$, when $k \geq 2$. The *sequential bottom-up σ-star of T* is

$$T_\sigma^{s,b,*} = \bigcup_{k \geq 0} T_\sigma^{s,b,k}.$$

Note that the definition of bottom-up σ-powers explicitly sets $T_\sigma^{s,b,1}$ to be equal to T. This is done because $T_\sigma^{s,b,0} \cdot_\sigma^s T$ can be a strict subset of T if some trees of T contain no occurrences of σ. Figure 1 illustrates the definitions of top-down star and bottom-up star.

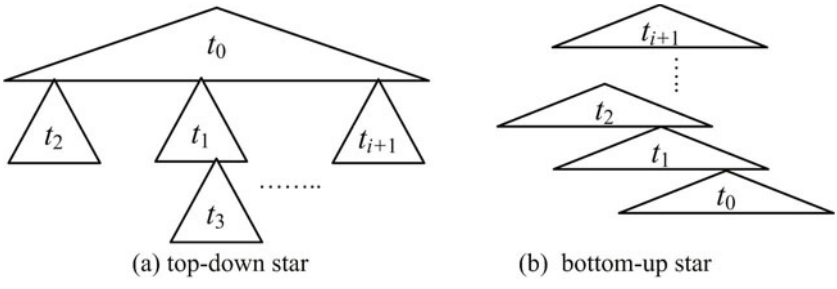


Fig. 1. A tree in $T_\sigma^{s,t,*}$ (a) and in $T_\sigma^{s,b,*}$ (b). Here $t_0, t_1, \ldots t_{i+1}$ are trees in T.

Example 3.1. It is easy to see that sequential concatenation is non-associative. Consider a ranked alphabet Σ determined by $\Sigma_2 = \{\omega\}$, $\Sigma_0 = \{\sigma\}$ and let $t = \omega(\sigma, \sigma)$. Now $t \cdot_\sigma^s t = \{\omega(\omega(\sigma, \sigma), \sigma), \omega(\sigma, \omega(\sigma, \sigma))\}$ and $t_1 = \omega(\omega(\sigma, \sigma), \omega(\sigma, \sigma)) \in t \cdot_\sigma^s (t \cdot_\sigma^s t)$ but, on the other hand, $t_1 \notin (t \cdot_\sigma^s t) \cdot_\sigma^s t$.

To illustrate the difference of top-down and bottom-up star, respectively, consider $T = \{\omega(\sigma, \sigma)\}$. We note that $T_\sigma^{s,t,*} = F_\Sigma$ and

$$T_\sigma^{s,b,*} = \{r \in F_\Sigma \mid \text{ each non-leaf node of } r \text{ has at least one leaf as a child } \}.$$

Note that with $T = \{\omega(\sigma, \sigma)\}$, $T_\sigma^{s,b,k}$, $k \geq 0$, consists of trees of height (exactly) k. The trees of $T_\sigma^{s,b,*}$ all consist of a path labeled by binary symbols ω and all children of nodes of the path that "diverge" from the path are labeled by the leaf symbol σ.

The following characterization of bottom-up σ-star as the smallest set closed under concatenation with T from the right follows directly from the definition of bottom-up star. The characterization will be used in the next section.

Lemma 3.1. *For $\sigma \in \Sigma_0$ and $T \subseteq F_\Sigma$, define $\mathrm{cl}_\sigma(T)$ as the smallest set $S \subseteq F_\Sigma$ such that (i) $T \cup \{\sigma\} \subseteq S$, and (ii) $t_1 \cdot^s_\sigma t_2 \in S$ for every $t_2 \in T$ and $t_1 \in S$. Then $\mathrm{cl}_\sigma(T) = T^{s,b,*}$.*

Completely analogously we can define, for $T \subseteq F_\Sigma$, the parallel σ-star of T, denoted $T^{p,*}_\sigma$. Since parallel concatenation is associative, we do not need to distinguish the bottom-up and top-down variants. However, we note that with $T = \{\omega(\sigma, \sigma)\}$, $T^{p,*}_\sigma$ consists of all balanced trees over the ranked alphabet Σ, where $\Sigma_2 = \{\omega\}$, $\Sigma_0 = \{\sigma\}$. Since the "straightforward" definition of Kleene-star based on parallel concatenation does not preserve regularity, in fact, [9] defines a regularity preserving σ-iteration operation by defining the kth ($k \geq 1$) power of T by parallel-concatenating the union of all the ith powers of T, $0 \leq i \leq k-1$, with the tree language T.

It is easy to verify that the definition of the σ-iteration operation (based on parallel concatenation) given in section 7 of [9] coincides with the sequential top-down star defined above, and in the following we will focus only on the sequential variants of iterated concatenation. The top-down (respectively, bottom-up) σ-powers and σ-star of a tree language T are in the following denoted $T^{t,k}_\sigma$, ($k \geq 0$), and $T^{t,*}_\sigma$ (respectively, $T^{b,k}_\sigma$ and $T^{b,*}_\sigma$), that is, we drop the superscript "s" in the notation.

4 State Complexity of Bottom-Up Star

We establish for the bottom-up star operation a tight state complexity bound that is of a different order of magnitude than the state complexity of Kleene-star for string languages. First we give an upper bound for the state complexity of bottom-up star.

Lemma 4.1. *Suppose that tree language L is recognized by a DTA with n states. For $\sigma \in \Sigma_0$, the tree language $L^{b,*}_\sigma$ can be recognized by a DTA with $(n + \frac{3}{2})2^{n-1}$ states.*

Proof. Let $A = (\Sigma, Q, Q_F, g_A)$ be a DTA with n states recognizing the tree language L. Without loss of generality we assume that σ_{g_A} is defined, because otherwise

$$L(A)^{b,*}_\sigma = L(A)^{b,0}_\sigma \cup L(A)^{b,1}_\sigma = \{\sigma\} \cup L(A),$$

and it is easy to construct a DTA with $n+1$ states that recognizes $L(A) \cup \{\sigma\}$.

Choose three disjoint subsets of $2^Q \times (Q \cup \{\mathrm{dead}\})$ by setting

(i) $P_1 = \{(S, q) \mid S \in 2^Q, \{q, \sigma_{g_A}\} \subseteq S, q \in Q_F\}$,
(ii) $P_2 = \{(S, q) \mid S \in 2^Q, q \in S \cap (Q - Q_F)\}$,
(iii) $P_3 = \{(S, \mathrm{dead}) \mid S \in 2^Q, S \neq \emptyset\}$.

Here dead is a new element not in Q. Now define a DTA $B = (\Sigma, P, P_F, g_B)$ where

$$P = P_1 \cup P_2 \cup P_3 \cup \{p_{\mathrm{new}}\}, \quad P_F = \{(S, q) \in P \mid S \cap Q_F \neq \emptyset\} \cup \{p_{\mathrm{new}}\}.$$

We define the transitions of B by setting, $\sigma_{g_B} = p_{\text{new}}$, and for $\tau \in \Sigma_0 - \{\sigma\}$,

$$\tau_{g_B} = \begin{cases} (\{\tau_{g_A}, \sigma_{g_A}\}, \tau_{g_A}) & \text{if } \tau_{g_A} \in Q_F, \\ (\{\tau_{g_A}\}, \tau_{g_A}) & \text{if } \tau_{g_A} \in Q - Q_F, \\ \text{undefined}, & \text{if } \tau_{g_A} \text{ is undefined.} \end{cases} \tag{2}$$

To define transitions on Σ_m, $m \geq 1$, we view p_{new} as the state $(\{\sigma_{g_A}\}, \sigma_{g_A})$, and hence every state of B is represented in the form (S, q), $S \subseteq Q$, $q \in Q$. (Note that p_{new} is not the same as $(\{\sigma_{g_A}\}, \sigma_{g_A})$, because the former is an accepting state and the latter need not be accepting.) For $\tau \in \Sigma_m$ and $(S_1, q_1), \ldots, (S_m, q_m) \in P$, we first denote

$$X = \bigcup_{i=1}^{m} \{\tau_{g_A}(q_1, \ldots, q_{i-1}, z, q_{i+1}, \ldots, q_m) \mid z \in S_i\}$$

Now we define

$$\tau_{g_B}((S_1, q_1), \ldots, (S_m, q_m)) \tag{3}$$

to be equal to

(i) $(X \cup \{\sigma_{g_A}\}, \tau_{g_A}(q_1, \ldots, q_m))$ if $\tau_{g_A}(q_1, \ldots, q_m) \in Q_F$,
(ii) $(X, \tau_{g_A}(q_1, \ldots, q_m))$ if $\tau_{g_A}(q_1, \ldots, q_m) \in Q - Q_F$,
(iii) (X, dead) if $X \neq \emptyset$ and $\tau_{g_A}(q_1, \ldots, q_m)$ is undefined.

In the remaining case, where $X = \emptyset$ and $\tau_{g_A}(q_1, \ldots, q_m)$ is undefined, also (3) is undefined. Note that if for some $1 \leq i \leq m$, $q_i = \text{dead}$, this implies automatically that $\tau_{g_A}(q_1, \ldots, q_m)$ is undefined.

Recall that if (S, q), $S \subseteq Q$, $q \in Q$ is a state of B then $q \in S$ and, furthermore, if $q \in Q_F$ then $\sigma_{g_A} \in S$. The transitions of g_B preserve this property and the state in (i) (in (ii), (iii), respectively) is an element of P_1 (an element of P_2, P_3, respectively).

The second component of the state of B simply simulates the computation of A on the current subtree, and goes to the state dead if the next state of A is undefined. Intuitively, the first component of the state of B consists of all states that A could reach at the current subtree t' assuming that

$$\text{in } t' \text{ at most one subtree of } L(A)^{b,k},\ k \geq 0, \text{ has been replaced by a leaf } \sigma. \tag{4}$$

Inductively, assume that B assigns to the root of tree t_i a state $(S_i, (t_i)_{g_A})$ where $S_i \subseteq Q$ satisfies the property (4) for t_i, $i = 1, \ldots, m$. Now the rule (3) assigns to the root of tree $t = \tau(t_1, \ldots, t_m)$ a state (S, q) where $q = \tau_{g_A}((t_1)_{g_A}, \ldots, (t_m)_{g_A})$ and S consists of all states that A could reach at the root of t assuming the computation uses as arguments $q_1, \ldots, q_m$ where at most one of the q_i's can be replaced by an arbitrary state from S_i, $1 \leq i \leq m$. This means that the state (S, q) again satisfies the property (4) for the tree t.

The choice of the set of final states P_F and Lemma 3.1 now imply that $L(B) = L(A)^{b,*}_{\sigma}$.

It remains to estimate the worst-case size of B. We note that if $Q_F = \{\sigma_{g_A}\}$, in B only states of the form $(\{q\}, q)$, $q \in Q$, can be reachable, and p_{new} can be

identified with $(\{\sigma_{g_A}\}, \sigma_{g_A})$. In this case $L(A)^{b,*}_{\sigma}$ has a DTA with n states. Thus, without loss of generality we assume that Q_F contains a final state distinct from σ_{g_A}.

We note that $|P_1| = |Q_F| \cdot 2^{n-2}$, $|P_2| = |Q - Q_F| \cdot 2^{n-1}$ and $|P_3| = 2^n - 1$. Here the estimation of the size of P_1 relies on the above observation that we can exclude the possibility $Q_F = \{\sigma_{g_A}\}$. Thus, the cardinality of $P_1 \cup P_2 \cup P_3 \cup \{p_{\text{new}}\}$ is maximized as $(n + \frac{3}{2})2^{n-1}$ when $|Q_F| = 1$. ∎

The upper bound of Lemma 4.1 is of a different order of magnitude than the known state complexity of Kleene-star for string languages [25]. It remains to verify that the bound of Lemma 4.1 can be reached in the worst case.

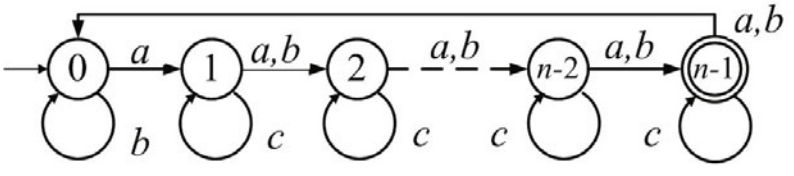


Fig. 2. The DFA A from [25] with added c-transitions

Figure 2 represents a DFA A used in [25,26] for the lower bound construction for Kleene-star where we have added transitions on the symbol c. Note that A is an incomplete DFA since the c transition on 0 is undefined. Based on A we define in the following a tree automaton M_A.

