6th International Conference on Computability and Complexity in Analysis

CCA 2009, August 18–22, 2009, Ljubljana, Slovenia

Edited by

Andrej Bauer
Peter Hertling
Ker-I Ko

OASIcs – Vol. 11 – CCA 2009 www.dagstuhl.de/oasics

Editors

Andrej Bauer
Faculty of Mathematics and Physics
University of Ljubljana
Jadranska 21
1000 Ljubljana, Slovenia
andrej.bauer@andrej.com

Peter Hertling
Fakultät für Informatik
Universität der
Bundeswehr München
85577 Neubiberg, Germany
peter.hertling@unibw.de

Ker-I Ko
Department of Computer Science
State University of
New York at Stony Brook
Stony Brook, NY 11794-4400, USA
keriko@cs.sunysb.edu

ACM Classification 1998
G. Mathematics of Computing, F.2 Analysis of Algorithms and Problem Complexity

ISBN 978-3-939897-12-5

Published online and open access by
Schloss Dagstuhl – Leibniz-Center for Informatics GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany.

Publication date
November, 2009.

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Digital Object Identifier: 10.4230/OASIcs.CCA.2009.i

ISBN 978-3-939897-12-5 **ISSN 2190-6807** **http://www.dagstuhl.de/oasics**

OASIcs – OpenAccess Series in Informatics

OASIcs aims at a suitable publication venue to publish peer-reviewed collections of papers emerging from a scientific event. OASIcs volumes are published according to the principle of Open Access, i.e., they are available online and free of charge.

ISSN 2190-6807

www.dagstuhl.de/oasics

Proceedings of the Sixth International Conference on Computability and Complexity in Analysis

CCA 2009

Ljubljana, Slovenia, August 18–22, 2009

Edited by:

Andrej Bauer, Peter Hertling, Ker-I Ko

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009
http://drops.dagstuhl.de/opus/volltexte/2009/2248

Editors

Andrej Bauer
Faculty of Mathematics and Physics
University of Ljubljana
Jadranska 21
1000 Ljubljana, Slovenia
E-mail: andrej.bauer@andrej.com

Peter Hertling
Fakultät für Informatik
Universität der Bundeswehr München
85577 Neubiberg, Germany
E-mail: peter.hertling@unibw.de

Ker-I Ko
Department of Computer Science
State University of New York at Stony Brook
Stony Brook, NY 11794-4400, USA
E-mail: keriko@cs.sunysb.edu

Published by: Schloss Dagstuhl – Leibniz-Zentrum für Informatik
ISBN: 978-3-939897-12-5

Conference Organization

Program Committee

Andrej Bauer, *Ljubljana, Slovenia*
Vasco Brattka, *Cape Town, South Africa*
Mark Braverman, *Cambridge, USA*
Pieter Collins, *Amsterdam, The Netherlands*
Peter Hertling, *Munich, Germany*, co-chair
Hajime Ishihara, *Ishikawa, Japan*
Ker-I Ko, *Stony Brook, USA*, co-chair
Robert Rettinger, *Hagen, Germany*
Victor Selivanov, *Novosibirsk, Russia*
Alex Simpson, *Edinburgh, Great Britain*
Dieter Spreen, *Siegen, Germany*
Frank Stephan, *Singapore*
Xizhong Zheng, *Glenside, USA*

Organizing Committee

Andrej Bauer, chair
Iztok Kavkler
Davorin Lešnik
Matija Pretnar

Invited Speakers

Mark Braverman, *Cambridge, USA*
Vladik Kreinovich, *El Paso, USA*
Dana Scott, *Pittsburgh, USA*
Ning Zhong, *Cincinnati, USA*

Tutorials

Martín Escardó, *Birmingham, UK*
Bas Spitters *and* Russell O'Connor, *Eindhoven, The Netherlands*

Table of Contents

Conference Organization iii

Table of Contents iv

Author Index vii

Preface 1
Andrej Bauer, Peter Hertling, and Ker-I Ko

Invited Talks

Computability and Complexity of Julia Sets (Invited Talk) 3
Mark Braverman

From Interval Computations to Constraint-Related Set Computations: Towards Faster Estimation of Statistics and ODEs under Interval and p-Box Uncertainty (Invited Talk) 5
Vladik Kreinovich

Semilattices, Domains, and Computability (Invited Talk) 17
Dana Scott

Computable Analysis of Differential Equations (Invited Talk) 19
Ning Zhong

Tutorials

Theory and Practice of Higher-type Computation (Tutorial) 21
Martín Escardó

Computer Verified Exact Analysis (Tutorial) 23
Bas Spitters and Russell O'Connor

Contributed Papers

Computing Conformal Maps onto Canonical Slit Domains 25
Valentin V. Andreev and Timothy H. McNicholl

Canonical Effective Subalgebras of Classical Algebras as Constructive Metric Completions 37
Andrej Bauer and Jens Blanck

Realisability and Adequacy for (Co)induction 49
Ulrich Berger

A Constructive Study of Landau's Summability Theorem 61
Josef Berger and Douglas Bridges

Separations of Non-monotonic Randomness Notions 71
Laurent Bienvenu, Rupert Hölzl, Thorsten Kräling, and Wolfgang Merkle

Weihrauch Degrees, Omniscience Principles and Weak Computability 83
Vasco Brattka and Guido Gherardi

Effective Choice and Boundedness Principles in Computable Analysis 95
Vasco Brattka and Guido Gherardi

Computability of Homology for Compact Absolute Neighbourhood Retracts 107
Pieter Collins

$\mathbf{\Sigma}^0_\alpha$-Admissible Representations (Extended Abstract) 119
Matthew de Brecht and Akihiro Yamamoto

Uniqueness, Continuity, and Existence of Implicit Functions in Constructive Analysis 131
Hannes Diener and Peter Schuster

Relativizations of the P =? DNP Question for the BSS Model 141
Christine Gaßner

Curves That Must Be Retraced 149
Xiaoyang Gu, Jack H. Lutz, Elvira Mayordomo

Effective Dispersion in Computable Metric Spaces 161
Zvonko Iljazović

On Oscillation-free ε-random Sequences II 173
Jöran Mielke and Ludwig Staiger

Computability of Probability Distributions and Distribution Functions . . . 185
Takakazu Mori, Yoshiki Tsujii, and Mariko Yasugi

How Discontinuous is Computing Nash Equilibria? (Extended Abstract) . 197
Arno Pauly

Towards the Complexity of Riemann Mappings (Extended Abstract) 209
Robert Rettinger

On the Computability of Rectifiable Simple Curve (Extended Abstract) . 221
Robert Rettinger and Xizhong Zheng

A Note on Closed Subsets in Quasi-zero-dimensional Qcb-spaces (Extended Abstract) 233
Matthias Schröder

Random Iteration Algorithm for Graph-Directed Sets 245
Yoshiki Tsujii, Takakazu Mori, Mariko Yasugi, and Hideki Tsuiki

Computable Separation in Topology, from $\boldsymbol{T_0}$ to $\boldsymbol{T_3}$ 257
Klaus Weihrauch

Real Computation with Least Discrete Advice: A Complexity Theory of Nonuniform Computability 269
Martin Ziegler

Author Index

Andreev, Valentin V. 25

Bauer, Andrej 37
Berger, Josef 61
Berger, Ulrich 49
Bienvenu, Laurent 71
Blanck, Jens 37
Brattka, Vasco 83, 95
Braverman, Mark 3
Bridges, Douglas 61

Collins, Pieter 107

de Brecht, Matthew 119
Diener, Hannes 131

Escardó, Martín 21

Gaßner, Christine 141
Gherardi, Guido 83, 95
Gu, Xiaoyang 149

Hölzl, Rupert 71

Iljazović, Zvonko 161

Kräling, Thorsten 71
Kreinovich, Vladik 5

Lutz, Jack H. 149

Mayordomo, Elvira 149
McNicholl, Timothy H. 25
Merkle, Wolfgang 71
Mielke, Jöran 173
Mori, Takakazu 185, 245

O'Connor, Russell 23

Pauly, Arno 197

Rettinger, Robert 209, 221

Schröder, Matthias 233
Schuster, Peter 131
Scott, Dana 17
Spitter, Bas 23
Staiger, Ludwig 173

Tsuiki, Hideki 245
Tsujii, Yoshiki 185, 245

Weihrauch, Klaus 257

Yamamoto, Akihiro 119
Yasugi, Mariko 185, 245

Zheng, Xizhong 221
Zhong, Ning 19
Ziegler, Martin 269

Preface

Andrej Bauer[1], Peter Hertling[2], and Ker-I Ko[3]

[1] Faculty of Mathematics and Physics, University of Ljubljana, Slovenia
[2] Fakultät für Informatik, Universität der Bundeswehr München, Germany
[3] Department of Computer Science, State University of New York at Stony Brook, USA

The Sixth International Conference on Computability and Complexity in Analysis, CCA 2009, took place on August 18 to 22, 2009, in Ljubljana, Slovenia. It was the 15th event in a series of workshops, seminars and conferences on CCA. For more information about CCA see `http://cca-net.de`.

The conference is concerned with Computable Analysis, the theory of computability and complexity over real-valued data. Computability theory studies the limitations and abilities of computers in principle. Computational complexity theory provides a framework for understanding the cost of solving computational problems, as measured by the requirement for resources such as time and space. In particular, Computable Analysis supplies an algorithmic foundation for numerical computation.

Scientists working in the area of computability and complexity over the real numbers and over more general continuous data structures come from different fields, such as theoretical computer science, domain theory, logic, constructive mathematics, computer arithmetic, numerical mathematics and all branches of analysis.

The conference program consisted of 4 invited talks, 2 tutorials of three talks each, and 24 contributed talks. These proceedings contain the abstracts or extended abstracts of the invited talks, tutorials, and a selection of 22 contributed articles. We thank all authors for their contributions, the program committee members and the additional referees for their careful refereeing work, and the organizing committee members for their work as well.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 1
http://drops.dagstuhl.de/opus/volltexte/2009/2249

Computability and Complexity of Julia Sets (Invited Talk)

Mark Braverman

Microsoft Research New England, Cambridge, MA, USA

Studying dynamical systems is key to understanding a wide range of phenomena ranging from planetary movement to climate patterns to market dynamics. Various numerical tools have been developed to address specific questions about dynamical systems, such as predicting the weather or planning the trajectory of a satellite. However, the theory of computation behind these problems appears to be very difficult to develop. In fact, little is known about computability of even the most natural problems arising from dynamical systems.

In this talk I will survey the recent study of the computational properties of dynamical systems that arise from iterating quadratic polynomials on the complex plane. These give rise to the amazing variety of fractals known as Julia sets, and are closely connected to the Mandelbrot set. Julia sets are perhaps the most drawn objects in Mathematics due to their fascinating fractal structure. The theory behind them is even more fascinating, and the dynamical systems generating them are in many ways archetypal. I will present both positive and negative results on the computability and complexity of Julia sets.

In conclusion of the talk I will discuss possible future directions and challenges in the study of the computability and complexity of dynamical systems.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 3
http://drops.dagstuhl.de/opus/volltexte/2009/2250

From Interval Computations to Constraint-Related Set Computations: Towards Faster Estimation of Statistics and ODEs under Interval and p-Box Uncertainty (Invited Talk)

Vladik Kreinovich

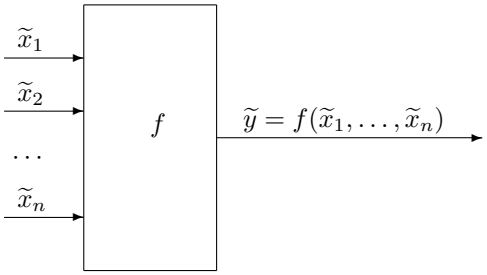
Department of Computer Science
University of Texas at El Paso, El Paso, TX 79968, USA
vladik@utep.edu

Abstract. Interval computations estimate the uncertainty of the result of data processing in situations in which we only know the upper bounds Δ on the measurement errors. In this case, based on the measurement result $\widetilde{x}$, we can only conclude that the actual (unknown) value x of the desired quantity is in the interval $[\widetilde{x} - \Delta, \widetilde{x} + \Delta]$.
In interval computations, at each intermediate stage of the computation, we have intervals of possible values of the corresponding quantities. As a result, we often have bounds with excess width. To remedy this problem, in our previous papers, we proposed an extension of interval technique to *set computations*, where on each stage, in addition to intervals of possible values of the quantities, we also keep sets of possible values of pairs (triples, etc.). In this paper, we show that in several practical problems, such as estimating statistics (variance, correlation, etc.) and solutions to ordinary differential equations (ODEs) with given accuracy, this new formalism enables us to find estimates in feasible (polynomial) time.

1 Formulation of the Problem

Need for data processing. In many real-life situations, we are interested in the value of a physical quantity y that is difficult or impossible to measure directly. Examples of such quantities are the distance to a star and the amount of oil in a given well. Since we cannot measure y directly, a natural idea is to measure y *indirectly*. Specifically, we find some easier-to-measure quantities $x_1, \ldots, x_n$ which are related to y by a known relation $y = f(x_1, \ldots, x_n)$; this relation may be a simple functional transformation, or complex algorithm (e.g., for the amount of oil, numerical solution to a partial differential equation). Then, to estimate y, we first measure or estimate the values of the quantities $x_1, \ldots, x_n$, and then we use the results $\widetilde{x}_1, \ldots, \widetilde{x}_n$ of these measurements (estimations) to compute an estimate $\widetilde{y}$ for y as $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 5-16
http://drops.dagstuhl.de/opus/volltexte/2009/2251



Computing an estimate for y based on the results of direct measurements is called *data processing*; data processing is the main reason why computers were invented in the first place, and data processing is still one of the main uses of computers as number crunching devices.

Measurement uncertainty: from probabilities to intervals. Measurement are never 100% accurate, so in reality, the actual value x_i of i-th measured quantity can differ from the measurement result $\widetilde{x}_i$. Because of these *measurement errors* $\Delta x_i \stackrel{\text{def}}{=} \widetilde{x}_i - x_i$, the result $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$ of data processing is, in general, different from the actual value $y = f(x_1, \ldots, x_n)$ of the desired quantity y.

It is desirable to describe the error $\Delta y \stackrel{\text{def}}{=} \widetilde{y} - y$ of the result of data processing. To do that, we must have some information about the errors of direct measurements.

What do we know about the errors Δx_i of direct measurements? First, the manufacturer of the measuring instrument must supply us with an upper bound Δ_i on the measurement error. If no such upper bound is supplied, this means that no accuracy is guaranteed, and the corresponding "measuring instrument" is practically useless. In this case, once we performed a measurement and got a measurement result $\widetilde{x}_i$, we know that the actual (unknown) value x_i of the measured quantity belongs to the interval $\mathbf{x}_i = [\underline{x}_i, \overline{x}_i]$, where $\underline{x}_i = \widetilde{x}_i - \Delta_i$ and $\overline{x}_i = \widetilde{x}_i + \Delta_i$.

In many practical situations, we not only know the interval $[-\Delta_i, \Delta_i]$ of possible values of the measurement error; we also know the probability of different values Δx_i within this interval. This knowledge underlies the traditional engineering approach to estimating the error of indirect measurement, in which we assume that we know the probability distributions for measurement errors Δx_i.

In practice, we can determine the desired probabilities of different values of Δx_i by comparing the results of measuring with this instrument with the results of measuring the same quantity by a standard (much more accurate) measuring instrument. Since the standard measuring instrument is much more accurate than the one use, the difference between these two measurement results is practically equal to the measurement error; thus, the empirical distribution of this difference is close to the desired probability distribution for measurement error. There are two cases, however, when this determination is not done:

– First is the case of cutting-edge measurements, e.g., measurements in fundamental science. When we use the largest particle accelerator to measure

the properties of elementary particles, there is no "standard" (much more accurate) located nearby that we can use for calibration: our accelerator is the best we have.
- The second case is the case of measurements in manufacturing. In principle, every sensor can be thoroughly calibrated, but sensor calibration is so costly – usually costing ten times more than the sensor itself – that manufacturers rarely do it.

In both cases, we have no information about the probabilities of Δx_i; the only information we have is the upper bound on the measurement error.

In this case, after we performed a measurement and got a measurement result $\widetilde{x}_i$, the only information that we have about the actual value x_i of the measured quantity is that it belongs to the interval $\mathbf{x}_i = [\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i]$. In such situations, the only information that we have about the (unknown) actual value of $y = f(x_1, \ldots, x_n)$ is that y belongs to the range $\mathbf{y} = [\underline{y}, \overline{y}]$ of the function f over the box $\mathbf{x}_1 \times \ldots \times \mathbf{x}_n$:

$$\mathbf{y} = [\underline{y}, \overline{y}] = f(\mathbf{x}_1, \ldots, \mathbf{x}_n) \stackrel{\text{def}}{=} \{f(x_1, \ldots, x_n) \,|\, x_1 \in \mathbf{x}_1, \ldots, x_n \in \mathbf{x}_n\}.$$

The process of computing this interval range based on the input intervals $\mathbf{x}_i$ is called *interval computations*; see, e.g., [4].

Outline. We start by recalling the basic techniques of interval computations and their drawbacks, then we will describe the new set computation techniques and describe a class of problems for which these techniques are efficient. Finally, we talk about how we can extend these techniques to other types of uncertainty (e.g., classes of probability distributions).

2 Interval Computations: Brief Reminder

Interval computations: main idea. Historically the first method for computing the enclosure for the range is the method which is sometimes called "straightforward" interval computations. This method is based on the fact that inside the computer, every algorithm consists of elementary operations (arithmetic operations, min, max, etc.). For each elementary operation $f(a, b)$, if we know the intervals $\mathbf{a}$ and $\mathbf{b}$ for a and b, we can compute the exact range $f(\mathbf{a}, \mathbf{b})$. The corresponding formulas form the so-called *interval arithmetic*:

$$[\underline{a}, \overline{a}] + [\underline{b}, \overline{b}] = [\underline{a} + \underline{b}, \overline{a} + \overline{b}]; \quad [\underline{a}, \overline{a}] - [\underline{b}, \overline{b}] = [\underline{a} - \overline{b}, \overline{a} - \underline{b}];$$

$$[\underline{a}, \overline{a}] \cdot [\underline{b}, \overline{b}] = [\min(\underline{a} \cdot \underline{b}, \underline{a} \cdot \overline{b}, \overline{a} \cdot \underline{b}, \overline{a} \cdot \overline{b}), \max(\underline{a} \cdot \underline{b}, \underline{a} \cdot \overline{b}, \overline{a} \cdot \underline{b}, \overline{a} \cdot \overline{b})];$$

$$1/[\underline{a}, \overline{a}] = [1/\overline{a}, 1/\underline{a}] \text{ if } 0 \notin [\underline{a}, \overline{a}]; \quad [\underline{a}, \overline{a}]/[\underline{b}, \overline{b}] = [\underline{a}, \overline{a}] \cdot (1/[\underline{b}, \overline{b}]).$$

In straightforward interval computations, we repeat the computations forming the program f step-by-step, replacing each operation with real numbers by the corresponding operation of interval arithmetic. It is known that, as a result, we get an enclosure $\mathbf{Y} \supseteq \mathbf{y}$ for the desired range.

From main idea to actual computer implementation. Not every real number can be exactly implemented in a computer; thus, e.g., after implementing an operation of interval arithmetic, we must enclose the result $[r^-, r^+]$ in a computer-representable interval: namely, we must round-off r^- to a smaller computer-representable value $\underline{r}$, and round-off r^+ to a larger computer-representable value $\overline{r}$.

Sometimes, we get excess width. In some cases, the resulting enclosure is exact; in other cases, the enclosure has excess width. The excess width is inevitable since straightforward interval computations increase the computation time by at most a factor of 4, while computing the exact range is, in general, NP-hard (see, e.g., [5]), even for computing the population variance $V = \frac{1}{n} \cdot \sum_{i=1}^{n} (x_i - \overline{x})^2$, where $\overline{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i$ (see [3]).

If we get excess width, then we can use more sophisticated techniques to get a better estimate, such as centered form, monotonicity, bisection, etc. (see, e.g., [4]). These methods usually decrease the excess width, but do not completely eliminate it.

Reason for excess width. The main reason for excess width is that intermediate results are dependent on each other, and straightforward interval computations ignore this dependence. For example, the actual range of $f(x_1) = x_1 - x_1^2$ over $\mathbf{x}_1 = [0, 1]$ is $\mathbf{y} = [0, 0.25]$. Computing this f means that we first compute $x_2 := x_1^2$ and then subtract x_2 from x_1. According to straightforward interval computations, we compute $\mathbf{r} = [0, 1]^2 = [0, 1]$ and then $\mathbf{x}_1 - \mathbf{x}_2 = [0, 1] - [0, 1] = [-1, 1]$. This excess width comes from the fact that the formula for interval subtraction implicitly assumes that both a and b can take arbitrary values within the corresponding intervals $\mathbf{a}$ and $\mathbf{b}$, while in this case, the values of x_1 and x_2 are clearly not independent: x_2 is uniquely determined by x_1, as $x_2 = x_1^2$.

3 Constraint-Based Set Computations

Main idea. The main idea behind constraint-based set computations (see, e.g., [1]) is to remedy the above reason why interval computations lead to excess width. Specifically, at every stage of the computations, in addition to keeping the *intervals* $\mathbf{x}_i$ of possible values of all intermediate quantities x_i, we also keep several *sets*:

- sets $\mathbf{x}_{ij}$ of possible values of pairs (x_i, x_j);
- if needed, sets $\mathbf{x}_{ijk}$ of possible values of triples (x_i, x_j, x_k); etc.

In the above example, instead of just keeping two intervals $\mathbf{x}_1 = \mathbf{x}_2 = [0, 1]$, we would then also generate and keep the set $\mathbf{x}_{12} = \{(x_1, x_1^2) \,|\, x_1 \in [0, 1]\}$. Then, the desired range is computed as the range of $x_1 - x_2$ over this set – which is exactly $[0, 0.25]$.

To the best of our knowledge, in interval computations context, the idea of representing dependence in terms of sets of possible values of tuples was first described by Shary; see, e.g., [6] and references therein.