Choose $\Sigma = \Sigma_0 \cup \Sigma_1 \cup \Sigma_2$ where $\Sigma_0 = \{e\}$, $\Sigma_1 = \{a, b, c\}$ and $\Sigma_2 = \{a_2, d_2\}$. We define a DTA $M_A = (\Sigma, Q_A, Q_{A,F}, g_A)$, where $Q_A = \{0, 1, \ldots, n{-}1\}$, $Q_{A,F} = \{n-1\}$ and the transition function g_A is defined by setting:

(i) $e_{g_A} = 0$, $c_{g_A}(i) = i$, $1 \le i \le n-1$,
(ii) $a_{g_A}(i) = (a_2)_{g_A}(i, i) = i + 1$, $0 \le i \le n-2$,
$a_{g_A}(n-1) = (a_2)_{g_A}(n-1, n-1) = 0$,
(iii) $b_{g_A}(i) = i + 1$, $1 \le i \le n-2$, $b_{g_A}(j) = 0$, $j \in \{0, n-1\}$,
(iv) $(d_2)_{g_A}(0, i) = i$, $i = 0, 2, 3, \ldots, n-1$, $(d_2)_{g_A}(1, 1) = 1$.

All transitions of g_A not listed above are undefined. Intuitively, the construction of M_A can be, roughly speaking, explained as follows. Denote by T_d the subset of F_Σ consisting of trees without any occurrences of the binary symbol d_2, thus the only binary symbol in trees of T_d is a_2. On a tree $t \in T_d$, the DTA M_A simulates the computation of A on each string of symbols starting from a node of height one, where occurrences of a_2 are "interpreted" simply as a. The computations on different paths verify that for any $u \in \text{dom}(t)$ labeled by a_2 and any nodes v_1 and v_2 of height one below u, the simulated computations started from v_1 and v_2 agree at u. Furthermore, if $u = \varepsilon$, the simulated computation has to accept.

Note that the original DFA has no transitions on d, and the transitions on d_2 have been added for a technical reason that will be used in the proof of

Lemma 4.3. Also, the above intuitive description is not completely precise on how M_A operates on binary symbols a_2 where one child is a leaf (that gets assigned the state 0) and the other child is not a leaf. The following Lemmas 4.2 and 4.3 rely only on the formal definition of the transition function g_A of M_A. The above intuitive description of the operation of M_A is intended only as a guide that may be useful in understanding the operation of the DTA constructed to recognize the bottom-up e-star of $L(M_A)$. Finally, note that the d_2-transitions will be needed only to establish the reachability of one particular state, and in most of the technical constructions the above intuitive description of the operation of M_A (based on the DFA A of Figure 2) can be sufficient.

Using the construction of the proof of Lemma 4.1, based on M_A we construct a DTA $M_B = (\Sigma, Q_B, Q_{B,F}, g_B)$ that recognizes the tree language $L(M_A)_e^{b,*}$. We make the convention that the sink-state "dead" used in the proof is denoted by n. Thus the set of states Q_B consists of the special state p_{new} assigned to e and all pairs

$$(P, q),\ P \subseteq \{0, \ldots, n-1\},\ 0 \leq q \leq n, \tag{5}$$

where $0 \leq q \leq n-1$ implies $q \in P$, $q = n-1$ implies $0 \in P$ and $q = n$ implies $P \neq \emptyset$. The number of pairs as in (5) is $(n + \frac{3}{2})2^{n-1} - 1$.

In the following two lemmas we establish that M_B is a minimal DTA. That is, first we show that all states of Q_B are pairwise inequivalent with respect to the Myhill-Nerode equivalence relation extended to trees. Second we show that all states of Q_B are reachable, that is, for each $q \in Q_B$ there exists $t \in F_\Sigma$ such that $t_{g_B} = q$. The proof of our first lemma assumes that all states are reachable which will be established next in Lemma 4.3[1].

Lemma 4.2. *All states of M_B are pairwise inequivalent.*

Proof. For the sake of convenience, we assume that we have already proven that all states of M_B are reachable (Lemma 4.3). Thus, in order to distinguish two states with respect to the Myhill-Nerode relation, we can use an arbitrary configuration of M_B where one leaf is replaced by the given states. More formally, in order to show that two distinct states of Q_B, p_1 and p_2, are inequivalent, it is sufficient to find $t \in F_{\Sigma^{M_B}}[x]$ such that the computation of M_B started from the configuration $t(x \leftarrow p_1)$ accepts if and only if the computation started from the configuration $t(x \leftarrow p_2)$ does not accept.

We first show that any two distinct states (S_1, q_1) and (S_2, q_2) as in (5) are not equivalent. After that we consider the special state p_{new}. We begin by considering the case where neither of q_1 or q_2 is equal to n (which was used to denote the dead state of M_A).

Case $0 \leq q_1, q_2 \leq n-1$: (a) Assume $S_1 \neq S_2$ and $s \in S_1 - S_2$ (The other possibility is completely symmetric.) After reading $n - s - 1$ unary symbols a, a final state is reached from state (S_1, q_1). On the other hand, since (S_2, q_2) is as in (5), $q_2 \neq s$. This means that the computation C that begins with

[1] The proof of Lemma 4.3 does not rely on Lemma 4.2.

(S_2, q_2) and reads $n - s - 1$ unary symbols a ends with a non-final state. Note that at some point during the computation C, the second component may become $n-1$ which adds an element 0 to the first component. However, at the end of the computation C the first component cannot contain $n - 1$.

(b)(i) Next we consider the case $S_1 = S_2 = S$, $\{0, 1, \ldots, n-2\} \not\subseteq S$ and $q_1 \neq q_2$. According to the definition of the states (5), $q_1, q_2 \in S$. Choose $p \in \{0, 1, \ldots, n-2\} - S$ and consider a tree $t_1 = a^{2n-2-q_1} a_2((\{q_1, p\}, p), x) \in F_{\Sigma^{M_B}}[x]$. Since $p \in \{0, 1, \ldots, n-2\}$, $(\{q_1, p\}, p)$ is a legal state (5). Consider the computation of M_B on tree $t_1(x \leftarrow (S, q_1))$. Since $p \notin S$ the state $(\{q_1 + 1\}, n)$ is assigned to the root of the subtree $a_2((\{q_1, p\}, q_1), (S, q_1))$. (Here addition is modulo n.) After this the computation reads the $2n-2-q_1$ unary symbols a in t_1 and ends in an accepting state. On the other hand, consider the computation of M_B on $t_1(x \leftarrow (S, q_2))$. Since $p \notin S$ and $q_2 \notin \{q_1, p\}$, the transition $(a_2)_{g_B}$ on arguments $(\{q_1, p\}, p)$, $(S, q_2))$ is undefined and the computation does not accept.

(b)(ii) Consider $S = \{0, 1, \ldots, n-2\}$, and hence we know that $q_1, q_2 \neq n-1$. From state (S, q_i) by reading a unary symbol b we get (S', q_i'), where $S' = \{0, 2, \ldots, n-2, n-1\}$. Since $q_1, q_2 \neq n-1$, $q_1' \neq q_2'$ and the states (S', q_1') and (S', q_2') are distinguished as in b(i) above.

(b)(iii) Consider then the possibility $S = \{0, 1, \ldots, n-1\}$ and $q_1 \neq q_2$. If $\{q_1, q_2\} \neq \{0, n-1\}$, by reading a unary symbol b from (S, q_1) and (S, q_2), respectively, we get two states (S', q_1'), (S', q_2'), $q_1' \neq q_2'$, that are distinguished as in the previous case[2]. Next consider the case $\{q_1, q_2\} = \{0, n-1\}$, and first assume that $n \geq 3$. By reading a unary symbol a we obtain states (S, q_1+1), $(S, q_2 + 1)$ where $q_1 + 1 \neq q_2 + 1$ and $q_i + 1 \neq n - 1$, $i = 1, 2$ (addition is modulo n). The states $(S, q_1 + 1)$ and $(S, q_2 + 1)$ can be distinguished as in the previous cases.

Finally consider the possibility $n = 2$ and $\{q_1, q_2\} = \{0, 1\}$. From state $(\{0, 1\}, 1)$ by reading unary symbols ca, we reach the accepting state $(\{0, 1\}, 0)$. On the other hand, a computation starting from $(\{0, 1\}, 0)$ by reading the unary symbols ca reaches the nonaccepting state $(\{0\}, 2)$.

Case where $q_2 = n$: First assume $q_1 \neq n$. Choose $t_2 \in F_{\Sigma^{M_B}}[x]$ by setting $t_2 = a^{n-2} a_2((\{0, 1\}, 1), b^{n-1}(x))$. Since $n - 1$ consecutive b-transitions take any state of A to state 0, the computation of M_B on $t_2(x \leftarrow (S_1, q_1))$ assigns state $(\{0\}, 0)$ to the root of the subtree $b^{n-1}((S_1, q_1))$. Then the state $(\{1\}, n)$ is reached at the root of the subtree $a_2((\{0, 1\}, 1), b^{n-1}((S_1, q_1)))$. A final state $(\{n-1\}, n)$ is reached after reading further $n - 2$ unary symbols a. On the other hand, in the computation of M_B on $t_2(x \leftarrow (S_2, n))$ the state $(\{0\}, n)$ is assigned to the root of the subtree $b^{n-1}((S_2, n))$. When reading the binary symbol a_2 with arguments $(\{0, 1\}, 1)$ and $(\{0\}, n)$ the computation step of M_B is undefined, and hence M_B does not accept $t_2(x \leftarrow (S_2, n))$.

Finally consider the case where also $q_1 = n$. Thus $S_1 \neq S_2$ and choose $s \in S_1 - S_2$. After reading $n - s - 1$ unary symbols a, a final state is reached

[2] The b-transitions of A violate injectivity only on states 0 and $n - 1$.

from state (S_1, n), and the same computation does not reach a final state from (S_2, n).

It remains to show that p_{new} is not equivalent with any state (S, q) as in (5). Since p_{new} is final, it is sufficient to consider states where $n-1 \in S$. Thus, by reading a unary symbol c from state (S, q) we get a state (S', q'), where $n-1 \in S'$ and $0 \leq q' \leq n$. On the other hand, computations starting from p_{new} are identical to computations starting from $(\{0\}, 0)$ and hence a computation step with unary symbol c is undefined. ■

Before the next lemma we introduce the following notation. For a unary tree representing a configuration of M_B, $t = z_1(z_2(\ldots z_m(z_0)\ldots)) \in F_{\Sigma^{M_B}}$, we define $\text{word}(t) = z_m z_{m-1} \ldots z_1$. Note that $\text{word}(t)$ consists of the sequence of symbols labeling the nodes of t bottom-up, and the label of the leaf is not included. In the following when we refer to $\text{word}(t)$ of a tree t, without further mention, this implies that t is a unary tree.

Lemma 4.3. *All states of M_B are reachable.*

Proof. The transition function of M_B assigns the special state p_{new} to leaf symbol e. Recall that from p_{new} the computation of M_B continues as from $(\{0\}, 0)$. Thus, after reading $n-1$ unary symbols a we reach the state $(\{0, n-1\}, n-1)$.

Inductively, we assume that a state $(\{0, 1, 2, \ldots, k, n-1\}, n-1)$, $0 \leq k < n-2$, is reachable. We show that $(\{0, 1, 2, \ldots, k+1, n-1\}, n-1)$ is also reachable. From state $(\{0, 1, 2, \ldots, k, n-1\}, n-1)$, we reach the state $Z_1 = (\{1, 2, \ldots, k+1, 0\}, 0)$ by reading a unary symbol a. By our assumption on k, $k+1 < n-1$. Thus from Z_1 we reach the state $Z_2 = (\{2, 3, \ldots, k+2, 0\}, 0)$ by reading b. Since $k < n-2$, all elements of $\{2, 3, \ldots, k+2, 0\}$ are distinct (that is, the b-transition does not take $k+1$ to 0). After reading $n-1$ symbols a, the state $(\{1, 2, \ldots, k+1, n-1, 0\}, n-1)$ is reached. The element 0 is added to the first component as the second component becomes $n-1$.

By the above inductive claim we now know that the state $(\{0, 1, \ldots, n-2, n-1\}, n-1)$ is reachable. After reading $i+1$ $a's$, state $(\{0, 1, \ldots, n-2, n-1\}, i)$ is reached, $0 \leq i \leq n-1$.

Inductively, assume that all states (S, j), where $|S| \geq k+1$, $1 \leq k < n$ and $0 \leq j \leq n-1$ as in (5) are reachable. We show that then also states where $|S| = k$ are reachable. Let (S, s_i) where $S = \{s_1, s_2, \ldots, s_k\}$, $1 \leq i \leq k$ and $0 \leq s_1 < s_2 < \ldots < s_k \leq n-1$ be an arbitrary state where $|S| = k$. Recall that in states of M_B, when the second component is not n, it must belong to the first component.

In the below cases (a) and (b), numbers $z \geq n$ are interpreted as the unique element of $\{0, 1, \ldots, n-1\}$ congruent to z modulo n.

(a-i) First consider the case where $s_i < n-1$. The following discussion assumes $n \geq 3$, and the case $n = 2$ is handled in case (a-ii). Since $|S| = k < n$, in the "cyclical sequence" of $s_1, \ldots, s_k$, there exist two consecutive numbers with difference at least two, where the difference between the numbers s_k and

s_1 is counted modulo n. More formally, either there exists $1 \leq j \leq k-1$ such that $s_{j+1} - s_j \geq 2$ or $n + s_1 - s_k \geq 2$. In the latter case we choose $j = k$. In the following we assume that $i \leq j$. The case where $i > j$ is similar and only some notations are changed. According to the inductive assumption, the state $Z_3 = (\{0, n-1\} \cup S_1, n + s_i - s_j - 1)$ where $S_1 = \{s_{j+1} - s_j - 1, s_{j+2} - s_j - 1, \ldots, s_k - s_j - 1, n + s_1 - s_j - 1, n + s_2 - s_j - 1, \ldots, n + s_{j-1} - s_j - 1\}$ is reachable. Note that since $0 \leq s_1 < s_2 < \ldots < s_k \leq n-1$ and $s_{j+1} - s_j \geq 2$, $|S_1 \cup \{0, n-1\}| = k+1$. After reading from state Z_3 a unary symbol b, we get the state $Z_4 = (\{0\} \cup S_2, n + s_i - s_j)$ where $S_2 = \{s_{j+1} - s_j, s_{j+2} - s_j, \ldots, s_k - s_j, n + s_1 - s_j, n + s_2 - s_j, \ldots, n + s_{j-1} - s_j\}$. Since $0 \leq s_1 < s_2 < \ldots < s_k \leq n-1$, $0 \notin S_2$. From state Z_4 we reach the state $(\{s_j, s_{j+1}, s_{j+2}, \ldots, s_k, n + s_1, n + s_2, \ldots, n + s_{j-1}\}, n + s_i)$ by reading s_j symbols a. The latter state is the state (S, s_i) that we wanted.

(a-ii) Assume that $s_i < n-1$ and $n = 2$. Now $k = 1$, and the only legal state (S, s_i), $|S| = k = 1$, $0 \leq s_i < 1$, is $(\{0\}, 0)$ (because we know that $s_i \in S$). The state $(\{0\}, 0)$ is reached from state p_{new} by reading unary symbols ab.

(b) Now consider the case where $s_i = n-1$, and thus $i = k$. This implies that $0 \in S$, and we have $s_i (= s_k) = n-1$ and $s_1 = 0$. Since $k < n$, there exists $1 \leq j \leq k-1$ such that $s_{j+1} - s_j \geq 2$. According to the inductive assumption, the state $Z_5 = (\{0, n-1\} \cup S_3, n - 2 - s_j)$ is reachable, where $S_3 = \{s_{j+1} - s_j - 1, s_{j+2} - s_j - 1, \ldots, s_{k-1} - s_j - 1, n-1-s_j-1, n+0-s_j-1, n + s_2 - s_j - 1, \ldots, n + s_{j-1} - s_j - 1\}$. Similarly as in (a) above we observe that $|S_3 \cup \{0, n-1\}| = k+1$. From state Z_5 we get the state $Z_6 = (\{s_{j+1} - s_j, s_{j+2} - s_j, \ldots, s_{k-1} - s_j, n-1-s_j, n+0-s_j, n+s_2-s_j, \ldots, n+s_{j-1}-s_j, 0\}, n-1-s_j)$ by reading a symbol b. After reading s_j symbols a, from state Z_6 we reach the state $(\{s_{j+1}, s_{j+2}, \ldots, s_{k-1}, n-1, n+0, n+s_2, \ldots, n+s_{j-1}, s_j\}, n-1)$. This means that we have reached the desired state $(S, n-1)$ with $S = \{0, s_2, \ldots, s_{k-1}, n-1\}$.

Up to now, we have shown that all that states (S, j), $S \subseteq \{0, \ldots, n-1\}$, $0 \leq j \leq n-1$ as in (5) are reachable. Next we will show that the states (S, n), $S \subset \{0, 1, \ldots, n-1\}$ are reachable.

We know that $(\{0, 1, \ldots, n-1\}, 0)$ is reachable and from this state we get $Z_7 = (\{1, \ldots, n-1\}, n)$ by reading a unary symbol c. From Z_7 we get all states (S, n), $|S| = n-1$ by cycling the elements of S using a-transitions. Now inductively, assume that all states (S, n), $n > |S| \geq k+1$, $k < n-1$ are reachable. Consider an arbitrary state (S, n) where $|S| = k$. Choose $0 \leq j \leq n-1$ such that $j \notin S$. By our inductive assumption the state $(S \cup \{j\}, n)$ is reachable. From this state we reach (S, n) by reading the sequence of unary symbols $a^{n-j}ca^j$. Note that transitions on a always add one modulo n to states of S and the c-transition deletes the element 0 and is the identity on all other elements.

It remains to consider the state $(\{0, 1, \ldots, n-1\}, n)$. We know that states $(\{0, 1\}, 0)$ and $(\{0, 1, \ldots, n-1\}, 1)$ are reachable. According to the definition of d_2-transitions of M_A, the d_2-transition of M_B with arguments $(\{0, 1\}, 0)$ and $(\{0, 1, \ldots, n-1\}, 1)$ gives the state $(\{0, 1, \ldots, n-1\}, n)$. ■

Note that above the transitions on d_2 were needed only to establish that the state $(\{0,1,\ldots,n-1\},n)$ is reachable in M_B. The transitions of d_2 in M_A did not have a similar intuitive interpretation as the other transitions based on the DFA A, and they were introduced only for the technical purpose needed at the end of the proof of Lemma 4.3.

By Lemmas 4.1, 4.2 and 4.3 we have a tight bound for the state complexity of bottom-up star that differs by an order of magnitude from the known bound for Kleene-star of string languages [25].

Theorem 4.1. *If A is a DTA with n states, the bottom-up star of $L(A)$ can be recognized by a DTA with $(n+\frac{3}{2})\cdot 2^{n-1}$ states. For every $n \geq 2$, there exists an n-state DTA A and $\sigma \in \Sigma_0$ such that the minimal DTA for $L(A)^{b,*}_{\sigma}$ has $(n+\frac{3}{2})\cdot 2^{n-1}$ states.*

5 State Complexity of Top-Down Star

Here we give a tight state complexity bound for top-down star of regular tree languages. The top-down iteration of the concatenation operation allows the replacement of subtrees at arbitrary locations and, as can perhaps be expected, the state complexity is similar as for the Kleene-star of string languages. For completeness, we give a brief construction for the upper bound, because we are considering incomplete automata and the known state complexity bounds for ordinary DFAs are stated in terms of complete DFAs [25,26]. The state complexity results for complete and incomplete DFAs, respectively, differ slightly for operations such as union [25,7] or concatenation [25,16].

Theorem 5.1. *Let A be a DTA with n states and $\sigma \in \Sigma_0$. The top-down σ-star of the tree language recognized by A, $L(A)^{t,*}_{\sigma}$, can be recognized by a DTA with $\frac{3}{4}\cdot 2^n$ states and this bound can be reached in the worst case.*

Proof. Denote $A=(\Sigma,Q_A,Q_{A,F},g_A)$ and let $q_{\rm new}$ be a new element not in Q_A. We can assume that σ_{g_A} is defined because otherwise $L(A)^{t,*}_{\sigma}=L(A)\cup\{\sigma\}$.

We define $B=(\Sigma,Q_B,Q_{B,F},g_B)$, where

$$Q_B=\{q_{\rm new}\}\cup\{\emptyset\neq P\subseteq Q_A \mid P\cap Q_{A,F}\neq\emptyset \text{ implies } \sigma_{g_A}\in P\},$$

$$Q_{B,F}=\{q_{\rm new}\}\cup\{P\in Q_B \mid P\cap Q_{A,F}\neq\emptyset\}.$$

The transitions of B are defined for $\tau\in\Sigma_0-\{\sigma\}$ by setting

$$\tau_{g_B}=\begin{cases}\{\tau_{g_A},\sigma_{g_A}\} & \text{if } \tau_{g_A}\in Q_{A,F},\\ \{\tau_{g_A}\} & \text{if } \tau_{g_A}\in Q_A-Q_{A,F},\\ \text{undefined} & \text{if } \tau_{g_A} \text{ is undefined.}\end{cases}$$

For the leaf symbol σ used to define the star-operation, we set $\sigma_{g_B}=q_{\rm new}$. For $m\geq 1$, $\tau\in\Sigma_m$ and $X_1,\ldots,X_m\in Q_B$ we define $\tau_{g_B}(X_1,\ldots,X_m)=Y\cup Z$, where

$$Y=\{\tau_{g_A}(x_1,\ldots,x_m) \mid x_i\in X_i \text{ if } X_i\in 2^{Q_A},\ x_i=\sigma_{g_A} \text{ if } X_i=q_{\rm new}, 1\leq i\leq m\},$$

and $Z=\{\sigma_{g_A}\}$ if $Y\cap Q_{A,F}\neq\emptyset$, $Z=\emptyset$ otherwise.

The construction of B is similar as the construction used to recognize the Kleene-star of a string language. Note that the state q_{new} is used as a copy of σ_{g_A} because the latter state is not, in general, accepting. We leave to the reader the details of verifying that B recognizes $L(A)^{t,*}_{\sigma}$.

To get the upper bound on the number of states, we note that if $Q_{A,F} = \{\sigma_{g_A}\}$, then we can identify q_{new} and σ_{g_A} and in the resulting DTA the number of reachable states is (at most) n. Thus we can assume that $Q_{A,F} - \{\sigma_{g_A}\} \neq \emptyset$. In the case where $\sigma_{g_A} \notin Q_{A,F}$, we observe that $(2^{|Q_{A,F}|} - 1) \cdot 2^{n-|Q_{A,F}|-1}$ of the elements $P \in 2^{Q_A} - \{\emptyset\}$ contain an element of $Q_{A,F}$ and, at the same time, do not contain σ_{g_A}. Thus, by choosing $|Q_{A,F}| = 1$, the cardinality of Q_B is maximized as $|Q_B| = 2^n - 1 - (2-1) \cdot 2^{n-2} + 1 = \frac{3}{4} \cdot 2^n$. It is easy to verify that this bound cannot be exceeded with $\sigma_{g_A} \in Q_{A,F}$, $|Q_{A,F}| \geq 2$.

When restricted to unary trees, the top-down (or bottom) star operation coincides with Kleene-star on string languages. Theorem 5.5 of [25] gives a complete DFA C with n states such that the state complexity of the Kleene-star of $L(C)$ is $\frac{3}{4} \cdot 2^n$. Furthermore, C does not have a dead state, which means that the same lower bound construction works for incomplete DFAs. ■

6 Conclusion

The lower bound construction for Kleene-star in [25] uses a two-letter alphabet, and hence the worst-case state complexity of top-down star can be achieved over a ranked alphabet with two unary and one nullary symbol. It is clear that one unary and one nullary symbol is not sufficient, because it is known that the state complexity of Kleene-star for string languages over a one-letter alphabet is $(n-1)^2 + 1$ [25]. With one binary symbol ω and one nullary symbol σ, we can encode strings over a two letter alphabet as trees "built up" from elements $\omega(\sigma, x)$ and $\omega(x, \sigma)$. In this way one clearly gets an exponential lower bound construction, however, we do not know whether one binary and one nullary symbol is sufficient to reach the precise bound of Theorem 5.1.

Our lower bound construction for Theorem 4.1 uses a ranked alphabet of six symbols. The state complexity for bottom-up star is of different order of magnitude than the corresponding bound for string languages. This means that the worst-case constructions essentially need to rely on "tree properties" and finding the minimal alphabet size remains an open question.