How can we propagate this set uncertainty via arithmetic operations? Let us describe this on the example of addition, when, in the computation of f, we use two previously computed values x_i and x_j to compute a new value $x_k := x_i + x_j$. In this case, we set $\mathbf{x}_{ik} = \{(x_i, x_i + x_j) \,|\, (x_i, x_j) \in \mathbf{x}_{ij}\}$,

$$\mathbf{x}_{jk} = \{(x_j, x_i + x_j) \,|\, (x_i, x_j) \in \mathbf{x}_{ij}\},$$

and for every $l \neq i, j$, we take

$$\mathbf{x}_{kl} = \{(x_i + x_j, x_l) \,|\, (x_i, x_j) \in \mathbf{x}_{ij}, (x_i, x_l) \in \mathbf{x}_{il}, (x_j, x_l) \in \mathbf{x}_{jl}\}.$$

From main idea to actual computer implementation. In interval computations, we cannot represent an arbitrary interval inside the computer, we need an enclosure. Similarly, we cannot represent an arbitrary set inside a computer, we need an enclosure.

To describe such enclosures, we fix the number C of granules (e.g., $C = 10$). We divide each interval $\mathbf{x}_i$ into C equal parts $\mathbf{X}_i$; thus each box $\mathbf{x}_i \times \mathbf{x}_j$ is divided into C^2 subboxes $\mathbf{X}_i \times \mathbf{X}_j$. We then describe each set $\mathbf{x}_{ij}$ by listing all subboxes $\mathbf{X}_i \times \mathbf{X}_j$ which have common elements with $\mathbf{x}_{ij}$; the union of such subboxes is an enclosure for the desired set $\mathbf{x}_{ij}$.

This implementation enables us to implement all above arithmetic operations. For example, to implement $\mathbf{x}_{ik} = \{(x_i, x_i + x_j) \,|\, (x_i, x_j) \in \mathbf{x}_{ij}\}$, we take all the subboxes $\mathbf{X}_i \times \mathbf{X}_j$ that form the set $\mathbf{x}_{ij}$; for each of these subboxes, we enclosure the corresponding set of pairs $\{(x_i, x_i + x_j) \,|\, (x_i, x_j) \in \mathbf{X}_i \times \mathbf{X}_j\}$ into a set $\mathbf{X}_i \times (\mathbf{X}_i + \mathbf{X}_j)$. This set may have non-empty intersection with several subboxes $\mathbf{X}_i \times \mathbf{X}_k$; all these subboxes are added to the computed enclosure for $\mathbf{x}_{ik}$. Once can easily see if we start with the exact range $\mathbf{x}_{ij}$, then the resulting enclosure for $\mathbf{x}_{ik}$ is an $(1/C)$-approximation to the actual set – and so when C increases, we get more and more accurate representations of the desired set.

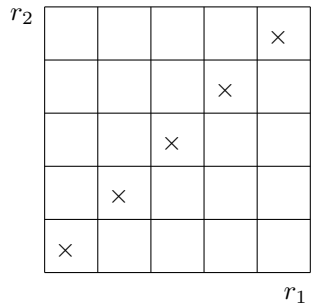
Similarly, to find an enclosure for

$$\mathbf{x}_{kl} = \{(x_i + x_j, x_l) \,|\, (x_i, x_j) \in \mathbf{x}_{ij}, (x_i, x_l) \in \mathbf{x}_{il}, (x_j, x_l) \in \mathbf{x}_{jl}\},$$

we consider all the triples of subintervals $(\mathbf{X}_i, \mathbf{X}_j, \mathbf{X}_l)$ for which $\mathbf{X}_i \times \mathbf{X}_j \subseteq \mathbf{x}_{ij}$, $\mathbf{X}_i \times \mathbf{X}_l \subseteq \mathbf{x}_{il}$, and $\mathbf{X}_j \times \mathbf{X}_l \subseteq \mathbf{x}_{jl}$; for each such triple, we compute the box $(\mathbf{X}_i + \mathbf{X}_j) \times \mathbf{X}_l$; then, we add subboxes $\mathbf{X}_k \times \mathbf{X}_l$ which intersect with this box to the enclosure for $\mathbf{x}_{kl}$.

First example: computing the range of $x - x$. For $f(x) = x - x$ on $[0, 1]$, the actual range is $[0, 0]$, but straightforward interval computations lead to an enclosure $[0, 1] - [0, 1] = [-1, 1]$. In straightforward interval computations, we have $r_1 = x$ with the exact interval range $\mathbf{r}_1 = [0, 1]$, and we have $r_2 = x$ with the exact interval range $\mathbf{x}_2 = [0, 1]$. The variables r_1 and r_2 are dependent, but we ignore this dependence.

In the new approach: we have $\mathbf{r}_1 = \mathbf{r}_2 = [0, 1]$, and we also have $\mathbf{r}_{12}$:



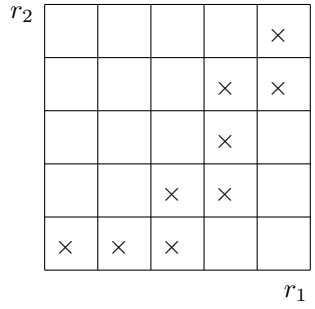
For each small box, we have $[-0.2, 0.2]$, so the union is $[-0.2, 0.2]$.

If we divide into more pieces, we get an interval closer to 0.

Second example: computing the range of $x - x^2$. In straightforward interval computations, we have $r_1 = x$ with the exact interval range interval $\mathbf{r}_1 = [0, 1]$, and we have $r_2 = x^2$ with the exact interval range $\mathbf{x}_2 = [0, 1]$. The variables r_1 and r_2 are dependent, but we ignore this dependence and estimate $\mathbf{r}_3$ as $[0, 1] - [0, 1] = [-1, 1]$.

In the new approach: we have $\mathbf{r}_1 = \mathbf{r}_2 = [0, 1]$, and we also have $\mathbf{r}_{12}$. First, we divide the range $[0, 1]$ into 5 equal subintervals $\mathbf{R}_1$. The union of the ranges $\mathbf{R}_1^2$ corresponding to these 5 subintervals $\mathbf{R}_1$ is $[0, 1]$, so $\mathbf{r}_2 = [0, 1]$. We divide this interval $\mathbf{r}_2$ into 5 equal sub-intervals $[0, 0.2]$, $[0.2, 0.4]$, etc. We now compute the set $\mathbf{r}_{12}$ as follows:

- for $\mathbf{R}_1 = [0, 0.2]$, we have $\mathbf{R}_1^2 = [0, 0.04]$, so only sub-interval $[0, 0.2]$ of the interval $\mathbf{r}_2$ is affected;
- for $\mathbf{R}_1 = [0.2, 0.4]$, we have $\mathbf{R}_1^2 = [0.04, 0.16]$, so also only sub-interval $[0, 0.2]$ is affected;
- for $\mathbf{R}_1 = [0.4, 0.6]$, we have $\mathbf{R}_1^2 = [0.16, 0.36]$, so two sub-intervals $[0, 0.2]$ and $[0.2, 0.4]$ are affected, etc.

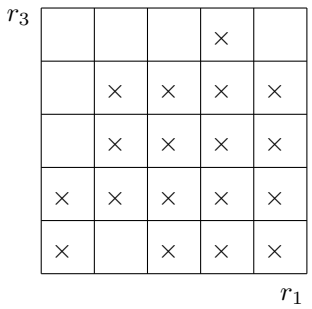


For each possible pair of small boxes $\mathbf{R}_1 \times \mathbf{R}_2$, we have $\mathbf{R}_1 - \mathbf{R}_2 = [-0.2, 0.2]$, $[0, 0.4]$, or $[0.2, 0.6]$, so the union of $\mathbf{R}_1 - \mathbf{R}_2$ is $\mathbf{r}_3 = [-0.2, 0.6]$.

If we divide into more and more pieces, we get the enclosure which is closer and closer to the exact range $[0, 0.25]$.

How to compute $\mathbf{r}_{ik}$. The above example is a good case to illustrate how we compute the range $\mathbf{r}_{13}$ for $r_3 = r_1 - r_2$. Indeed, since $\mathbf{r}_3 = [-0.2, 0.6]$, we divide this range into 5 subintervals $[-0.2, -0.04]$, $[-0.04, 0.12]$, $[0.12, 0.28]$, $[0.28, 0.44]$, $[0.44, 0.6]$.

- For $\mathbf{R}_1 = [0, 0.2]$, the only possible $\mathbf{R}_2$ is $[0, 0.2]$, so $\mathbf{R}_1 - \mathbf{R}_2 = [-0.2, 0.2]$. This covers $[-0.2, -0.04]$, $[-0.04, 0.12]$, and $[0.12, 0.28]$.
- For $\mathbf{R}_1 = [0.2, 0.4]$, the only possible $\mathbf{R}_2$ is $[0, 0.2]$, so $\mathbf{R}_1 - \mathbf{R}_2 = [0, 0.4]$. This interval covers $[-0.04, 0.12]$, $[0.12, 0.28]$, and $[0.28, 0.44]$.
- For $\mathbf{R}_1 = [0.4, 0.6]$, we have two possible $\mathbf{R}_2$:
  - for $\mathbf{R}_2 = [0, 0.2]$, we have $\mathbf{R}_1 - \mathbf{R}_2 = [0.2, 0.6]$; this covers $[0.12, 0.28]$, $[0.28, 0.44]$, and $[0.44, 0.6]$;
  - for $\mathbf{R}_2 = [0.2, 0.4]$, we have $\mathbf{R}_1 - \mathbf{R}_2 = [0, 0.4]$; this covers $[-0.04, 0.12]$, $[0.12, 0.28]$, and $[0.28, 0.44]$.
- For $\mathbf{R}_1 = [0.6, 0.8]$, we have $\mathbf{R}_1^2 = [0.36, 0.64]$, so three possible $\mathbf{R}_2$: $[0.2, 0.4]$, $[0.4, 0.6]$, and $[0.6, 0.8]$, to the total of $[0.2, 0.8]$. Here, $[0.6, 0.8] - [0.2, 0.8] = [-0.2, 0.6]$, so all 5 subintervals are affected.
- Finally, for $\mathbf{R}_1 = [0.8, 1.0]$, we have $\mathbf{R}_1^2 = [0.64, 1.0]$, so two possible $\mathbf{R}_2$: $[0.6, 0.8]$ and $[0.8, 1.0]$, to the total of $[0.6, 1.0]$. Here, $[0.8, 1.0] - [0.6, 1.0] = [-0.2, 0.4]$, so the first 4 subintervals are affected.



Limitations of this approach. The main limitation of this approach is that when we need an accuracy ε, we must use $\sim 1/\varepsilon$ granules; so, if we want to compute the result with k digits of accuracy, i.e., with accuracy $\varepsilon = 10^{-k}$, we must consider exponentially many boxes ($\sim 10^k$). In plain words, this method is only applicable when we want to know the desired quantity with a given accuracy (e.g., 10%).

Cases when this approach is applicable. In practice, there are many problems when it is sufficient to compute a quantity with a given accuracy: e.g., when we detect an outlier, we usually do not need to know the variance with a high accuracy, an accuracy of 10% is more than enough.

Let us describe the case when interval computations do not lead to the exact range, but set computations do – of course, the range is "exact" modulo accuracy of the actual computer implementations of these sets.

Example: estimating variance under interval uncertainty. Suppose that we know the intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of possible values of $x_1, \ldots, x_n$, and we need to compute the range of the variance $V = \frac{1}{n} \cdot M - \frac{1}{n^2} \cdot E^2$, where $M \stackrel{\text{def}}{=} \sum_{i=1}^{n} x_i^2$ and $E \stackrel{\text{def}}{=} \sum_{i=1}^{n} x_i$.