References

1. Calude, C.S., Salomaa, K., Roblot, T.: Finite state complexity. Theoret. Comput. Sci. 412, 5668–5677 (2011)
2. Calude, C.S., Salomaa, K., Roblot, T.: State-size hierarchy for finite-state complexity. Internat. J. Foundations Comput. Sci. (accepted, 2011)
3. Comon, H., Dauchet, M., Gilleron, R., Jacquemard, F., Lugiez, D., Löding, C., Tison, S., Tommasi, M.: Tree Automata Techniques and Applications (2007), electronic book available at `tata.gforge.inria.fr`

4. Cristau, J., Löding, C., Thomas, W.: Deterministic Automata on Unranked Trees. In: Liśkiewicz, M., Reischuk, R. (eds.) FCT 2005. LNCS, vol. 3623, pp. 68–79. Springer, Heidelberg (2005)
5. Domaratzki, M., Okhotin, A.: State complexity of power. Theoret. Comput. Sci. 410, 2377–2392 (2009)
6. Ellul, K., Krawetz, B., Shallit, J., Wang, M.-W.: Regular expressions: New results and open problems. J. Automata, Lang. and Combinatorics 10, 407–437 (2005)
7. Gao, Y., Salomaa, K., Yu, S.: Transition complexity of incomplete DFAs. Fundamenta Informaticae 110, 143–158 (2011)
8. Gao, Y., Yu, S.: State Complexity of Four Combined Operations Composed of Union, Intersection, Star and Reversal. In: Holzer, M., Kutrib, M., Pighizzini, G. (eds.) DCFS 2011. LNCS, vol. 6808, pp. 158–171. Springer, Heidelberg (2011)
9. Gécseg, F., Steinby, M.: Tree languages. In: [20], vol. III, pp. 1–68
10. Gruber, H., Holzer, M.: Tight Bounds on the Descriptional Complexity of Regular Expressions. In: Diekert, V., Nowotka, D. (eds.) DLT 2009. LNCS, vol. 5583, pp. 276–287. Springer, Heidelberg (2009)
11. Holzer, M., Kutrib, M.: Descriptional and computational complexity of finite automata — A survey. Inf. Comput. 209, 456–470 (2011)
12. Holzer, M., Salomaa, K., Yu, S.: On the state complexity of k-entry deterministic finite automata. J. Automata, Languages and Combinatorics 6, 453–466 (2001)
13. Jirásková, G., Okhotin, A.: On the State Complexity of Operations on Two-Way Finite Automata. In: Ito, M., Toyama, M. (eds.) DLT 2008. LNCS, vol. 5257, pp. 443–454. Springer, Heidelberg (2008)
14. Kapoutsis, C.A.: Nondeterminism Is Essential in Small 2FAs with Few Reversals. In: Aceto, L., Henzinger, M., Sgall, J. (eds.) ICALP 2011, Part II. LNCS, vol. 6756, pp. 198–209. Springer, Heidelberg (2011)
15. Martens, W., Niehren, J.: On the minimization of XML schemas and tree automata for unranked trees. J. Comput. System Sci. 73, 550–583 (2007)
16. Piao, X., Salomaa, K.: Operational state complexity of deterministic unranked tree automata. In: Proc. of DCFS 2010, pp. 181–192 (2010)
17. Piao, X., Salomaa, K.: Transformations between different models of unranked bottom-up tree automata. Fund. Informaticae 109, 405–424 (2011)
18. Piao, X., Salomaa, K.: State complexity of star and quotient operation for unranked tree automata, School of Computing, Queen's University Technical Report No. 2011-577 (19 pp.) (2011)
19. Piao, X., Salomaa, K.: State Trade-Offs in Unranked Tree Automata. In: Holzer, M., Kutrib, M., Pighizzini, G. (eds.) DCFS 2011. LNCS, vol. 6808, pp. 261–274. Springer, Heidelberg (2011)
20. Rozenberg, G., Salomaa, A. (eds.): Handbook of Formal Languages, vol. I–III. Springer, Heidelberg (1997)
21. Salomaa, A., Salomaa, K., Yu, S.: State complexity of combined operations. Theoret. Comput. Sci. 383, 140–152 (2007)
22. Salomaa, A., Salomaa, K., Yu, S.: Undecidability of the State Complexity of Composed Regular Operations. In: Dediu, A.-H., Inenaga, S., Martín-Vide, C. (eds.) LATA 2011. LNCS, vol. 6638, pp. 489–498. Springer, Heidelberg (2011)
23. Schwentick, T.: Automata for XML, — a survey. J. Comput. System Sci. 73, 289–315 (2007)
24. Shallit, J.: A Second Course in Formal Languages and Automata Theory. Cambridge University Press (2009)
25. Yu, S.: Regular languages. In: [20], vol. I, pp. 41–110
26. Yu, S., Zhuang, Q., Salomaa, K.: The state complexity of some basic operations on regular languages. Theoret. Comput. Sci. 125, 315–328 (1994)

Composition Sequences and Synchronizing Automata

Arto Salomaa

Turku Centre for Computer Science
Joukahaisenkatu 3–5 B, 20520 Turku, Finland
asalomaa@utu.fi

Abstract. This paper investigates functional completeness and synchronization of finite automata, within the framework of composition of functions over a finite domain. Results about decidability and complexity are obtained, as well as partial criteria for synchronizability and simplified proofs of earlier results.

1 Introduction and Basic Notions

Consider functions $g(x)$ whose domain is a fixed finite set N with n elements, $n \geq 2$, and whose range is included in N. We will mostly deal with the case where this abstract setup is applied to finite deterministic automata. We make the following *convention*, valid throughout this paper: n always stands for the number of elements in the basic set N. We can visualize N simply as the set consisting of the first n natural numbers: $N = \{1, 2, \ldots, n\}$. Clearly, there are altogether n^n functions in the set N^N we are considering. For convenience, we will refer to the cardinality of the range of a function g as the *genus* of g. Thus, permutations are of genus n, and constants are of genus 1.

The two interpretations mostly considered in this set-up are *many-valued logic* and *finite automata.* In the former, the set N consists of n *truth values* and the functions are truth functions. We will not consider many-valued logic in this paper. The reader is referred to [11] or, for more details, to [13], where questions of composition theory related to the ones studied in this paper are discussed.

In the interpretation dealing with finite automata, the set N consists of the *states* the automaton, whereas each letter of the input alphabet induces a specific function: the next state when reading that letter. The automata considered will be *deterministic* and *complete*: for each state q and input letter a, the transition function δ defines a unique next state $\delta(q, a)$. Our automata do not have specified initial or final states. Thus, our automata will be triples $\mathcal{A} = (Q, \Sigma, \delta)$, where Q is the set of states (consisting of n elements), Σ is the input alphabet, and δ maps the product $Q \times \Sigma$ into Q. The domain of the mapping is extended in the usual way to $Q \times \Sigma^*$, and specific automata will be depicted as labeled graphs. In the early days of automata theory, such automata were often referred to as *Medvedev automata*, after Ju.T. Medvedev. He translated the seminal Princeton book *Automata Studies* from 1956 (see [7]) into Russian, and included his own

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 403–416, 2012.

article in the translation. The latter remained largely unknown in the West. When we speak of automata or finite automata in the sequel, we always mean Medvedev automata.

Coming back to our basic setup of functions, each letter a of the input alphabet defines a function $g_a(x) = \delta(x, a)$. We will often identify g_a with a. In this association, catenation of letters corresponds to *composition* of functions. We read compositions from *left to right:* first a, then b. This is in accordance of reading the input words of a finite automaton from left to right. Because of this convention, it is natural to write the argument x of a function to the left: $(x)ab = ((x)a)b$.

Consider a nonempty set $\mathcal{F}$ of functions, for instance, the set defined by all functions g_a, where a runs through the input letters of a specific automaton. We will consider the set $\mathcal{G}(\mathcal{F})$ of all functions *generated* by $\mathcal{F}$, that is, obtained as compositions (with arbitrarily many composition factors) of functions from $\mathcal{F}$. Assume that a particular function g can be expressed as a composition of functions $a_i, i = 1, 2, \ldots, k$, belonging to $\mathcal{F}$: $g = a_1 a_2 \ldots a_k$, where some of the functions a_i may coincide. Then the word $a_1 a_2 \ldots a_k$ is referred to as a *composition sequence* for g. The number k is referred to as the *length* of the composition sequence. The function g is often referred to as the *target function.* Our composition sequences have to be nonempty, implying that the identity function is not necessarily in $\mathcal{G}(\mathcal{F})$. Clearly, $\mathcal{G}(\mathcal{F})$ can be viewed as the *semigroup* generated by $\mathcal{F}$. The set $\mathcal{F}$ is termed *complete* if $\mathcal{G}(\mathcal{F})$ contains all n^n functions.

Since n is finite, a specific function f can always be defined by a table. Omitting the argument values, this amounts to giving the *value sequence* of f, that is, the sequence $f(1),\ f(2), \ldots,\ f(n)$ of its values for the increasing values of the argument. This will often be done in the sequel. We use the notation $\mathcal{F}_\mathcal{A}$ to indicate that $\mathcal{F}$ equals the set of transition functions g_a of a finite automaton $\mathcal{A}$.

In this paper we will study the semigroup $\mathcal{G}(\mathcal{F}_\mathcal{A}))$, in particular composition sequences and their length. The fundamental notions are defined as follows.

Definition 1. *A finite automaton $\mathcal{A}$ is* functionally complete *if $\mathcal{F}_\mathcal{A}$ is complete. It is* synchronizing or synchronizable *if $\mathcal{G}(\mathcal{F}_\mathcal{A})$ contains a constant function. If $\mathcal{A}$ is synchronizing, then any composition sequence for a constant function is referred to as a* synchronizing word.

Observe that if y is a synchronizing word, so is xyz, for any words x and z.

2 Functional Completeness of Finite Automata

The reader is referred to [12] for a proof of the following result. We exclude the case $n = 2$, for which the two functions with the value sequences 21 and 11 form a complete set.

Theorem 1. *Assume that $n \geq 3$. Then three functions generate all functions if and only if two of them generate the symmetric group S_n and the third one is of genus $n - 1$. No less than three functions generate all functions.*

According to [8], given any nonidentical permutation, another permutation can be effectively constructed such that the two permutations form a basis of the whole symmetric group. The case $n = 4$ is exceptional because a permutation in the Klein Four-Group cannot be extended in this way to a basis. Thus, we obtain the following corollary of Theorem 1.

Theorem 2. *Assume that $n \neq 4$. Given a nonidentical permutation a and a function c of genus $n-1$, a function b can be effectively constructed such that the set $\{a, b, c\}$ is complete.*

By Theorem 1, the automaton defined by Figure 1 is functionally complete. The circular permutation a and the transposition b generate the symmetric group, whereas c defines a function of genus $n-1$. Apart from the trivial case of two-state automata, at least three input letters are needed for functional completeness.

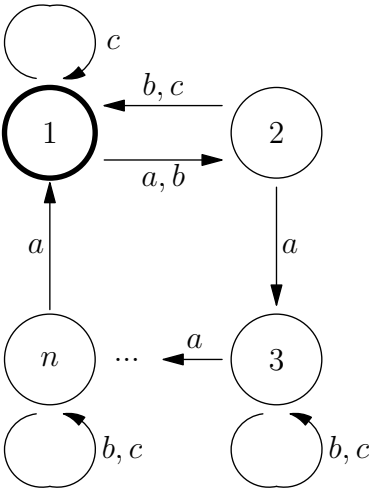


Fig. 1. A functionally complete automaton

As another example, assume that $n = 6$ and that three functions a, b, c are defined by the value sequences 213456, 234561 and 112345, respectively. Thus, a is the transposition (12), b is the circular permutation (123456), whereas c is of genus 5 and maps 1 to itself and all the other numbers to the preceding number. The target function f is defined by the value sequence 311344. Then the composition sequence (perhaps not the shortest possible one)

$$b^4d^2cdb^4db^2d^3bc^2db^4d^2b^3d^2b^2d^4ac^2db^4d^2b^3d^2$$

defines f, where we have abbreviated $d = ab$.

To get an overall picture of composition sequences, we still present an exhaustive classification in the case $n = 3$. This is a good illustration of many of

the basic phenomena. We define the functions in a complete set $\{a, b, c\}$ by the value sequences 231, 132 and 223, respectively. Thus, a is the circular permutation (123), b is the transposition (23), whereas g is of genus 2 and maps 1 to 2 but keeps 2 and 3 fixed.

The following array lists all of the 27 functions, giving in each case the value sequence and a shortest possible composition sequence.

value sequence	composition		value sequence	composition
111	ca^2ca^2	$\bowtie$	112	ca^2
113	cba	$\bowtie$	121	aca^2
122	a^2cab	$\bowtie$	123	b^2
131	$acba$	$\bowtie$	132	b
133	a^2ca	$\bowtie$	211	a^2ca^2
212	$acab$	$\bowtie$	213	ba
221	cab	$\bowtie$	222	ca^2c
223	c	$\bowtie$	231	a
232	ac	$\bowtie$	233	a^2cb
311	$abcba$	$\bowtie$	312	a^2
313	aca	$\bowtie$	321	ab
322	a^2c	$\bowtie$	323	acb
331	ca	$\bowtie$	332	cb
333	ca^2ca	$\bowtie$		

Thus, altogether 10 different functions have composition sequences of length ≤ 2. Additionally, 6 functions have sequences of length 3, and 6 further functions of length 4. The remaining exceptional functions require a longer composition sequence.

One of the constants is represented by a sequence of length 4 but no constant by a shorter sequence. The composition sequence ca^2ca^2 for the constant 1 is of special interest. Reading the sequence from left to right, consider the range of the function obtained so far. When c is applied to the whole set $N = \{1, 2, 3\}$, we get the range $\{2, 3\}$. When a is applied to the latter, we get the range $\{1, 3\}$, and so forth. Altogether we get the sequence of ranges

$$\{1, 2, 3\}, \{2, 3\}, \{1, 3\}, \{1, 2\}, \{2\}, \{3\}, \{1\}.$$

It has no repetitions and contains all nonempty subsets of N. In the sequel we will express these two properties by saying that the composition sequence ca^2ca^2 is *range-reduced* and *range-complete*. Each of the words

$$cabca^2,\ cabcba,\ ca^2cba,\ ca^2ca^2$$

has these two properties.

Since there are altogether 120 nonempty words of length ≤ 4, some functions possess many representations using such words. The greatest number is possessed by the function with the value sequence 223 which has 17 such representations.

3 Depth of Compositions

We now discuss the *length* of composition sequences. The *depth* of a function f *with respect* to the set $\mathcal{F}$, in symbols $D(\mathcal{F}, f)$, is the shortest composition sequence of f in terms of functions in $\mathcal{F}$. If no such sequence exists, the depth is defined to be ∞.