This problem is important, e.g., in detecting outliers. Outliers are useful in many application areas. For example, in medicine, to detect possible illnesses, we analyze the healthy population, compute the averages $E[x]$ and the standard deviations $\sigma[x]$ of different characteristics x, and if for some person, the value of a blood pressure, weight, body temperature, etc., is outside the corresponding 2- or 3-sigma interval $[E[x] - k_0 \cdot \sigma[x], E[x] + k_0 \cdot \sigma[x]]$, then we perform additional tests to see if there is any hidden problem with this person's health. Similarly, in geophysics, when we look for rare minerals, we know the typical values for a given area, and if at some location, the values of the geophysical characteristics are outliers (i.e., they are outside the corresponding interval), then these area are probably the most promising.

Traditional algorithms for detecting outliers assume that we know the exact values x_i of the corresponding characteristics but in practice, these values often come from estimates or crude measurements. For example, most routine blood pressure measurements performed at health fairs, in drugstores, at the dentist office, etc., are very approximate, with accuracy 10 or more; their objective is not to find the exact values of the corresponding characteristics but to make sure that we do not miss a dangerous anomaly. When we estimate the mean and the standard deviations based on these approximate measurements, we need to take into account that these values are very approximate, i.e., that, in effect, instead of the exact value x_i (such as 110), we only know that the actual (unknown) value of the blood pressure is somewhere within the interval $[\widetilde{x}_i - \Delta_i, \widetilde{x}_i + \Delta_i] = [110 - 10, 110 + 10] = [100, 120]$.

In all these situations, we need to compute the range on the variance V under the interval uncertainty on x_i.

A natural way to to compute V is to compute the intermediate sums $M_k \stackrel{\text{def}}{=} \sum_{i=1}^{k} x_i^2$ and $E_k \stackrel{\text{def}}{=} \sum_{i=1}^{k} x_i$. We start with $M_0 = E_0 = 0$; once we know the pair (M_k, E_k), we compute $(M_{k+1}, E_{k+1}) = (M_k + x_{k+1}^2, E_k + x_{k+1})$. Since the values of M_k and E_k only depend on $x_1, \ldots, x_k$ and do not depend on x_{k+1}, we can conclude that if (M_k, E_k) is a possible value of the pair and x_{k+1} is a possible value of this variable, then $(M_k + x_{k+1}^2, E_k + x_{k+1})$ is a possible value of (M_{k+1}, E_{k+1}). So, the set $\mathbf{p}_0$ of possible values of (M_0, E_0) is the single point $(0, 0)$; once we know the set $\mathbf{p}_k$ of possible values of (M_k, E_k), we can compute $\mathbf{p}_{k+1}$ as

$$\{(M_k + x^2, E_k + x) \,|\, (M_k, E_k) \in \mathbf{p}_k, x \in \mathbf{x}_{k+1}\}.$$

For $k = n$, we will get the set $\mathbf{p}_n$ of possible values of (M, E); based on this set, we can then find the exact range of the variance $V = \frac{1}{n} \cdot M - \frac{1}{n^2} \cdot E^2$.

What C should we choose to get the results with an accuracy $\varepsilon \cdot \overline{V}$? On each step, we add the uncertainty of $1/C$; to, after n steps, we add the inaccuracy of n/C. Thus, to get the accuracy $n/C \approx \varepsilon$, we must choose $C = n/\varepsilon$.

What is the running time of the resulting algorithm? We have n steps; on each step, we need to analyze C^3 combinations of subintervals for E_k, M_k, and x_{k+1}. Thus, overall, we need $n \cdot C^3$ steps, i.e., n^4/ε^3 steps. For fixed accuracy $C \sim n$, so we need $O(n^4)$ steps – a polynomial time, and for $\varepsilon = 1/10$, the coefficient at n^4 is still 10^3 – quite feasible.

For example, for $n = 10$ values and for the desired accuracy $\varepsilon = 0.1$, we need $10^3 \cdot n^4 \approx 10^7$ computational steps – "nothing" for a Gigaherz (10^9 operations per second) processor on a usual PC. For $n = 100$ values and the same desired accuracy, we need $10^4 \cdot n^4 \approx 10^{12}$ computational steps, i.e., 10^3 seconds (15 minutes) on a Gigaherz processor. For $n = 1000$, we need 10^{15} steps, i.e., 10^6 computational steps – 12 days on a single processor or a few hours on a multiprocessor machine.

In comparison, the exponential time 2^n needed in the worst case for the exact computation of the variance under interval uncertainty, is doable ($2^{10} \approx 10^3$ step) for $n = 10$, but becomes unrealistically astronomical ($2^{100} \approx 10^{30}$ steps) already for $n = 100$.

Comment. When the accuracy increases $\varepsilon = 10^{-k}$, we get an exponential increase in running time – but this is OK since, as we have mentioned, the problem of computing variance under interval uncertainty is, in general, NP-hard.

Other statistical characteristics. Similar algorithms can be presented for computing many other statistical characteristics. For example, for every integer $d > 2$, the corresponding higher-order central moment $C_d = \dfrac{1}{n} \cdot \sum_{i=1}^{n} (x_i - \overline{x})^d$ is a linear combination of d moments $M^{(j)} \stackrel{\text{def}}{=} \sum_{i=1}^{n} x_i^j$ for $j = 1, \ldots, d$; thus, to find the exact range for C_d, we can keep, for each k, the set of possible values of d-dimensional tuples $(M_k^{(1)}, \ldots, M_k^{(d)})$, where $M_k^{(j)} \stackrel{\text{def}}{=} \sum_{i=1}^{k} x_i^j$. For these computations, we need $n \cdot C^{d+1} \sim n^{d+2}$ steps – still a polynomial time.

Another example is covariance $\text{Cov} = \dfrac{1}{n} \cdot \sum_{i=1}^{n} x_i \cdot y_i - \dfrac{1}{n^2} \cdot \sum_{i=1}^{n} x_i \cdot \sum_{i=1}^{n} y_i$. To compute covariance, we need to keep the values of the triples (Cov_k, X_k, Y_k), where $\text{Cov}_k \stackrel{\text{def}}{=} \sum_{i=1}^{k} x_i \cdot y_i$, $X_k \stackrel{\text{def}}{=} \sum_{i=1}^{k} x_i$, and $Y_k \stackrel{\text{def}}{=} \sum_{i=1}^{k} y_i$. At each step, to compute the range of

$$(\text{Cov}_{k+1}, X_{k+1}, Y_{k+1}) = (\text{Cov}_k + x_{k+1} \cdot y_{k+1}, X_k + x_{k+1}, Y_k + y_{k+1}),$$

we must consider all possible combinations of subintervals for Cov_k, X_k, Y_k, x_{k+1}, and y_{k+1} – to the total of C^5. Thus, we can compute covariance in time $n \cdot C^5 \sim n^6$.

Similarly, to compute correlation $\rho = \mathrm{Cov}/\sqrt{V_x \cdot V_y}$, we can update, for each k, the values of $(C_k, X_k, Y_k, X_k^{(2)}, Y_k^{(2)})$, where $X_k^{(2)} = \sum\limits_{i=1}^{k} x_i^2$ and $Y_k^{(2)} = \sum\limits_{i=1}^{k} y_i^2$ are needed to compute the variances V_x and V_y. These computations require time $n \cdot C^7 \sim n^8$.

Systems of ordinary differential equations (ODEs) under interval uncertainty. A general system of ODEs has the form $\dot{x}_i = f_i(x_1, \ldots, x_m, t)$, $1 \le i \le m$. Interval uncertainty usually means that the exact functions f_i are unknown, we only know the expressions of f_i in terms of parameters, and we have interval bounds on these parameters.

There are two types of interval uncertainty: we may have global parameters whose values are the same for all moments t, and we may have noise-like parameters whose values may different at different moments of time – but always within given intervals. In general, we have a system of the type

$$\dot{x}_i = f_i(x_1, \ldots, x_m, t, a_1, \ldots, a_k, b_1(t), \ldots, b_l(t)),$$

where f_i is a known function, and we know the intervals $\mathbf{a}_j$ and $\mathbf{b}_j(t)$ of possible values of a_i and $b_j(t)$.

Example. For example, the case of a differential inequality when we only know the bounds $\underline{f}_i(x_1, \ldots, x_n, t)$ and $\overline{f}_i(x_1, \ldots, x_n, t)$ on $f_i(x_1, \ldots, x_n, t)$ can be described as

$$\widetilde{f}_i(x_1, \ldots, x_n, t) + b_1(t) \cdot \Delta(x_1, \ldots, x_n, t),$$

where $\widetilde{f}_i(x_1, \ldots, x_n, t) \stackrel{\mathrm{def}}{=} (\underline{f}_i(x_1, \ldots, x_n, t) + \overline{f}_i(x_1, \ldots, x_n, t))/2$,

$$\Delta(x_1, \ldots, x_n, t) \stackrel{\mathrm{def}}{=} (\overline{f}_i(x_1, \ldots, x_n, t) - \underline{f}_i(x_1, \ldots, x_n, t))/2,$$

and $\mathbf{b}_1(t) = [-1, 1]$.

Solving systems of ordinary differential equations (ODEs) under interval uncertainty. For the general system of ODEs, Euler's equations take the form

$$x_i(t + \Delta t) = x_i(t) + \Delta t \cdot f_i(x_1(t), \ldots, x_m(t), t, a_1, \ldots, a_k, b_1(t), \ldots, b_l(t)).$$

Thus, if for every t, we keep the set of all possible values of a tuple

$$(x_1(t), \ldots, x_m(t), a_1, \ldots, a_k),$$

then we can use the Euler's equations to get the exact set of possible values of this tuple at the next moment of time.

The reason for exactness is that the values $x_i(t)$ depend only on the previous values $b_j(t - \Delta t)$, $b_j(t - 2\Delta t)$, etc., and not on the current values $b_j(t)$.

To predict the values $x_i(T)$ at a moment T, we need $n = T/\Delta t$ iterations.

To update the values, we need to consider all possible combinations of $m+k+l$ variables $x_1(t), \ldots, x_m(t), a_1, \ldots, a_k, b_1(t), \ldots, b_l(t)$; so, to predict the values at moment $T = n \cdot \Delta t$ in the future for a given accuracy $\varepsilon > 0$, we need the running time $n \cdot C^{m+k+l} \sim n^{k+l+m+1}$. This is is still polynomial in n.

Additional advantage of our technique: possibility to take constraints into account. Traditional formulations of the interval computation problems assume that we can have arbitrary tuples $(x_1, \ldots, x_n)$ as long as $x_i \in \mathbf{x}_i$ for all i. In practice, we may have additional constraints on x_i. For example, we may know that x_i are observations of a smoothly changing signal at consequent moments of time; in this case, we know that $|x_i - x_{i+1}| \le \varepsilon$ for some small known $\varepsilon > 0$. Such constraints are easy to take into account in our approach.

For example, if know that $\mathbf{x}_i = [-1, 1]$ for all i and we want to estimate the value of a high-frequency Fourier coefficient $f = x_1 - x_2 + x_3 - x_4 + \ldots - x_{2n}$, then usual interval computations lead to an enclosure $[-2n, 2n]$, while, for small ε, the actual range for the sum $(x_1 - x_2) + (x_3 - x_4) + \ldots$ where each of n differences is bounded by ε, is much narrower: $[-n \cdot \varepsilon, n \cdot \varepsilon]$ (and for $x_i = i \cdot \varepsilon$, these bounds are actually attained).