In general, the *depth of a function* f is defined by the equation

$$D(f) = \max(D(\mathcal{F}, f)),$$

where $\mathcal{F}$ ranges over all sets in terms of which f has a composition sequence. The depth $D(f)$ tells how long a composition sequence can be in the worst case. If a composition sequence w for f can be written in the form $w = w_1w_2w_3$, where the sequences w_1 and w_1w_2 define the same function, then also w_1w_3 is a composition sequence for f and, consequently, the original sequence w is not minimal. Since the total number of functions is n^n, we get the upper bound $D(f) \leq n^n$, for any f. The following result shows that no polynomial upper bound (in terms of n) can be obtained. The proof, [12], uses an idea applied in the discussion of the payoff for the transition from a nondeterministic to a deterministic finite automaton. For completeness, we outline the proof also here.

Theorem 3. *There is no polynomial $P(n)$ such that $D(f) \leq P(n)$ holds for all functions f.*

Proof. Let p_i be the ith prime, and consider numbers n of the form $n = p_1 + p_2 + \cdots + p_k$. Let a be a permutation, defined as the product of k cycles of lengths $p_1, p_2, \ldots, p_k$. Let the set $\mathcal{F}$ consist of a only. Let the target function f be the identity function. Clearly,

$$D(\mathcal{F}, f) = p_1p_2 \cdots p_k.$$

Now the well-known estimate $p_k \leq k^2$, $k > 1$, leads easily to the desired result. □

The proof shows also that there are specific functions having no polynomial upper bound for their depth. The method is quite general: instead of the identity function we can choose, for instance, the function mapping each element in a cycle to the preceding element.

The *complete depth* $D_C(f)$ of a function f is defined also by the equation $D_C(f) = \max(D(\mathcal{F}, f))$ but now $\mathcal{F}$ ranges over *complete* sets of functions. It follows by the definition that every function f satisfies $D_C(f) \leq D(f)$. However, lower bounds such as the one given for $D(f)$ in the proof of Theorem 3 are much harder to obtain for $D_C(f)$, for the simple reason that we have much less leeway if we have to restrict the attention to complete sets $\mathcal{F}$ only. On the other hand, we do not know examples of functions for which the above inequality is strict. Such functions do not exist if $n = 2$, and also probably not for $n = 3$. It still seems that the following conjecture, presented also in [12], holds.

Conjecture 1. Assume that $n \geq 4$. Then there is a function f with the property

$$D_C(f) < D(f).$$

Consider, finally. again a set $\mathcal{F}$ of functions. By definition, a composition sequence w for a function f is *reduced* if no sequence obtained from w by removing some letters is a composition sequence for f. Clearly, a minimal composition sequence is always reduced. The converse does not necessarily hold, many examples can be obtained from the table in the preceding section. For instance, consider the function with the value sequence 123. The composition sequence a^3 is reduced but not minimal because also b^2 is a composition sequence for this function. A composition sequence w is not reduced if it can be written in the form $w = w_1w_2w_3$, where the sequences w_1 and w_1w_2 define the same function.

4 A Combinatorial Lemma

No good characterizations are known for synchronizing automata. Of course synchronizability is a decidable property but, as will be seen in the sequel, it is often very challenging to find out of a particular automaton whether or not it is synchronizing. There are special cases where it is easier to tell whether or not a given automaton is synchronizing.

The following result is useful in case of *circular automata*, that is, automata where one of the letters affects a circular permutation of the states.

Lemma 1. *Let $N = \{1, 2, \ldots, n\} = N_1 \cup \ldots \cup N_k$, $k \geq 2$, be a partition of N into nonempty pairwise disjoint subsets, not all of the same cardinality, let x_i, $1 \leq i \leq k$, be different elements of N, and let $P = (12 \ldots n)$ be a circular permutation. Then some power P^t of P maps two of the elements x_i, $1 \leq i \leq k$, into the same subset N_j.*

Proof. Assume that N_p, $1 \leq p \leq k$, is a subset of maximal cardinality, that is, no other subset has more elements than N_p. Consider the sets

$$M_j = P^{x_j}(N_p),\ 1 \leq j \leq k.$$

They are all of the same cardinality as N_p. Because the sets N_j are not all of the same cardinality, the sets M_j cannot be pairwise disjoint. Let $\mu \neq \nu$ be such that M_μ and M_ν intersect. Hence, also the sets

$$P^{-x_\nu}(M_\nu) = N_p \text{ and } P^{-x_\nu}(M_\mu) = P^{x_\mu - x_\nu}(N_p)$$

intersect. (The exponents of P denote smallest nonnegative remainders modulo n.) Hence, there is an element $\alpha \in N_p$ such that also the element $P^{x_\mu - x_\nu}(\alpha) = \beta \in N_p$. Choose now t such that $P^t(x_\mu) = \beta$. Then

$$P^t(x_\nu) = P^t(P^{x_\nu - x_\mu}(x_\mu)) = P^{x_\nu - x_\mu}(P^t(x_\mu)) = P^{x_\nu - x_\mu}(\beta) = \alpha \in N_p.$$

Hence, P^t maps both of the elements x_μ and x_ν into N_p. □

The cardinality assumption is necessary in Lemma 1. For instance, assume that

$$n = 4,\ N_1 = \{1, 2\},\ N_2 = \{3, 4\},\ x_1 = 1,\ x_2 = 3.$$

Then no power of the permutation (1234) maps both x_1 and x_2 into the same set N_j.

A somewhat weaker version of Lemma 1 was established in [15], where it was also assumed that n is prime and that $x_i \in N_i$, for all i, $1 \leq i \leq k$.

Assume that the letter b gives rise to a function of genus $k < n$ in a circular automaton. Then, by Lemma 1, a function of genus $< k$ can be generated as follows. First number the states in such a way that the letter a affects the circular permutation $(12 \ldots n)$. Further, for $i = 1, \ldots, k$, let N_i be the set of arguments for which b assumes the value x_i. Then ba^tb, where t is the exponent from Lemma 1, is of genus $< k$.

However, Lemma 1 is not strong enough to show that the automaton is synchronizing because sometimes the procedure cannot be repeated. An example is given in Figure 2. The function b is of genus 3. Functions of genus 2 can be generated but no function of genus 1. The automaton is not synchronizing.

5 Synchronizing Words

The phenomenon depicted in Figure 2 never occurs if n is a prime number. In this case the subsets corresponding to the constructed function of a smaller genus cannot be all of the same cardinality, which enables repeated applications of Lemma 1. Hence, the following theorem holds true. (The result should be credited to [15] because of the central lemma.)

Theorem 4. *A circular automaton with n states is synchronizing if and only if it contains a letter giving rise to a function of genus $< n$.*

Although no good general characterizations of synchronizability are known, the study of synchronous automata has a long history, going back to the classical paper [7] about experimenting with finite automata. When one does not know the state the automaton is in, one gets the situation under control if a synchronizing word is available. The early paper [5] is related.

A technique common in many-valued logic can be used for constructing classes of non-synchronizable automata. We say that a function g is *self-conjugate* under a permutation P if it satisfies the equation $g = PgP^{-1}$. It is easy to see that, if in an automaton with the state set N every function is self-conjugate under a permutation P mapping no element of N into itself, then the automaton is not synchronizable. This follows because P commutes with every generated function but clearly does not commute with any constant.

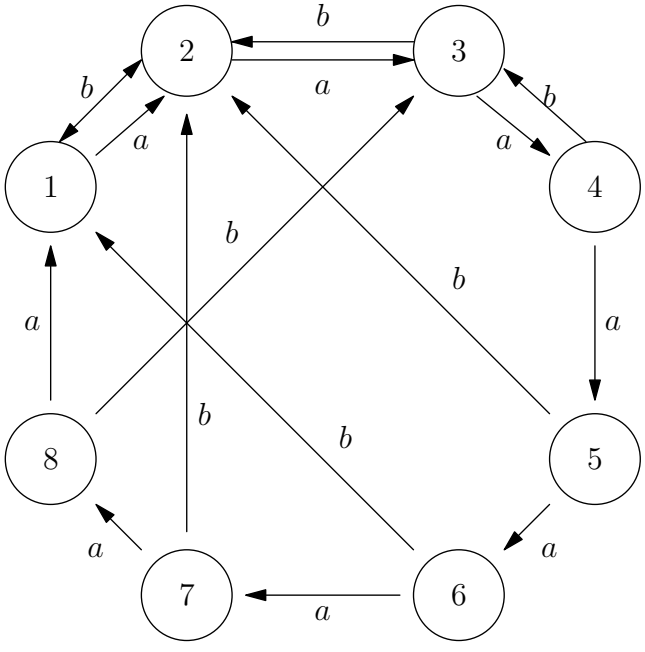


Fig. 2. Lemma 1 is applicable only once

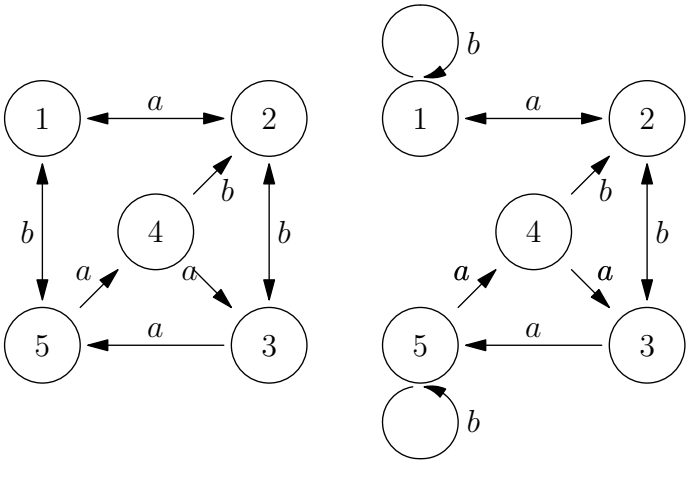


Fig. 3. Difference in synchronization

As an example, let n be even, $n = 2m$, and let the automaton $\mathcal{A}$ have two input letters a and b, where a affects the circular permutation $(12 \ldots n)$, and b maps the states $1, \ldots, m$ (resp. $m+1, \ldots, n$) into $m+1$ (resp. into 1).Then

$\mathcal{A}$ is not synchronizable, although many functions of genus 2 can be expressed as composition sequences in terms of a and b. Self-conjugacy is affected by the product of transpositions $P = (1, m+1)(2, m+2)\cdots(m, 2m)$.

Often a tiny difference makes a synchronizing automaton non-synchronizing. At a first look the left automaton in Figure 3 seems "more synchronizing" than the right one. However, the opposite is the case. The right automaton is synchronizing (with the synchronizing word $ba^2baba^2b^2aba^2b$), whereas the left one is not.

6 Decidability and Complexity

It is obvious, due to finite upper bounds, that all reasonable problems are *decidable* in our setup. For instance, given a function f and a set set $\mathcal{F}$, we can decide whether or not f is in the set generated by $\mathcal{F}$. This follows because of the trivial upper bound n^n for the length of minimal composition sequences. We can also test, by trying out all of the (finitely many) possibilities, whether or not a given composition sequence for f is minimal or whether it is reduced.

As far as complexity issues are concerned, practically all of the basic problems seem to be *intractable.* We will now present a result in this direction dealing with synchronization.

We will refer to the following problem as the *partial synchronization problem.* Given an automaton $\mathcal{A}$ and a subset Q_1 of its state set, one has to find a word w such that $qw = q'w$ holds for all states $q, q' \in Q_1$. Such a word w is *synchronizing for* Q_1. If such a word exists, $\mathcal{A}$ is called *synchronizable with respect to* Q_1. (Observe that the synchronizing state, that is, the state where the states in Q_1 are mapped by w, need not be in Q_1.)

Theorem 5. *The partial synchronization problem is NP-hard.*

Proof. We use reduction to SAT, the satisfiability problem for propositional formulas in conjunctive normal form. (The same argument applies also to the problem 3-SAT, where each of the disjunctive clauses contains only three terms.)

Consider a propositional formula in conjunctive normal form, having k variables $x_1, \ldots, x_k$, and consisting of a conjunction of l clauses of disjunctions $\beta_i,\ i = 1, \ldots, l$. We now choose $n = kl + 2$ and arrange the first $n - 2$ numbers $1, \ldots, kl$ in an array as follows.

$$\begin{array}{ccc} 1 & 2 & \cdots\ l \\ l+1 & l+2 & \cdots\ 2l \\ \vdots & \vdots & \vdots \\ (k-1)l+1 & (k-1)l+2 & \cdots\ kl \end{array}$$

We consider the automaton $\mathcal{A}$ with two input letters a and b, and having the transitions defined as follows. The states n and $n-1$ are "sinks" for both functions:

$$(n-1)a = (n-1)b = n-1,\ (n)a = (n)b = n.$$

For the states in the array above, that is, for the states up to $n-2$, the two functions are defined as follows. Consider the state $(i-1)l+j = u(i,j)$ in the position (i,j) in the array. If the variable x_i is not negated (resp. negated) in β_j, then a (resp. b) maps $u(i,j)$ to n. In all other cases both a and b map $u(i,j)$ to the next element $u(i+1,j)$ in the same column, with the following exception. The states $u(k,j)$ in the last row are mapped into $n-1$ by both a and b.