Computation of f means computing the values $f_k = x_1 - x_2 + \ldots + (-1)^{k+1} \cdot x_k$ for $k = 1, \ldots$ At each stage, we keep the set $\mathbf{s}_k$ of possible values of (f_k, x_k), and use this set to find

$$\mathbf{s}_{k+1} = \{(f_k + (-1)^k \cdot x_{k+1}, x_{k+1}) \,|\, (f_k, x_k) \in \mathbf{s}_k \,\&\, |x_k - x_{k+1}| \le \varepsilon\}.$$

In this approach, when computing f_{2k}, we take into account that the value x_{2k} must be ε-close to the value x_k and thus, that we only add $\le \varepsilon$. Thus, our approach leads to almost exact bounds – modulo implementation accuracy $1/C$.

In this simplified example, the problem is linear, so we could use linear programming to get the exact range, but set computations work for similar nonlinear problems as well.

Classes of probability distributions and p-boxes: a reminder. Often, in addition to the interval $\mathbf{x}_i$ of possible values of the inputs x_i, we also have partial information about the probabilities of different values $x_i \in \mathbf{x}_i$. An exact probability distribution can be described, e.g., by its cumulative distribution function $F_i(z) = \mathrm{Prob}(x_i \le z)$. In these terms, a partial information means that instead of a single cdf, we have a *class* $\mathcal{F}$ of possible cdfs.

A practically important particular case of this partial information is when, for each z, instead of the exact value $F(z)$, we know an interval $\mathbf{F}(z) = [\underline{F}(z), \overline{F}(z)]$ of possible values of $F(z)$; such an "interval-valued" cdf is called a *probability box*, or a *p-box*, for short; see, e.g., [2].

Propagating p-box uncertainty via computations: a problem. Once we know the classes $\mathcal{F}_i$ of possible distributions for x_i, and a data processing algorithms $f(x_1, \ldots, x_n)$, we would like to know the class $\mathcal{F}$ of possible resulting distributions for $y = f(x_1, \ldots, x_n)$.

Idea. For problems like systems of ODES, it is sufficient to keep, and update, for all t, the set of possible joint distributions for the tuple $(x_1(t), \ldots, a_1, \ldots)$.

From idea to computer implementation. We would like to estimate the values with some accuracy $\varepsilon \sim 1/C$ and the probabilities with the similar accuracy $1/C$. To describe a distribution with this uncertainty, we divide both the x-range and the probability (p-) range into C granules, and then describe, for each x-granule, which p-granules are covered. Thus, we enclose this set into a finite union of p-boxes which assign, to each of x-granules, a finite union of p-granule intervals.

A general class of distributions can be enclosed in the union of such p-boxes. There are finitely many such assignments, so, for a fixed C, we get a finite number of possible elements in the enclosure.

We know how to propagate uncertainty via simple operations with a finite amount of p-boxes (see, e.g., [2]), so for ODEs we get a polynomial-time algorithm for computing the resulting p-box for y.

For p-boxes, we need further improvements to make this method practical. Formally, the above method is polynomial-time. However, it is not yet practical beyond very small values of C. Indeed, to describe a p-subbox, we need to attach one of C probability granules to each of C x-granules; these are $\sim C^C$ such attachments, so we need $\sim C^C$ subboxes. For $C = 10$, we already get an unrealistic 10^{10} increase in computation time. (In contrast, for interval computations, we need a feasible number $C = 10^2$ of subboxes.)

Acknowledgments. This work was supported in part by the National Science Foundation grant HRD-0734825, by Grant 1 T36 GM078000-01 from the National Institutes of Health, and by Grant MSM 6198898701 from MŠMT of Czech Republic.

References

1. M. Ceberio, S. Ferson, V. Kreinovich, et al., "How To Take Into Account Dependence Between the Inputs: From Interval Computations to Constraint-Related Set Computations", *Proc. 2nd Int'l Workshop on Reliable Engineering Computing*, Savannah, Georgia, February 22–24, 2006, pp. 127–154; final version in *Journal of Uncertain Systems*, 2007, Vol. 1, No. 1, pp. 11–34.
2. S. Ferson. *RAMAS Risk Calc 4.0*. CRC Press, Boca Raton, Florida, 2002.
3. S. Ferson, L. Ginzburg, V. Kreinovich, L. Longpré, and M. Aviles, "Computing Variance for Interval Data is NP-Hard", *ACM SIGACT News*, 2002, Vol. 33, No. 2, pp. 108–118.
4. L. Jaulin et al., *Applied Interval Analysis*, Springer, London, 2001.
5. V. Kreinovich, A. Lakeyev, J. Rohn, and P. Kahl, *Computational complexity and feasibility of data processing and interval computations*, Kluwer, Dordrecht, 1997.
6. S. P. Shary, "Solving tied interval linear systems", *Siberian Journal of Numerical Mathematics*, 2004, Vol. 7, No. 4, pp. 363–376 (in Russian).

Semilattices, Domains, and Computability (Invited Talk)

Dana Scott

Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA

As everyone knows, one popular notion of Scott domain is defined as a bounded complete algebraic cpo. These are closely related to algebraic lattices: (i) A Scott domain becomes an algebraic lattice with the adjunction of an (isolated) top element. (ii) Every non-empty Scott-closed subset of an algebraic lattice is a Scott domain. Moreover, the isolated (= compact) elements of an algebraic lattice form a semilattice (under join). This semilattice has a zero element, and, provided the top element is isolated, it also has a unit element. The algebraic lattice itself may be regarded as the ideal completion of the semilattice of isolated elements. This is all well known. What is not so clear that is that there is an easy-to-construct domain of countable semilattices giving isomorphic copies of all countably based domains. This approach seems to have advantages over both "information systems" or more abstract lattice formulations, and it makes definitions of solutions to domain equations very elementary to justify. The "domain of domains" also has an immediate computable structure.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 17
http://drops.dagstuhl.de/opus/volltexte/2009/2252

Computable Analysis of Differential Equations (Invited Talk)

Ning Zhong

Department of Mathematical Sciences, University of Cincinnati, Cincinnati, OH, USA

In this talk, we discuss some algorithmic aspects of the local and global existence theory for various ordinary and partial differential equations. We will present a sample of results and give some idea of the motivation and general philosophy underlying these results.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 19
http://drops.dagstuhl.de/opus/volltexte/2009/2253

Theory and Practice of Higher-type Computation (Tutorial)

Martín Escardó

School of Computer Science, University of Birmingham, Birmingham B15 2TT, UK

In higher-type computation, established by Kleene and Kreisel in the late 1950's (independently), one works with the data types obtained from the discrete natural numbers by closing under finite products and function spaces. For the theory of higher-type programming languages, it is natural to work with a corresponding hierarchy, or type structure, of domains, identified by Ershov and Scott in the late 1960's (again independently). The Kleene–Kreisel and Ershov–Scott hierarchies account for total and partial computation respectively.

In this tutorial I'll explain the theory and practice of higher-type computation and programming languages, and develop old and new applications.

From a theoretical point of view, I'll present Kleene–Kreisel spaces and Ershov–Scott domains, and relate the two. Moreover, I'll discuss common generalizations, chiefly QCB spaces and equilogical spaces, which admit further useful closure properties, and their relationship to TTE (Schröder, Simpson. Scott, Bauer, Weihrauch and many others). I'll also present a natural higher-type model of computation/programming language, namely PCF (Platek, Scott, Plotkin).

From a practical point of view, I'll introduce a fragment of the language Haskell as a faithful implementation of PCF. Moreover, I'll develop and run several examples (and prove theorems about them), pertaining to (i) exhaustive search of infinite sets in finite time (in particular Ulrich Berger's algorithm and generalizations), and (ii) computation with real numbers (in particular Alex Simpson's integration algorithm and generalizations).

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 21
http://drops.dagstuhl.de/opus/volltexte/2009/2254

Computer Verified Exact Analysis (Tutorial)

Bas Spitters and Russell O'Connor

Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands

This tutorial will illustrate how to use the Coq proof assistant to implement effective and provably correct computation for analysis. Coq provides a dependently typed functional programming language that allows users to specify both programs and formal proofs.

We will introduce dependent type theory and show how it can be used to develop both mathematics and programming. We will show how to use dependent type theory to implement constructive analysis. Specifically we will cover how to implement effective real numbers and effective integration.

This work will be done using the Coq proof assistant. The tutorial will cover how to use the Coq proof assistant. Attendees are encouraged to download and install Coq 8.2 from `http://coq.inria.fr/download` and also download and make the full system of C-CoRN from `http://c-corn.cs.ru.nl/download.html` beforehand.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 23
http://drops.dagstuhl.de/opus/volltexte/2009/2255

Computing Conformal Maps onto Canonical Slit Domains

Valentin V. Andreev[1] and Timothy H. McNicholl[2]

[1] Department of Mathematics
Lamar University
Beaumont, Texas 77710
andreev@math.lamar.edu

[2] Department of Mathematics
Lamar University
Beaumont, Texas 77710
timothy.h.mcnicholl@gmail.com

Abstract. We extend the results of [2] by computing conformal maps onto the canonical slit domains in Nehari [14]. Along the way, we demonstrate the computability of solutions to Neuman problems.

1 Introduction

Let $\hat{\mathbb{C}}$ denote the extended complex plane. A *domain* is an open connected subset of $\hat{\mathbb{C}}$. A domain is *degenerate* if a component of its complement consists of a single point. A domain is *n-connected* if its complement has precisely n connected components and *finitely connected* if it is n-connected for some n.

In studying conformal mappings between domains in the extended complex plane it is convenient for both theoretical and practical purposes to introduce the so-called *canonical* domains and to study conformal maps of arbitrary domains onto these canonical domains. If the domain is 1-connected and non-degenerate, the canonical domain is the unit disk. In the case of doubly connected non-degenerate domains, the canonical domain is the annulus $\{z \in \mathbb{C} : r_1 < |z| < r_2\}$. The *modulus* of this annulus is r_2/r_1. It is well-known that annuli with different moduli are not conformally equivalent (see, *e.g.*, [14] p. 333). When considering conformal mappings of domains with connectivity $n \geq 3$, it is convenient to consider canonical domains with different geometric characteristics.

Paul Koebe [11] outlined an iteration method for finding the conformal mapping from an n-connected domain to a circular domain (a domain whose complement consists of n disjoint closed disks). The convergence of his method was proved by Gaier [6], and the computability by Andreev, Daniel, and McNicholl [2]. The circular domains are the canonical domains in the recent constructions of the Schwarz–Christoffel mappings for domains that are sufficiently separated (see [3] and [5] and the refernces therein) and have been used as canonical domains in aircraft engineering as early as 1928 [1] and later by Halsey [8]. For

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 25-36
http://drops.dagstuhl.de/opus/volltexte/2009/2256

numerous applications to nonlinear problems in mechanics see the monograph [13]. An appealing property of the circular domains as canonical domains is that recently there have been found explicit formulas for the Green's function (and, hence, for the Bergman kernel) for circular domains [10] and for the modified Green's function [4], which then is used to derive explicit formulas for conformal maps of circular domains onto the canonical slit domains. The formula in the latter paper contains infinite products which converge for domains that are sufficiently separated. However, it is not known if they always converge.

Paul Koebe [12] introduced several of the canonical slit domains. There have been demonstrated deep connections between the Dirichlet and Neumann problems in multiply connected domains and conformal slit mappings, potential theory and extremal problems [14], [15], [16].

We define here the canonical slit domains presented in Nehari's book [14].

The slit disk domain

Let $\mathbb{D}$ denote the unit disk centered at the origin. These domains are obtained by removing finitely many arcs from $\mathbb{D}$. Each of these arcs must be an arc of a circle centered at the origin.

The slit annulus

These domains are obtained by removing finitely many arcs from an annulus whose outer circle is $\partial\mathbb{D}$. Again, each of these arcs must be an arc of a circle centered at the origin.

The circular slit domain

These domains are obtained by removing from $\hat{\mathbb{C}}$ one or more arcs. Again, each of these arcs must be an arc of a circle centered at the origin.

The radial slit domain

These domains are obtained by removing from $\hat{\mathbb{C}}$ one or more line segments which do not pass through the origin. Each of these line segments, when extended indefinitely in both directions, must yield a line that passes through the origin.

The parallel slit domain

These domains are obtained by removing from $\hat{\mathbb{C}}$ one or more parallel line segments.

We will first show that one can compute the conformal mappings onto a slit disk domain using a result of Max Schiffer [15]. We will then use the relations described in Nehari [14] between the conformal maps onto these domains to

compute the conformal mappings onto the slit annulus and circular slit domains. We use the constructions in Schiffer's monograph to compute the maps onto the radial slit and parallel slit domain.

We will use Type-Two Effectivity [17] as our model of computation over spaces whose cardinality is that of the reals. The naming systems we will use are described in Section 3.1 of [2]. Since these are the only naming systems we will use, we will suppress their mention. We will also talk about computations on objects when it is clear that we are really talking about computations with names of objects. We will write our proofs in a fairly informal style. In particular, we will rely on the informal definitions in Section 3.1 or [2].

2 Background from complex and harmonic analysis

Let D be a Jordan domain with boundary curves $\Gamma_1, \ldots, \Gamma_n$. For each $z \in D$, define $\omega(z, \Gamma_j, D)$ to be the value at z of the solution to the Dirichlet problem with boundary data

$$f(\zeta) = \begin{cases} 1 & \zeta \in \Gamma_j \\ 0 \text{ otherwise} \end{cases}$$

The function ω is called *harmonic measure*.

The *normal derivative* of u is denoted $\frac{\partial u}{\partial n}$ and is defined to be

$$\left(\frac{\partial u}{\partial x} y'(t) - \frac{\partial u}{\partial y} x'(t)\right) \frac{1}{|x'(t) + iy'(t)|}$$

when (x, y) is a positively oriented smooth Jordan curve. In this case, we also define

$$\frac{\partial u}{\partial s} = \left(\frac{\partial u}{\partial x} x'(t) + \frac{\partial u}{\partial y} y'(t)\right) \frac{1}{|x'(t) + iy'(t)|}.$$

If v is a harmonic conjugate of u, then it follows from the Cauchy-Riemann equations that

$$\frac{\partial u}{\partial n} = \frac{\partial v}{\partial s}, \text{ and}$$
$$\frac{\partial v}{\partial n} = -\frac{\partial u}{\partial s}.$$

If u is harmonic, and if γ is a boundary component of $\mathrm{dom}(u)$, then the *period of the conjugate of u about γ* is defined to be

$$\frac{1}{2\pi} \int_\gamma \frac{\partial u}{\partial n} |dz|$$

where $|dz| =_{df} |z'(t)|dt$ is the differential of arc length. To make sense of this integral, we first use Schwarz Reflection to extend the domain of u to an open set containing γ.

Let G_D denote the Green's function of domain D. The following well-known result will be useful and is Corollary II.2.6 of [7].

Proposition 1. *Suppose D is a Jordan domain with smooth boundary curves $\Gamma_1, \ldots, \Gamma_n$. Then,*

$$\omega(\zeta, \Gamma_j, D) = -\frac{1}{2\pi} \int_{\Gamma_j} \frac{\partial G_D(z, \zeta)}{\partial n_z} |dz|.$$

Suppose f is a conformal map of a domain D onto a domain D_1, and that γ, γ_1 are boundary components of D, D_1 respectively. Suppose that whenever $\{z_n\}_{n=0}^{\infty}$ is a sequence of points in D such that $\lim_{n\to\infty} d(z_n, \gamma) = 0$, $\lim_{n\to\infty} d(f(z_n), \gamma_1) = 0$. We say that *$f$ maps γ to γ_1*.

We will follow the convention of identifying a curve with its parameterizations.

3 A summary of previous results

The following three results, which are Theorems 5.2, 5.5, and 6.2 of [2], form the cornerstone for our work. Intuitively, the first of these theorems states that differentiation of harmonic functions is a computable operation.

Theorem 1 (Computable differentiation of harmonic functions). *From a name of a harmonic function, u, we may compute a name of $u'|_{\mathbb{C}}$.*

The second of these results states, roughly speaking, that solving Dirichlet problems (*i.e.* finding a harmonic function on a Jordan domain from the knowledge of its values on the boundary of the domain) is a computable operation.

Theorem 2 (Computable Solution of Dirichlet Problems). *Given a name of a Jordan domain D and names of smooth $\gamma_1, \ldots, \gamma_n$ and their derivatives, if $\gamma_1, \ldots, \gamma_n$ are the distinct boundary components of D, and if we are also given a name of a continuous $f : \partial D \to \mathbb{R}$, then we can compute a solution of the corresponding Dirichlet problem. Furthermore, we can compute an extension of this solution to $\overline{D}$.*

The third of these results demonstrates the computability of a matrix (known as the *Riemann matrix*) whose components are the periods around the boundary components of the harmonic conjugates of the harmonic measure functions.

Theorem 3 (Computability of the Riemann Matrix). *Given the same initial data as in Theorem 2, we can compute a name of the period of the harmonic conjugate of $\omega(\cdot, \Gamma_i, D)$ around Γ_j.*

4 Single-valued and multi-valued harmonic conjugates

Suppose u is a harmonic function with domain D. If D is 1-connected, then a harmonic conjugate of u may be defined by the equation

$$v(\zeta) = \int_{\zeta_1}^{\zeta} \frac{\partial u}{\partial n} |dz| \tag{1}$$

If D is multiply connected, then the right side of (1) may depend on the path of integration. In this case, u is said to have a *multi-valued harmonic conjugate*. Otherwise, u is said to have a *single-valued harmonic conjugate*. It follows that if D is contained in the interior of one of its boundary components then u has a single-valued harmonic conjugate if and only if its period around every other boundary component is zero.

It is well-known that if D is finitely connected and bounded by smooth Jordan curves, one of which contains D in its interior, then one can add a unique linear combination of the harmonic measure functions of the boundary components of D to u and obtain a function with a single-valued harmonic conjugate. Our first goal is to show that this can be done effectively.

Lemma 1. *Given a name of a harmonic function u defined on a finitely connected domain, D, and names of the boundary components of D, $\gamma_1, \ldots, \gamma_n$, we may compute $b_1, \ldots, b_{n-1}$ such that*

$$u + \sum_{j=1}^{n-1} b_j \omega(\cdot, \gamma_j, D) \tag{2}$$

has a single-valued harmonic conjugate provided $\gamma_1, \ldots, \gamma_n$ are smooth Jordan curves, D is contained in the interior of γ_n, and we are also given names of $\gamma_1', \ldots, \gamma_n'$.

Proof. Let $R_{k,j}$ be the period of $\omega(\cdot, \gamma_j, D)$ about γ_k.

We first want to compute the period of the conjugate of u about each γ_k. Denote this period by p_k. To compute p_k, we want γ_k to be positively oriented. This can be checked by using the *winding number*

$$\int_{\gamma_k} \frac{1}{z - \zeta}.$$

We can effectively search for a rational rectangle R such that $\overline{R} \subseteq \mathbb{C} - \gamma_k$ on which this winding number is non-zero. If this value is positive, we can in addition discover a positive lower bound on it. If it is negative, then we can in addition discover a negative upper bound on it. In the former case, we know γ_k is positively oriented. Otherwise, it is negatively oriented in which case we can reparameterize it positively. Hence, we will assume without loss of generality that each γ_k is positively oriented.

Now, let $R_{k,j}$ be the period of $\omega(\cdot, \gamma_j, D)$ about γ_k. It is well-known that the matrix $(R_{k,j})_{k,j=1,\ldots,n-1}$ is invertible. (See, *e.g.*, Section I.10 of [14].) To ensure that the function in (2) has no conjugate period about γ_k, $k = 1, \ldots, n-1$, it suffices to show that

$$R_{k,1} b_1 + \ldots + R_{k,n-1} b_{n-1} = -p_k.$$

It now follows from the results in [18] that $b_1, \ldots, b_{n-1}$ can be computed from the given information.

5 The slit disk domain

The following lemma will be useful.

Lemma 2 (Conformal Reconfiguring Lemma). *Given a name of a non-degenerate domain D, a name of its boundary, and the number of its boundary components, we can compute a domain D_1, its boundary, a conformal map f of D onto D_1, and smooth Jordan curves $\gamma_1, \ldots, \gamma_n$ and their derivatives such that $\gamma_1, \ldots, \gamma_n$ are the boundary components of D_1. Furthermore, γ_n is a circle. Furthermore, if we are in addition given a name of a boundary component γ of D we can ensure that f maps γ to γ_n.*

Proof. Follow the first n steps of the Koebe Construction (for details see Section 2 and Theorem 4.6 of [2]): let $D_{0,1}, ..., D_{0,n}$ denote the connected components of the complement of D. At the first step with the help of a Riemann mapping, map the complement of $D_{0,1}$ conformally onto the unit disk $D'_{0,1}$. It follows from Theorem 5.1 of [9] that we can compute this map from the given data. The boundary δ_1 of $D_{0,1}$ is transformed into the unit circle δ'_1. $D_{0,2}$ is mapped onto $D'_{0,2}$, δ_2 into δ'_2 *etc.*. At step two map the complement of $D'_{0,2}$ onto the unit disk using a Riemann mapping. The image of δ'_1 under the second Riemann mapping is an analytic curve. After n analogous steps the images $\gamma_1, ..., \gamma_n$ of $\delta_1, ..., \delta_n$ are analytic curves and γ_n is a circle. The

Theorem 4. *Given a name of a finitely connected, non-degenerate domain D, a name of its boundary, a name of one of its boundary components, γ, a name of a point $\zeta_0 \in \mathbb{D}$, and the number of boundary components of D, we can compute a conformal mapping of D onto a slit disk domain that maps ζ_0 to 0 and γ to $\partial\mathbb{D}$.*

Proof. We first apply the Conformal Reconfiguring Lemma. Let $f, D_1, \gamma_1, \ldots, \gamma_n$ be thusly obtained. We may assume f maps γ onto γ_n. Let $\zeta_1 = f(\zeta_0)$. We can now compute the center and radius of γ_n. Label these ξ and R respectively. Let $D_2, \Gamma_1, \ldots, \Gamma_n, \zeta_2$ be the images of $D_1, \gamma_1, \ldots, \gamma_n, \zeta_1$ under the inversion map $z \mapsto \frac{R^2}{z-\xi}$.

Let G be the Green's function of D_2. It follows from Theorem 2 that we can compute G from the given information.