We now claim that $\mathcal{A}$ is synchronizable with respect to the set $\{1,\ldots,l\}$ exactly in case the original propositional formula is satisfiable.

Observe first that the columns in our array correspond to the clauses and the rows to the variables of the propositional formula, and that every word of length $\geq k$ maps every state to either $n-1$ or n.

Assume first that $t_1, t_2, \ldots, t_k$ is a truth value assignment for the variables $x_1, x_2, \ldots, x_k$ satisfying the formula. Let $c_1 c_2 \cdots c_k = w$ be the word such that $c_i = a$ (resp. $c_i = b$) if t_i is the truth value "true" (resp. "false"), for $i = 1, 2, \ldots, k$. Consider any number j, $1 \leq j \leq l$. Let x_i be a variable (there may be several of them) in the clause β_j satisfying β_j. (Thus, t_i is "true" or "false" according as x_i appears in β_j non-negated or negated.) By the definition of a and b, and by the choice of w,

$$(j)c_1 \ldots c_i = n = (j)w.$$

Because j was arbitrary, w is a synchronizing word for the set $\{1,\ldots,l\}$.

Conversely, assume that the word w is synchronizing for the set $\{1,\ldots,l\}$ Then also the prefix u of w of length k, $u = d_1 d_2 \cdots d_k$ is synchronizing for $\{1,\ldots,l\}$. We now define a truth value assignment $t_1, t_2, \ldots, t_k$ such that t_i is "true" (resp. "false") if $d_i = a$ (resp. $d_i = b$), for $1 \leq i \leq k$. Consider an arbitrary clause β_j. We know that $(j)u = n$. Consequently, for some i, $(j)d_1 \cdots d_i = n$. We choose the smallest such i and conclude that the assignment t_i for x_i satisfies β_j. Since again j was arbitrary, the truth value assignment thus defined satisfies the propositional formula.

To conclude the proof, we still have to take care of a minor detail. A truth value assignment satisfying *none* of the clauses leads also to a synchronizing word w because $xw = n-1$ for $x \in \{1,\ldots,l\}$. The existence of such an assignment can be excluded by having an identically true clause in the original propositional formula. □

7 The Černý Automaton

The automaton defined in Figure 4 was discussed in [2], and in more detail in [3]. It is connected with the following conjecture, often referred to as the problem of the longest open standing in finite automata theory.

Conjecture 2. **(Černý)** Every synchronizing automaton with n states has a synchronizing word of length $(n-1)^2$.

The number $(n-1)^2$ in the conjecture is usually presented as an upper bound. Our formulation is equivalent, since you can add arbitrary prefixes and suffixes to a synchronizing word.

There is an extensive literature concerning this conjecture. It has been established in numerous special cases, for instance, for circular automata in [4]. We refer to [6] for some overall considerations, and to [1,10,14] as some recent approaches.

However, all upper bounds so far obtained in the general case are cubic in n. It is also very difficult to construct examples where the bound $(n-1)^2$ is actually reached. In fact, the automaton of Figure 4 is the only known example for a general n. The other few known examples, where the upper bound is reached, are automata with a specific number, such as 4 or 6, of states. The only known functionally complete automata reaching the upper bound have less than 4 states. All this indicates that the value $(n-1)^2$ is not a natural borderline, that is, that the conjecture does not hold true.

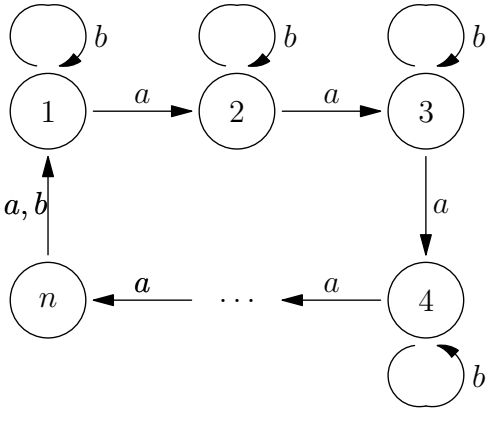


Fig. 4. The Černý automaton

Let us go back to our original framework of compositions of functions over a domain with n elements, and investigate the Černý automaton in this setup. Thus, we have a set $\mathcal{F}$ consisting of two functions a and b, where a is the circular permutation $(12\ldots n)$, and b maps n to 1 but keeps the numbers $1, 2, \ldots, n-1$ unchanged. Thus, the value sequences of a and b are $2, 3, \ldots, n, 1$ and $1, 2, \ldots, n-1, 1$, respectively. We consider the depth of the constant 1. Clearly, $(ba^{n-1})^{n-2}b$ is a sequence yielding the constant 1. But is it the shortest? For the sake of completeness (and since the matter is by no means clear), we outline here the argument originally due to [9].

Before we do this, let us try some other sequences. Since the aim is to reduce the genus from n to 1, one is tempted to apply the *greedy algorithm*, that is, to reduce the genus whenever possible. This makes sense because no backtracking is needed: if a correct sequence exists at all, any sequence can be continued to yield a correct one. For instance, let $n = 8$, giving the value sequences 23456781 and 12345671 for a and b. Then the sequence $(ba^2)^3b$ of length 10 reduces the genus to 4, as opposed to the prefix $(ba^7)^3b$ of length 25 of the orthodox sequence.

However, later on one pays the price because the short sequence yields the "bad" range $\{1, 3, 5, 7\}$ of cardinality 4. Indeed, in the continuation altogether length 20 (versus 33 in the orthodox sequence) is needed to get genus 3, length 33 (versus 41) to get genus 2, and length 61 (versus 49) to get genus 1. In the genus 2 one starts with the range $\{1, 5\}$ and has to go through all 28 pairs! Hence, often and certainly in case of the Černý automaton, the greedy algorithm leads to a synchronizing word longer than $(n-1)^2$.

We now show that one cannot do better than $(n-1)^2$. Consider a circle, where n spots at equal distances have been clockwise marked by the numbers $1, \ldots, n$. At the beginning each spot carries a stone. You have two possible moves. In the move a you transfer every stone to the spot lying clockwise next to the preceding spot of the stone. In the move b you transfer the stone in the spot n, if any, to the spot 1 and remove the stone, if any, from the spot 1. Other stones are left intact in b. The purpose is to reach a situation, where only one stone remains. What is the minimal number of moves for this? Considering how the moves were defined, it is clear that the answer gives the length of the shortest composition sequence for the constant 1.

A *configuration* in our setup is a circular word x of length n over the binary alphabet, where 1 (resp. 0) indicates a position carrying a stone (resp. an empty position). Thus, 1^n is the initial configuration. The *characteristic number* of a configuration x is the length of the longest factor of x consisting of 0's. Thus, the purpose is to increase the characteristic number from 0 to $n-1$ in the fastest possible way.

The move a never changes the characteristic number. The move b may increase it. When it does, it always increases it by 1. This happens exactly in case there is a stone in the position n and the longest factor of 0's (or one of the longest factors if there are several of equal length) ends at the position $n-1$. (Under these conditions, the move b increases the characteristic number, no matter whether or not there is a stone in the position 1.) These observations imply that one cannot do better than, always after applying b, move the stone in the position 1, as well as the empty spaces following it, $n-1$ steps ahead by the rule a. (Greedy actions to reduce genus do not increase the characteristic number!) But this gives exactly the composition sequence $(ba^{n-1})^{n-2}b$.

Instead of considering shortest composition sequences for constants, as is done in connection with the Černý automaton, one may try to construct longest possible composition sequences for constants. This can be viewed as a *dual* approach. Of course there is no upper bound, unless we make the natural assumption that the composition sequence must be *range-reduced*: no two prefixes of the composition sequence have the same range. Considering the total number of all possible ranges, we see that $2^n - 2$ is an absolute upper bound for the length. How many functions are needed to reach the upper bound? In Section 3 we gave the example ca^2ca^2 for the constant 1. Thus, for $n = 3$, two functions suffice.

For $n = 4$, consider the three functions a, b, c defined by the value sequences 2341, 2134, 1231. Then $ca^3ca^2cabca^3$ is a range-reduced composition sequence of

the maximal length 14 for the constant 4. (Clearly, range-reduced composition sequences of maximal length are always composition sequences for constants.)

How many functions must $\mathcal{F}$ have in order that $\mathcal{G}(\mathcal{F})$ contains a function having a range-reduced composition sequence of maximal length. Let $\gamma(n)$ be the least number of functions needed for this purpose if the basic set has n elements. (Observe that the upper bound $2^n - 2$ is not necessarily reached even if we consider complete sets $\mathcal{F}$ only.)

We saw above that $\gamma(3) = 2$ and $\gamma(4) = 3$. (It is easy to see that one cannot reduce these numbers.) Although $\gamma(n)$ increases with n, it seems likely that this does not go on indefinitely. We would like to conclude this paper with the following conjecture.

Conjecture 3. There is a number C such that $\gamma(n) \leq C$, for any n.

Dedication. This paper is dedicated to *Cris Calude* on the occasion of his 60th birthday. Friendship with Cris and our scientific collaboration have meant a lot to me. I knew of his work already around 1980 but after 1990 we have met on various occasions in Romania, Finland and elsewhere in Europe, in Canada and New Zealand. From the very beginning I was impressed by his sophistication in algorithmic information theory and related matters. Cris was always ready to carry the heaviest burden in our common editorial works. He is a great person to have as scientific collaborator and friend. I wish him continuing success in science and happiness in life in general.

References

1. Berlinkov, M.V.: On a Conjecture by Carpi and D'Alessandro. In: Gao, Y., Lu, H., Seki, S., Yu, S. (eds.) DLT 2010. LNCS, vol. 6224, pp. 66–75. Springer, Heidelberg (2010)
2. Černý, J.: Poznámka k homogénnym experimentom s konečnými automatmi. Mat.fyz.čas SAV 14, 208–215 (1964)
3. Černý, J., Pirická, A., Rosenauerová, B.: On directable automata. Kybernetika 7, 289–298 (1971)
4. Dubuc, L.: Sur les automates circulaires et la conjecture de Černý. Informatique Théorique et Applications 32, 21–34 (1998)
5. Ginsburg, S.: On the length of the smallest uniform experiment which distinguishes the terminal states of a machine. J. Assoc. Comput. Mach. 5, 266–280 (1958)
6. Mateescu, A., Salomaa, A.: Many-valued truth functions, Černý's conjecture and road coloring. EATCS Bulletin 68, 134–150 (1999)
7. Moore, E.F.: Gedanken experiments on sequential machines. In: Shannon, C.E., McCarthy, J. (eds.) Automata Studies, pp. 129–153. Princeton University Press (1956)
8. Piccard, S.: Sur les bases du groupe symétrique et les couples de substitutions qui engendrent un goupe régulier. Librairie Vuibert, Paris (1946)
9. Pin, J.-E.: Le problème de la synchronisation, Contribution a l'étude de la conjecture de Černý. Thèse de 3^{e} cycle a l'Université Pierre et Marie Curie, Paris 6 (1978)

10. Pribavkina, E.V., Rodaro, E.: Synchronizing automata with finitely many minimal synchronizing words. Inform. Comput. 209, 568–579 (2011)
11. Salomaa, A.: A theorem concerning the composition of functions of several variables ranging over a finite set. J. Symb. Logic 25, 203–208 (1960)
12. Salomaa, A.: Composition sequences for functions over a finite domain. Theor. Comput. Sci. 292, 263–281 (2003)
13. Stanković, R.S., Astola, J.T. (eds.): Reprints from the Early Days of Information Sciences: On the Contributions of Arto Salomaa to Multiple-Valued Logic. In: Tampere International Center for Signal Processing, TICSP Series, vol. 50 (2009)
14. Steinberg, B.: The Averaging Trick and the Černý Conjecture. In: Gao, Y., Lu, H., Seki, S., Yu, S. (eds.) DLT 2010. LNCS, vol. 6224, pp. 423–431. Springer, Heidelberg (2010)
15. Yablonskii, S.V.: Functional constructions in k-valued logic (in Russian). Tr. Matem. Inst. im. V.A. Steklova 51(5), 5–142 (1958)

On the Connected Partition Dimension of a Wheel Related Graph

Ioan Tomescu

Faculty of Mathematics and Computer Science
University of Bucharest
Str. Academiei, 14, 010014 Bucharest, Romania
ioan@fmi.unibuc.ro

Abstract. Let G be a connected graph. For a vertex $v \in V(G)$ and an ordered k-partition $\Pi = \{S_1, S_2, ..., S_k\}$ of $V(G)$, the representation of v with respect to Π is the k-vector $r(v|\Pi) = (d(v, S_1), d(v, S_2), ..., d(v, S_k))$, where $d(v, S_i)$ denotes the distance between v and S_i. The k-partition Π is said to be resolving if the k-vectors $r(v|\Pi)$, $v \in V(G)$, are distinct. The minimum k for which there is a resolving k-partition of $V(G)$ is called the partition dimension of G, denoted by $pd(G)$. If each subgraph $< S_i >$ induced by S_i $(1 \leq i \leq k)$ is required to be connected in G, the corresponding notions are connected resolving k-partition and connected partition dimension of G, denoted by $cpd(G)$. Let the graph J_{2n} be obtained from the wheel with $2n$ rim vertices W_{2n} by alternately deleting n spokes. In this paper it is shown that for every $n \geq 4$ $pd(J_{2n}) \leq 2\lceil\sqrt{2n}\rceil + 1$ and $cpd(J_{2n}) = \lceil(2n+3)/5\rceil$ applying Chebyshev's theorem and an averaging technique.