Let $\omega_j(z) = \omega(z, \Gamma_j, D_2)$. Compute $b_1(\zeta), \ldots, b_{n-1}(\zeta)$ as in the proof of Lemma 1 for the function $G(\cdot, \zeta)$. Let

$$m(z, \zeta) = G(z, \zeta) + \sum_{j=1}^{n-1} b_j(\zeta)\omega_j(\zeta).$$

It follows that m has no conjugate period about any of $\gamma_1, \ldots, \gamma_{n-1}$. A fairly straightforward calculation shows that m has a period of 1 about ζ. So, for all $z_0 \in D_2 - \{\zeta_2\}$, let

$$g(z_0) = \exp\left(-m(z, \zeta_2) - i\int_{\zeta_2}^{z_0} \frac{\partial m(z, \zeta_2)}{\partial n_z}|dz|\right).$$

Extend g to all of D_2 by setting $g(\zeta_2) = 0$. It follows that g is single-valued and analytic. Note that $g(\zeta_2) = 0$.

A fairly straightforward calculation shows that g is the function in (A1.21) of [15]. Hence, g is the conformal mapping of D_2 onto a slit disk domain Ω that maps ζ_2 to 0 and Γ_n to $\partial\mathbb{D}$.

It now only remains to show that we can compute a name of g from the given data. It suffices to show that from the given data and a name of a point $z \in D_2$ we can compute a name of $g(z)$ (see *e.g.* Theorem 3.3.15.2 of [17]).

When $z \neq \zeta_2$ (which, if true, will eventually be witnessed as we read the name of z), we can through an effective search procedure discover a piecewise linear path of integration contained in $D_2 - \{\zeta_2\}$. If z is in a subbbasic neighborhood of ζ_2 whose closure is contained in D_2, we can compute a positive lower bound on $G(z, \zeta_2)$ and arrive at a subbasic neighborhood of 0 which will contain $g(z)$. In either case, the computed neighborhoods will converge to $g(z)$ if the input neighborhoods converge to z.

It is worth noting that in the case when D is a circular domain, one can obtain explicit formulas for the slit-disk mapping function using the formulas for the Green's function and harmonic measure in [10].

6 Some immediate consequences of the slit disk result

Let 'SD' stand for 'slit disk', 'CS' for 'circular slit', *etc.*. We introduce some notation for the conformal maps onto these domains. Fix a non-degenerate, finitely connected domain D. Let $\zeta_0, \zeta_1 \in D$, and let $\gamma_1, \ldots, \gamma_n$ be the boundary components of D. We then let $f_{SD}(\cdot; D, \zeta_0, \gamma_j)$ denote the unique conformal map of D onto a slit disk domain that maps ζ_0 to 0 and γ_j onto $\partial\mathbb{D}$ whose derivative at ζ_0 is positive.

Let $f_{CS}(\cdot; D, \zeta_0, \zeta_1)$ be the conformal map of D onto a circular slit domain that maps ζ_0 to 0, ζ_1 to ∞, and whose residue at ζ_1 is 1.

Let $f_{PS}(\cdot; D, \zeta_1, \theta)$ be the conformal map of D onto a parallel slit domain where all slits have angle θ with the x-axis and whose Laurent expansion at ζ_1 is of the form

$$\frac{1}{z - \zeta_1} + a(z - \zeta_1) + b(z - \zeta_1)^2 + \ldots.$$

Let $f_{RS}(\cdot; D, \zeta_0, \zeta_1)$ be the conformal map of D onto a radial slit domain that maps ζ_0 to 0, ζ_1 to ∞, and whose residue at ζ_1 is 1.

Let $f_{SA}(\cdot; D, \gamma_j, \gamma_k)$ be the conformal map of D onto a slit annulus domain that maps γ_j onto $\partial\mathbb{D}$ and γ_k onto the inner circle.

We omit any of these parameters when they are made clear by context.

Suppose we are given a name of a finitely connected, non-degenerate domain D, a name of its boundary, and the number of its boundary components. It is now required to show that we can compute these other canonical maps uniformly in the parameters beyond the semicolon. In the case of f_{CS} and f_{SA}, this follows

from the following identities which are proven in Section VII.1 of [14].

$$f_{CS}(z;\zeta_0,\zeta_1) = \frac{f'_{SD}(\zeta_1;\zeta_1)}{f'_{SD}(\zeta_1;\zeta_0)} \frac{f_{SD}(z;\zeta_0)}{f_{SD}(z;\zeta_1)}$$
$$f_{SA}(z;\gamma_j,\gamma_k) = \frac{f_{SD}(z;\zeta_0,\gamma_j)}{f_{SD}(z;\zeta_0,\gamma_k)}$$

We now discuss the computation of $f_{PS}(\cdot;D,\zeta,\theta)$. Let ζ_x,ζ_y denote the real and imaginary parts of ζ respectively. It is shown on page 256 of [15] that

$$f_{PS}(z;D,\zeta,\pi/2) = -\frac{\partial}{\partial\zeta_x}\log f_{SD}(z;D,\zeta,\gamma_j)$$
$$f_{PS}(z;D,\zeta,0) = -\frac{1}{i}\frac{\partial}{\partial\zeta_y}\log f_{SD}(z;D,\zeta,\gamma_j)$$

It then follows (as on page 257 of [15]) that

$$f_{PS}(z;D,\zeta,\theta) = e^{i\theta}[\cos(\theta)f_{PS}(z;D,\zeta,0) - i\sin(\theta)f_{PS}(z;D,\zeta,\pi/2)].$$

Hence, we may compute $f_{PS}(\cdot;D,\zeta,\theta)$ from the given data.

In order to compute the conformal mappings onto the other canonical domains, we make a digression and consider the Neuman problem.

7 Digression: computing solutions to the Neuman problem

Let D be a bounded domain with smooth boundary curves $\Gamma_1,\ldots,\Gamma_n$. Let $f \in C(\partial D)$, and suppose $\int_{\partial D} f|dz| = 0$. The resulting *Neuman problem* is to find a harmonic function u on D such that

$$\frac{\partial u}{\partial n} = f \text{ on } \partial D \tag{3}$$
$$\int_{\partial D} u|dz| = 0 \tag{4}$$

Such solutions exist (see, *e.g.* Appendix B of [7]). The condition 4 ensures they are unique. Their computability will now be demonstrated by a well-known procedure (see *e.g.* proof of Theorem B.1 in [7]).

Theorem 5 (Computing solutions of Neuman problems). *Given a name of a bounded domain D, names of n smooth Jordan curves $\Gamma_1,\ldots,\Gamma_n$ which form its boundary components as well as names of their derivatives, and a name of $f \in C(\partial D)$ such that (3) holds, one can compute a name of the solution of the resulting Neuman problem.*

Proof. There is already a well-known 'procedure' for finding solutions to Neuman problems. The purpose of this proof is to explain this procedure and show that it can be implemented on a digital computing device.

By using the winding number and a simple search procedure, we can determine which of $\Gamma_1, \ldots, \Gamma_n$ contains D in its interior. Without loss of generality, suppose Γ_n is this curve.

As in the proof of Lemma 1, we can assume $\Gamma_1, \ldots, \Gamma_n$ are positively oriented.

Let $R_{j,k}$ be the period of the conjugate of ω_k about Γ_j. As noted in the proof of Lemma 1, the matrix $(R_{j,k})_{j,k=1,\ldots,n-1}$ is invertible. So, we can now compute the solution to the system of linear equations

$$R_{j,1}b_1 + \ldots + R_{j,n-1}b_{n-1} = \int_{\Gamma_j} f|dz| \; j = 1, \ldots, n-1.$$

Let

$$f_1 = f - \sum_{k=1}^{n-1} b_k \frac{\partial \omega_k}{\partial n}.$$

It follows that $\int_{\Gamma_j} f_1|dz| = 0$ if $j \in \{1, \ldots, n-1\}$.

It is an easy consequence of Green's Theorem that $R_{j,k} = R_{k,j}$. (See, *e.g.*, Section I.10 of [14].) It is also easy to show that for each j, the sum of the periods of the harmonic conjugates of $\omega_1, \ldots, \omega_n$ is 0. (One first notes that the sum of the harmonic measure functions is identically 1 on ∂D.) Since $\int_\Gamma f|dz| = 0$, it now follows by a fairly straightforward calculation that $\int_{\Gamma_n} f_1|dz| = 0$.

We now wish to define a function g on ∂D. We do so by defining it on each boundary component of D. When $\zeta \in \Gamma_j$, we let

$$g(\zeta) = \int_0^{t_0} f_1(\Gamma_j(t))|\Gamma_j'(t)|dt$$

where t_0 is such that $\Gamma_j(t_0) = \zeta$. Since $\int_{\Gamma_j} f_1|dz| = 0$, it follows that the choice of t_0 is irrelevant when $\zeta = \Gamma_j(0)$. Hence, g is well-defined.

It is now necessary to prove the following Lemma.

Lemma 3. *g can be computed from the given data.*

Proof. Let $\omega_j = \omega(\cdot, \Gamma_j, D)$.

Suppose we are given a name of a point $\zeta \in \partial D$ as input. From our name for a parametrization of Γ_j, we can compute names of Γ_j as a closed subset of the plane as well as a name of the open set $\mathbb{C} - \Gamma_j$. (See, *e.g.*, Theorem 6.2.4.4 of [17].) We then scan these names and our name for ζ until we find a rational rectangle R and an index j such that $\zeta \in R$, $R \cap \Gamma_j \neq \emptyset$, and $\overline{R} \cap \Gamma_k = \emptyset$ when $k \neq j$. Hence, we now know $\zeta \in \Gamma_j$. Begin computing a name for the function h defined by

$$h(t) = \int_0^t f_1(\Gamma_j(t))|\Gamma_j'(t)|dt.$$

We now continue scanning our name for ζ and our generated names for $\Gamma_j(0)$ and h. Suppose that at some point in this process we discover disjoint rational rectangles R_1, R_2 such that $\zeta \in R_1$ and $\Gamma_j(0) \in R_2$. So, we now know $\zeta \neq \Gamma_j(0)$. Hence, there is exactly one value of t for which $\Gamma_j(t) = \zeta$ and we can compute this value. (See, *e.g.* Corollary 6.3.5 of [17].) Hence, we can now compute $g(\zeta)$ directly from the definition of g.

Suppose on the other hand that at some point in this process no such rational rectangles have been discovered. We then search the portions of these names read so far for R, $[a_1, b_1], \ldots, [a_m, b_m]$, $R_1, \ldots, R_m$, R', I_1, and I_2 such that

1. $\zeta, \Gamma_j(0) \in R$,
2. Γ_j maps $[a_l, b_l]$ into R_l,
3. $[a, b] =_{df} \bigcup_l [a_l, b_1]$ is a subinterval of $(0, 1)$,
4. $R_l \cap \overline{R} = \emptyset$,
5. h maps $[0, a]$ into I_1, and
6. h maps $[b, 1]$ into I_2.

If this search fails, then we continue scanning. If it succeeds, then, although we do not know yet if $\zeta = \Gamma_j(0)$, we do know that any Γ_j preimage of ζ lies in $[0, a] \cup [b, 1]$. Note that $0 \in I_1 \cap I_2$. So, we can list, for each successful search of this kind, $I_1 \cup I_2$ as an interval that contains $g(\zeta)$. We can also in the future interleave listing of all rational intervals that contain $I_1 \cup I_2$.