Keywords: distance, resolving partition, partition dimension, connected partition dimension, Bertrand's postulate.

1 Introduction

If G is a connected graph, the distance $d(u, v)$ between two vertices u and v in G is the length of a shortest path between them. The diameter of G is the largest distance between two vertices in $V(G)$. For a vertex v of a graph G and a subset S of $V(G)$, the distance between v and S is $d(v, S) = \min\{d(v, x)|x \in S\}$. Let $\Pi = \{S_1, S_2, \ldots, S_k\}$ be an ordered k-partition of vertices of G and let v be a vertex of G. The representation $r(v|\Pi)$ of v with respect to Π is the k-tuple $(d(v, S_1), d(v, S_2), \ldots, d(v, S_k))$. If distinct vertices of G have distinct representations with respect to Π, then Π is called a resolving partition for $V(G)$. The cardinality of a minimum resolving partition is called the partition dimension of G, denoted by $pd(G)$ [3]. A resolving partition $\Pi = \{S_1, S_2, ..., S_k\}$ of $V(G)$ is called connected if each subgraph $< S_i >$ induced by S_i $(1 \leq i \leq k)$ is connected in G. The minimum k for which there is a connected resolving k-partition of $V(G)$ is called the connected partition dimension of G, denoted by $cpd(G)$ [10].

M.J. Dinneen et al. (Eds.): WTCS 2012 (Calude Festschrift), LNCS 7160, pp. 417–424, 2012.

The concepts of resolvability have previously appeared in the literature (see [4, 6–9]). These concepts have some applications in chemistry for representing chemical compounds [4], to problems of pattern recognition and image processing, some of which involve the use of hierarchical data structures [7] and physics [2].

If $d(x,S) \neq d(y,S)$ we shall say that the class S separates vertices x and y. If a class S of Π separates vertices x and y we shall also say that Π separates x and y. From these definitions it can be observed that the property of a given partition Π of the vertices of a graph G to be a resolving partition of $V(G)$ can be verified by investigating the pairs of vertices in the same class. Indeed, every vertex $x \in S_i$ $(1 \leq i \leq k)$ is at distance 0 from S_i, but is at a distance different from zero from any other class S_j with $j \neq i$. It follows that $x \in S_i$ and $y \in S_j$ are separated both by S_i and by S_j for every $i \neq j$.

The wheel W_n for $n \geq 3$ is the graph $C_n + K_1$ obtained by joining all vertices of a cycle $C_n = v_0, v_1, \ldots, v_{n-1}$ to a further vertex c called the center. Thus W_n contains $n+1$ vertices, the center and n rim vertices and has diameter 2. Let the graph denoted by J_{2n} be the graph obtained by joining vertices $v_0, v_2, v_4, \ldots, v_{2n-2}$ of a cycle $C_{2n} = v_0, v_1, \ldots, v_{2n-1}, v_0$ to a further vertex c called the center. This graph is also known as the gear graph [5] or Jahangir graph [12]. Thus J_{2n} is a bipartite graph, having $2n+1$ vertices, the center and $2n$ rim vertices and has diameter 4 when $n \geq 4$. Note that $J_4 \cong K_{2,3}$. A rim vertex of degree 3 will be called a major vertex and a rim vertex of degree 2 a minor vertex. Thus J_{2n} has n major and n minor vertices. In this paper we consider the partition dimension as well as the connected partition dimension of J_{2n} for any integer $n \geq 2$.

2 Main Results

In this section we determine first an upper bound for $pd(J_{2n})$, but the question of determining the exact value of this parameter for J_{2n} remains unsettled.

Theorem 1. *For every $n \geq 2$ we have $pd(J_{2n}) \leq p+1$, where p is the smallest prime number such that $p(p-1) \geq 2n$.*

Proof. Consider first $n = 2$. We have $pd(J_4) = 3$, $p = 3$ and the conclusion of the theorem is true. Let $n \geq 3$ and p be the smallest prime number such that $p(p-1) \geq 2n$. Since p is prime, the sequence $0, i, 2i, 3i, \ldots, (p-1)i$, where $1 \leq i \leq p-1$ and all numbers are reduced modulo p, is a permutation of the set $\{0, 1, \ldots, p-1\}$. Consider the sequence $(x_j)_{j=1,\ldots,p(p-1)} = X_1, X_2, \ldots, X_{(p-1)/2}$, where for each $1 \leq i \leq (p-1)/2$ the subsequence

$$X_i = 0, 0, i, i, 2i, 2i, 3i, 3i, \ldots, (p-1)i, (p-1)i$$

contains $2p$ terms and each pair of equal elements different from $0, 0$ is obtained from the previous one by adding i modulo p to each component. The resolving partition $\Pi = \{S_1, \ldots, S_{p+1}\}$ of $V(J_{2n})$ is defined as follows:

a) if $2n = p(p-1)$ then $S_{p+1} = \{c\}$ and each element v_i $(0 \leq i \leq 2n-1)$ is assigned to the class $S_{x_{i+1}+1}$;
b) if $2n < p(p-1)$ then $S_{p+1} = \{c, v_{2n-1}\}$ and each element v_i $(0 \leq i \leq 2n-2)$ is assigned to the class $S_{x_{i+1}+1}$.

From the construction it can be observed that for any two vertices v_i, v_{i+1} in the same class, vertices v_{i-1} and v_{i+2} belong to different classes. Also, if v_i and v_j belong to the same class S_p and $i < j$, $j \neq i+1$, then at least one pair of vertices from $\{v_{i-1}, v_{j-1}\}$, $\{v_{i-1}, v_{j+1}\}$, $\{v_{i+1}, v_{j-1}\}$, $\{v_{i+1}, v_{j+1}\}$ consists of vertices that belong to two classes S_q, S_r such that $q, r \neq p$ and $q \neq r$. In the case b) vertices c and v_{2n-1} can be separated by a class of Π. It follows that Π is a resolving partition of $V(J_{2n})$ having $p+1$ classes, which implies $pd(J_{2n}) \leq p+1$.

Corollary 1. *For every $n \geq 2$, $pd(J_{2n}) \leq 2\lceil\sqrt{2n}\rceil + 1$.*

Proof. Since prime number p must satisfy $p(p-1) \geq 2n$ we can take $p \geq \lceil\sqrt{2n}\rceil + 1$. We shall apply Bertrand's postulate, proved for the first time by Chebyshev, which asserts that for every $n \geq 1$, there is some prime number p with $n < p \leq 2n$ (see [1]). We deduce that there exists a prime number p such that $\lceil\sqrt{2n}\rceil < p \leq 2\lceil\sqrt{2n}\rceil$, hence $pd(J_{2n}) \leq p+1 \leq 2\lceil\sqrt{2n}\rceil + 1$.

Let $\Pi = \{S_1, \ldots, S_k\}$ be a connected resolving k-partition of $V(J_{2n})$ such that the center $c \in S_1$. Every class of Π different from S_1 induces a path consisting of consecutive vertices of C_{2n} and vertices of S_1 belonging to C_{2n} induce $r \geq 0$ disjoint paths $L_1, \ldots, L_r$ (numbered in the clockwise direction of running through C_{2n}), consisting each of consecutive vertices of C_{2n}. A sequence of consecutive vertices $v_i, v_{i+1}, \ldots, v_j$ on C_{2n} (indices are considered modulo $2n$) will be called a window if these vertices do not belong to S_1 but $v_{i-1}, v_{j+1} \in S_1$. It is clear that each window includes some classes of Π different from S_1. Each class of Π containing vertex v_i or v_j will be called a boundary class and v_i or v_j will be called a boundary vertex. Let W_i for $1 \leq i \leq r$ denote the window neighboring path L_i relatively to the clockwise direction. A path L_s, a window W_t or a class S_i of Π for $i \geq 2$ will be called an $\alpha - \beta$ path, window or class, respectively if it consists of consecutive vertices $v_p, v_{p+1}, \ldots, v_q$ and $d(v_p) = \alpha$ and $d(v_q) = \beta$ $(2 \leq \alpha, \beta \leq 3)$.

The following claims express some properties of paths L_i and classes $S_2, \ldots, S_k$ not containing the center in a resolving partition Π of $V(J_{2n})$:

Claim 1. Every 3–3 class has at most five vertices.

Indeed, if a 3–3 class has seven consecutive vertices $v_l, v_{l+1}, \ldots, v_{l+6}$, then v_{l+2} and v_{l+4} cannot be separated by any other class. In a similar way we can prove:

Claim 2. Any 2–3 class has at most four vertices.

Claim 3. Any 2–2 class contains at most three vertices.

Claim 4. $|L_i| \leq 5$ for every $1 \leq i \leq r$ if L_i is a 3–3 path.

Claim 5. $|L_i| \leq 4$ for every $1 \leq i \leq r$ if L_i is a 2–3 or a 3–2 path.

Claim 6. $|L_i| \leq 3$ for every $1 \leq i \leq r$ if L_i is a 2–2 path.

Claim 7. At most one path L_i $(1 \leq i \leq r)$ can be: a 3–3 path with five vertices, a 2–3 (or a 3–2) path with four vertices or a 2–2 path with three vertices and no two such paths can coexist.

Otherwise we can find two major vertices lying on two such paths having equal distances to all classes $S_2, \ldots, S_k$. Note that several L_i can be 3–3 paths containing three vertices each and these paths will be essential in obtaining a minimum connected resolving partition of $V(J_{2n})$.

Claim 8. No boundary class S_i $(i \geq 2)$ can be a 3–3 class with five vertices, nor a 3–2 class with four vertices having boundary vertex a major vertex.
If this situation occurs then two major vertices of S_i cannot be separated by Π.

Claim 9. Any window cannot contain a single class S_i of type: 2–2 with three vertices, unless either L_i or L_{i+1} (paths neighbouring S_i) contains only one vertex, or 2–3 with four vertices, or 3–3 with three or five vertices, since in these cases two vertices of S_i cannot be separated by Π.

Note that in a window still can exist a unique 2–3 class with two vertices.

Theorem 2. *We have* $cpd(J_{2n}) = \varphi(n)$*, where*

$$\varphi(n) = \begin{cases} 3 & \text{for } n = 2 \text{ or } n = 3, \\ \lceil \frac{2n+3}{5} \rceil & \text{for } n \geq 4. \end{cases}$$

Proof. We first show that $cpd(J_4) = cpd(J_6) = cpd(J_8) = 3$ by considering the following minimum connected resolving partitions:

- For $n = 2$, $S_1 = \{c, v_0\}$, $S_2 = \{v_1\}$, and $S_3 = \{v_2, v_3\}$.
- For $n = 3$, $S_1 = \{c, v_0, v_5\}$, $S_2 = \{v_1, v_2\}$, and $S_3 = \{v_3, v_4\}$.
- For $n = 4$, $S_1 = \{c, v_0, v_1, v_2, v_7\}$, $S_2 = \{v_3, v_4, v_5\}$, and $S_3 = \{v_6\}$.

Let $n \geq 5$ and $t = \lfloor n/5 \rfloor$. We shall define a resolving partition $\Pi = \{S_1, \ldots, S_{\varphi(n)}\}$ of $V(J_{2n})$ having $\varphi(n)$ classes as follows: $S_{2i} = \{v_{10(i-1)+3}, v_{10(i-1)+4}, v_{10(i-1)+5}, v_{10(i-1)+6}\}$ and $S_{2i+1} = \{v_{10(i-1)+7}, v_{10(i-1)+8}, v_{10(i-1)+9}\}$ for $1 \leq i \leq t$;

- if $n = 5t$, $S_1 = \{c, v_0, v_1, v_2, v_{10}, v_{11}, v_{12}, v_{20}, v_{21}, v_{22}, \ldots, v_{10t-10}, v_{10t-9}, v_{10t-8}\}$;
- if $n = 5t + 1$, $S_1 = \{c, v_0, v_1, v_2, v_{10}, v_{11}, v_{12}, \ldots, v_{10t-10}, v_{10t-9}, v_{10t-8}, v_{10t}, v_{10t+1}\}$;
- if $n = 5t + 2$, $S_1 = \{c, v_0, v_1, v_2, v_{10}, v_{11}, v_{12}, \ldots, v_{10t}, v_{10t+1}, v_{10t+2}\}$ and $S_{2t+2} = \{v_{10t+3}\}$;
- if $n = 5t + 3$, $S_1 = \{c, v_0, v_1, v_2, v_{10}, v_{11}, v_{12}, \ldots, v_{10t}, v_{10t+1}, v_{10t+2}, v_{10t+5}\}$ and $S_{2t+2} = \{v_{10t+3}, v_{10t+4}\}$;
- if $n = 5t + 4$, $S_1 = \{c, v_0, v_1, v_2, \ldots, v_{10t}, v_{10t+1}, v_{10t+2}, v_{10t+7}\}$ and $S_{2t+2} = \{v_{10t+3}, v_{10t+4}, v_{10t+5}\}$ and $S_{2t+3} = \{v_{10t+6}\}$.

It can be easily verified that any two elements in the same class have distinct representations and all these classes induce connected subgraphs, so $cpd(J_{2n}) \leq \varphi(n)$.

It remains to show that $cpd(J_{2n}) \geq \varphi(n)$ for all $n \geq 5$. Let $cpd(J_{2n}) = k$ and $\Pi = \{S_1, S_2, \ldots, S_k\}$ be a connected resolving k-partition of $V(J_{2n})$. Suppose that $S_1 = \{c\}$. By Claim 1 we obtain that the number of classes different from S_1 is $k - 1 \geq \lceil 2n/5 \rceil$, hence $cpd(J_{2n}) \geq \lceil 2n/5 \rceil + 1 \geq \varphi(n)$. Otherwise, $S_1 \neq \{c\}$ and the vertices of S_1 belonging to C_{2n} induce $r \geq 1$ disjoint paths $L_1, \ldots, L_r$ on C_{2n} and r windows $W_1, \ldots, W_r$ containing $n_1, \ldots, n_r$ classes, respectively. We shall prove that

$$|L_i| + |W_i| \leq 5n_i \tag{1}$$

or

$$|L_i| + |W_i| + |L_{i+1}| + |W_{i+1}| \leq 5(n_i + n_{i+1}) \tag{2}$$

for almost all values $1 \leq i \leq r$ ($i + 1$ is considered modulo r) and for at most a single value j, $1 \leq j \leq r$ we have

$$|L_j| + |W_j| = 5n_j + \alpha, \tag{3}$$

or

$$|L_j| + |W_j| + |L_{j+1}| + |W_{j+1}| = 5(n_j + n_{j+1}) + \alpha, \tag{4}$$

where $\alpha \in \{1, 2\}$.

For this we shall use the following remark: For the clockwise direction of running through C_{2n} let $l_1, l_2, \ldots, l_{n_i}$ be the sizes of the classes encountered by starting from the path L_i in the window W_i. If for an index h, $1 \leq h < n_i$ we have

$$(|L_i| + \sum_{j=1}^{h} l_j)/h \leq 5, \tag{5}$$

then by Claim 1 we shall still have $|L_i| + |W_i| \leq 5n_i$ since any class has at most five vertices. Also, if for an index h, $1 \leq h < n_i$ we have

$$|L_i| + \sum_{j=1}^{h} l_j \leq 5h + \alpha, \tag{6}$$

where $\alpha \in \{1, 2\}$, then either (1) or (3) will hold for the window W_i.

Consider now a generic path L and its associated window W. Since $1 \leq |L| \leq 5$ we have five cases to analyze:

1) $|L| = 1$. In this case L consists of a major vertex. The boundary class from W can have at most four vertices (when it is a 2–3 class) and in this case $|L| + l_1 \leq 5$ and (5) is verified for $h = 1$.

2) $|L| = 2$. We have two subcases: 2a) L is a 3–2 class or 2b) L is a 2–3 class.

2a) In this case the boundary class S_{i_1} can have at most 5 vertices. If $|S_{i_1}| = 5$ by Claim 9 this class cannot be single in W. If the next class S_{i_2} has at most 3 vertices, then (5) is satisfied for $l_1 = 5, l_2 = 3$ and $h = 2$. Since S_{i_2} is a 2–3 or a 2–2 class it follows that $|S_{i_2}| \leq 4$. If $|S_{i_2}| = 4$ then by Claim 8 it follows that S_{i_2} cannot be the last class in W. The next class S_{i_3} is a 2–3 or a 2–2 class, hence $|S_{i_3}| \leq 4$ and in this case (5) is satisfied for $l_1 = 5, l_2 = l_3 = 4$ and $h = 3$.

If $|S_{i_1}| = 4$ then by Claim 9 this class is not single in W. The next class S_{i_2} is a 3–3 or a 3–2 class. If $|S_{i_2}| \leq 4$ then (5) is verified. Otherwise S_{i_2} is a 3–3 class with 5 vertices and by Claim 8 S_{i_2} cannot be the last class in W. The next class S_{i_3} is a 2–3 or a 2–2 class, which implies $|S_{i_3}| \leq 4$ and (5) holds for $h = 3$. Finally, if $|S_{i_1}| \leq 3$ then (5) holds for $h = 1$.

2b) In this case the boundary class S_{i_1} is a 2–2 or a 2–3 class, hence $|S_{i_1}| \leq 4$. If $|S_{i_1}| \leq 3$ then (5) is verified. Otherwise, S_{i_1} is a 2–3 class with 4 vertices that cannot be single in W. The next class S_{i_2} is a 2–2 or a 2–3 class; therefore $|S_{i_2}| \leq 4$ and (5) is verified for $h = 2$.

3) $|L| = 3$. We distinguish two subcases: 3a) L is a 3–3 path and 3b) L is a 2–2 path.

3a) If the boundary class S_{i_1} has $|S_{i_1}| \leq 2$ we are done. Otherwise $|S_{i_1}| \in \{3, 4\}$ since S_{i_1} may be a 2–2 or a 2–3 class. If S_{i_1} is a 2–2 class with 3 vertices and it is followed by a class S_{i_2} with at most 4 vertices then we are done. If S_{i_2} is a 3–3 class with 5 vertices, then S_{i_2} cannot be the last class in W and must be followed by S_{i_3} with at most 4 vertices and (5) is verified for $h = 3$.

If S_{i_1} is a 2–2 class with 3 vertices, single in its window, say $W = W_i$ having neighboring paths $L = L_i$ and L_{i+1}, then $|W_i| = 3$ and by Claim 9 at least one path L_i or L_{i+1} contains only one vertex, hence $|L_{i+1}| = 1$. In this case $L = L_i$ has three vertices and $n_i = 1$. In the windows W_{i+1} the boundary class T_{i_1} is a 2–2 or a 2–3 class. If T_{i_1} is a 2–2 class (case 3a.1) then $|T_{i_1}| \leq 3$ and if T_{i_1} is a 2–3 class (case 3a.2) then $|T_{i_1}| \leq 4$ and by Claim 9 it is followed by a class T_{i_2} which is a 2–2 or a 2–3 class, hence $|T_{i_2}| \leq 4$.

We have $|L_i| + |W_i| = 6$, $|L_{i+1}| = 1$, $n_i = 1$ and (2) is equivalent to

$$|W_{i+1}| \leq 5n_{i+1} - 2.$$

In case 3a.1 we have $|W_{i+1}| \leq 3 + 5(n_{i+1} - 1)$ and in case 3a.2 we deduce $|W_{i+1}| \leq 4 + 4 + 5(n_{i+1} - 2)$, since any class contains at most five vertices, which proves (2).

If S_{i_1} is a 2–3 class with 4 vertices and $|S_{i_2}| \leq 3$ we are done. Otherwise S_{i_2} is a 2–3 class with 4 vertices which is followed by at least a class S_{i_3} with at most 4 vertices and we are done.

3b) If $|S_{i_1}| \leq 4$ then (6) holds for $h = 1$. The remaining case is when S_{i_1} is a 3–3 class with 5 elements. If the next class S_{i_2} has 3 elements, then (6) is verified for $h = 2$ and $\alpha = 1$. If S_{i_2} is a 2–3 class with 4 elements, then by Claim 8 it cannot be the last class in the window. If $|S_{i_3}| \leq 3$ then (5) is verified for $h = 3$. Otherwise, S_{i_3} is a 2–3 class with 4 elements and it cannot be the last class. The next class S_{i_4} has at most 4 vertices because its first vertex is minor and (5) is true for $h = 4$.

4) $|L| = 4$. We also have two subcases depending upon whether L is a 3–2 or a 2–3 path. Since at most one path L has 4 vertices we can change the direction of running on C_{2n} and the second case is reduced to the first one. Hence we can consider that L is a 3–2 path with 4 vertices. If $|S_{i_1}| \leq 3$ then (6) holds for $h = 1$. Otherwise S_{i_1} is a 3–2 class with 4 vertices or a 3–3 class with 5 vertices. If S_{i_1} is a 3–2 class with 4 vertices and $|S_{i_2}| \leq 2$ then (5) is verified for $h = 2$. If

$|S_{i_2}| \leq 4$ then (6) is verified for $h = 2$. If S_{i_2} is a 3–3 class with 5 vertices, then by the boundary condition it cannot be the last class, $|S_{i_3}| \leq 4$ and (6) is true for $h = 3$. If S_{i_1} is a 3–3 class with 5 vertices and $|S_{i_2}| \leq 3$, then (6) is verified for $h = 2$. The remaining case is when S_{i_2} is a 2–3 class with 4 vertices. Since S_{i_2} is not the last class and $|S_{i_3}| \leq 4$, (6) holds for $h = 3$.

5) $|L| = 5$ and L is a 3–3 path with 5 vertices. If $|S_{i_1}| \leq 2$ then (6) is true for $h = 1$. It remains to consider the subcases when $|S_{i_1}| \in \{3, 4\}$. If S_{i_1} is a 2–2 class with 3 vertices, single in its window $W = W_j$, having neighboring paths $L = L_j$ and L_{j+1}, then $|W_j| = |S_{i_1}| = 3$ and by Claim 9 we have $|L_{j+1}| = 1$. As in the case 3a) we deduce that (4) holds.

If S_{i_1} is a 2–2 class with 3 elements followed by S_{i_2} and $|S_{i_2}| \leq 4$ then (6) is satisfied for $h = 2$. Otherwise, S_{i_2} is a 3–3 class with 5 elements. But it cannot be the last class and the next class S_{i_3} has at most 4 elements, hence (6) is satisfied for $h = 3$ and $\alpha = 2$.

If S_{i_1} is a 2–3 class with 4 elements and $|S_{i_2}| \leq 3$ then (6) holds for $h = 2$. For this subcase it remains to consider the situation when S_{i_2} is a 2–3 class with 4 elements. S_{i_2} cannot be the last class and the next class S_{i_3} has at most 4 elements whence (6) holds for $h = 3$.

By Claim 7 it follows that (1) and (2) hold for all $i = 1, \ldots, r$ with at most one exception j, when $|L_j| + |W_j| = 5n_j + \alpha \leq 5n_j + 2$, respectively $|L_j| + |W_j| + |L_{j+1}| + |W_{j+1}| = 5(n_j + n_{j+1}) + \alpha \leq 5(n_j + n_{j+1}) + 2$.

By summing up in an appropriate manner inequalities (1)–(4) we get

$$2n \leq 5k - 3, \text{ or } k \geq (2n + 3)/5$$

since $\sum_{i=1}^{r}(|L_i| + |W_i|) = 2n$ and $\sum_{i=1}^{r} n_i = k - 1$.

It follows that $cpd(J_{2n}) \geq \lceil (2n + 3)/5 \rceil$, which concludes the proof.

Note that $cpd(W_n) = \lceil (n + 2)/3 \rceil$ [11] and the metric dimension (in the sense of [6]) of J_{2n} equals $\lfloor 2n/3 \rfloor$ for every $n \geq 4$ [12].

References

1. Aigner, M., Ziegler, G.M.: Proofs from THE BOOK. Springer, Heidelberg (1999)
2. Calude, C.S., Hertling, P.H., Svozil, K.: Embedding quantum universes into classical ones. Foundations of Physics 29(3), 349–379 (1999)
3. Chartrand, G., Salehi, E., Zhang, P.: The partition dimension of a graph. Aequationes Math. 59, 45–54 (2000)
4. Chartrand, G., Eroh, L., Johnson, M.A., Oellermann, O.R.: Resolvability in graphs and the metric dimension of a graph. Discrete Appl. Math. 105, 99–113 (2000)
5. Gallian, J.A.: Dynamic Survey #DS6: Graph Labeling. Electronic J. Combin., 1–58 (2007), http://www.combinatorics.org/Surveys/
6. Harary, F., Melter, R.A.: On the metric dimension of a graph. Ars Combin. 2, 191–195 (1976)
7. Melter, R.A., Tomescu, I.: Metric bases in digital geometry. Computer Vision, Graphics, and Image Processing 25, 113–121 (1984)
8. Slater, P.J.: Leaves of trees. Congr. Numer 14, 549–559 (1975)

9. Slater, P.J.: Dominating and reference sets in a graph. J. Math. Phys. Sci. 22, 445–455 (1988)
10. Saenpholphat, V., Zhang, P.: Connected partition dimension of graphs. Discussiones Mathematicae Graph Theory 22, 305–323 (2002)
11. Tomescu, I., Javaid, I., Slamin: On the partition dimension and connected partition dimension of wheels. Ars Combin. 84, 311–317 (2007)
12. Tomescu, I., Javaid, I.: On the metric dimension of the Jahangir graph. Bulletin Math. Soc. Sci. Math. Roum. 50(98) 4, 371–376 (2007)

Author Index