We now show that this process generates a name for $g(\zeta)$. Every interval listed contains $g(\zeta)$. So, we only need to show that every interval that contains $g(\zeta)$ is eventually listed. This is clearly true if $\zeta \neq \Gamma_j(0)$. Suppose $\zeta = \Gamma_j(0)$. It follows that there will be infinitely many successful search of the kind described above. It also follows that larger portions of these names are read, the diameter of $I_1 \cup I_2$ will tend to 0. It follows that a name of $0 = g(\Gamma_j(0))$ is written on the output tape.

We now compute the solution to the Dirichlet problem for D with boundary data g. Call this solution v_1. We now compute $a_1, \ldots, a_{n-1}$ such that

$$v =_{df} v_1 + \sum_{j=1}^{n-1} a_j \omega_j$$

has a single-valued harmonic conjugate. Note that since ω_j is constant on each curve of ∂D, $\frac{\partial v}{\partial s} = f_1$. Compute $\xi \in D$. Let:

$$u_1(z_0) = \int_{\xi}^{z_0} \frac{\partial(-v)}{\partial n} |dz|$$

$$u_2 = u_1 + \sum_{j=1}^{n-1} b_j \omega_j$$

Since u_1 is a harmonic conjugate of $-v$, it follows that the normal derivative of u_1 on ∂D is f_1. It now follows that f is the normal derivative of u_2 on ∂D. We

now complete our computation by setting

$$u = u_2 - \int_{\Gamma} u_2 |dz|.$$

If D is a bounded domain bounded by smooth Jordan curves $\Gamma_1, \ldots, \Gamma_n$, then the *Neuman function* of D, N_D, is defined by the following conditions.

1. $z \mapsto N(z,\zeta) + \log|z-\zeta|$ is harmonic.
2. $\frac{\partial}{\partial n_z} N(z,\zeta) = -\frac{2\pi}{L}$ on ∂D where L is the length of ∂D.
3. $\int_{\partial D} N(z,\zeta)|dz| = 0$.

Corollary 1. *From names of D, $\Gamma_1, \ldots, \Gamma_n$, $\Gamma_1', \ldots, \Gamma_n'$ as in Theorem 5, we can compute a name of N_D.*

8 The radial slit domain

We conclude with the following.

Theorem 6. *From a name of non-degnerate, finitely connected, domain D, a name of its boundary, names of distinct $\zeta_0, \zeta_1 \in D$, and the number of its boundary components, we can compute a name of $f_{RS}(\cdot; D, \zeta_0, \zeta_1)$.*

Proof. Let $u(z) = N_D(z,\zeta_0) - N_D(z,\zeta_1)$. It follows that u has a single-valued harmonic conjugate. So, let

$$\tilde{u}(z_0) = \int_{\zeta_0}^{z_0} \frac{\partial u}{\partial n} |dz|$$

when $z_0 \neq \zeta_0, \zeta_1$. Let $f = \exp(-(u+i\tilde{u}))$. Extend f to all of D by setting $f(\zeta_0) = 0$ and $f(\zeta_1) = \infty$. It is shown in [15] (page 265, (A1.62)) that $f_{RS}(\cdot, D, \zeta_0, \zeta_1) = f$. It only remains to demonstrate that we can compute f from the given data.

Suppose we are given the name of $z \in D$ as input. Scan the names of z, ζ_0, ζ_1. If at some point, we discover disjoint rational rectangles R_1, R_2, R_3 such that $z \in R_1$, $\zeta_0 \in R_2$, and $\zeta_1 \in R_3$, then we can compute $u(z)$ and $\tilde{u}(z)$ directly. Suppose at some point we have not found such rectangles. If we have discovered a rational rectangle R that contains z and ζ_0 but not ζ_1, we can compute a positive lower bound on $N_D(z,\zeta_0) - N_D(z,\zeta_1)$ and hence a neighborhood of 0 that contains $f(z)$. If we have discovered a rational rectangle R that contains z and ζ_1 but not ζ_0, we can compute a negative upper bound on $N_D(z,\zeta_0) - N_D(z,\zeta_1)$ and hence a neighborhood of ∞ that contains $f(z)$. By continuing this process indefinitely, we generate a name of $f(z)$.

References

1. N. I. Akhiezer, *Aerodynamical investigations*, vol. 7, Ukrain. Akad. Nauk Trudi Fiz.-Mat. Viddilu, 1928.

2. V.V. Andreev, D. Daniel, and T. H. McNicholl, *Computing conformal maps onto circular domains*, Submitted. Preprint avaliable at third author's webpage at www.math.lamar.edu.
3. D. Crowdy, *Schwarz-christoffel mappings to unbounded multiply connected polygonal regions*, Mathematical Proceedings of the Cambridge Philosophical Society **142** (2007), no. 2, 319–339.
4. D. Crowdy and J. Marshall, *Conformal mappings between canonical multiply connected domains*, Computational Methods and Function Theory **6** (2006), no. 1, 59 – 76.
5. T. K. Delillo, *Schwarz–christoffel mapping of bounded multiply connected domains*, Computational Methods Function Theory **6** (2006), 275–300.
6. D. Gaier, *Untersuchung zur durchführung der konformen abbildung mehrfach zusammenhängender gebiete*, Arch. Rat. Mech. Anal. **3** (1959), 149 – 178.
7. J.B. Garnett and D. E. Marshall, *Harmonic measure*, New Mathematical Monographs, 2, Cambridge University Press, Cambridge, 2005.
8. N. D. Halsey, *Potential flow analysis of multielement airfoils using conformal mapping*, American Institute of Aeronautics and Astronautics. Journal **17** (1979), no. 12, 1281–1288.
9. P. Hertling, *An effective Riemann Mapping Theorem*, Theoretical Computer Science **219** (1999), 225 – 265.
10. M. Jeong and V. Mityushev, *Bergman kernel for circular multiply connected domains*, Pacific Journal Mathematics **237** (2007), no. 1, 145–157.
11. P. Koebe, *Über die konforme Abbildung mehrfach-zusammenhängender Bereiche*, Jahresber. Dt. Math. Ver. **19** (1910), 339 – 348.
12. P. Koebe, *Abhandlungen zur theorie der konformen abbildung, iv*, Acta Math **41** (1918), 305–344.
13. V. V. Mityushev and S. V. Rogosin, *Constructive methods for linear and nonlinear boundary value problems for analytic functions*, CRC Monographs and Surveys in Pure and Applied Mathematics, vol. 108, Chapman and Hall, 2000.
14. Z. Nehari, *Conformal mapping*, Dover, 1975.
15. M. Schiffer, *Some recent developments in the theory of conformal mapping*, Appendix to: R. Courant. Dirichlet Principle, Conformal Mapping and Minimal Surfaces, Interscience Publishers, 1950.
16. M. Tsuji, *Potential theory*, Chelsea, New York, 1975.
17. K. Weihrauch, *Computable analysis, an introduction*, Springer-Verlag, 2000.
18. M. Ziegler and V. Brattka, *Computability in linear algebra*, Theoretical Computer Science **326** (2004), 187–211.

Canonical Effective Subalgebras of Classical Algebras as Constructive Metric Completions

Andrej Bauer[1] and Jens Blanck[2]

[1] Faculty of Mathematics and Physics
University of Ljubljana, Slovenia
Andrej.Bauer@andrej.com
[2] Computer Science Department
Swansea University, UK
j.e.blanck@swansea.ac.uk

Abstract. We prove general theorems about unique existence of effective subalgebras of classical algebras. The theorems are consequences of standard facts about completions of metric spaces within the framework of constructive mathematics, suitably interpreted in realizability models. We work with general realizability models rather than with a particular model of computation. Consequently, all the results are applicable in various established schools of computability, such as type 1 and type 2 effectivity, domain representations, equilogical spaces, and others.

1 Introduction

Given an algebra, by which we mean a set with constants and operations, is there a largest subalgebra which carries a computable structure, and is the structure unique up to computable isomorphism? Without further assumptions the answer is in general negative. For example, within the context of Recursive Mathematics every computable subfield of reals may be properly extended to a subfield which is again computable, and this remains true even if we require the subfields to be effectively complete. However, as was proved by Moschovakis [1], by requiring also that the strict linear order be semidecidable, we are left with only one choice, namely the recursive reals. An analogous result for type 2 effectivity was established by Hertling [2].

We show how these results, as well as others, can be seen as standard facts about completions of metric spaces in the context of constructive mathematics, suitably interpreted in realizability models. We prove two main theorems which together give conditions under which an algebra $\mathcal{A}$, equipped with a complete metric d, has a unique effective subalgebra $\mathcal{B}$ that is effectively complete and for which the relation $d(x, y) < q$ is semidecidable in $x, y \in \mathcal{B}$ and $q \in \mathbb{Q}$.

Rather than choosing a specific model of computation, we work in a general realizability model. Thus our results apply to established schools of computable mathematics, such as type 1 and type 2 effectivity, domain representations, equilogical spaces, and others.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 37-48
http://drops.dagstuhl.de/opus/volltexte/2009/2257

The outline of the extended abstract is as follows. Sections 2–4 introduce the necessary background, namely realizability models, algebras, and premetric spaces. Section 5 states the main theorems, from which two specific important cases are inferred in Section 6. We conclude with a brief discussion of possible further directions of research.

2 Assemblies and Realizability

Among the different kinds of realizability the most suitable one for our purposes is *relative realizability*, because it subsumes type 1 and type 2 effectivity, domain representations, equilogical spaces, and other standard models of computation, see [3]. We review the basic definitions here and refer the readers to [4] for background material on realizability.

A *partial combinatory algebra (PCA)* is a set A with a partial application operation[3] $\cdot$ such that there exist elements $\mathtt{k}, \mathtt{s} \in A$ satisfying $\mathtt{k}\ x\ y = x$ and[4] $\mathtt{s}\ x\ y\ z \simeq (x\ z)\ (y\ z)$. A PCA is a general model of computation which supports encoding of pairs, natural numbers, recursion, partial recursive functions, etc. An *elementary sub-PCA* is a subset $B \subseteq A$ which is closed under application and contains $\mathtt{k}$ and $\mathtt{s}$ suitable for A. For the rest of the discussion we fix a PCA $\mathbb{A}$ and an elementary sub-PCA $\mathbb{A}^\# \subseteq \mathbb{A}$. The elements of $\mathbb{A}$ as "arbitrary" and those of $\mathbb{A}^\#$ as "effective" data or programs, although the exact meaning of these words depends on the particular choice of $\mathbb{A}$ and $\mathbb{A}^\#$.

An *assembly* $\mathbf{S} = (S, \Vdash_S)$ is a set S together with a *realizability relation* $\Vdash_S \subseteq \mathbb{A} \times S$, such that for every $x \in S$ there is at least one $\mathtt{x} \in \mathbb{A}$ for which $\mathtt{x} \Vdash_S x$. A *realized map* $f : \mathbf{S} \to \mathbf{T}$ between assemblies is a map $f : S \to T$ between the underlying sets which is *tracked* by some $\mathtt{f} \in \mathbb{A}^\#$, which means that whenever $\mathtt{x} \Vdash_S x$ then[5] $\mathtt{f}\ \mathtt{x}{\downarrow}$ and $\mathtt{f}\ \mathtt{x} \Vdash_T f(x)$. Note that we require maps to be realized by the elements of the *sub*algebra $\mathbb{A}^\#$. Assemblies and realized maps form a category Asm. An assembly $\mathbf{S}$ is *modest*, or a *modest set*, if each realizer realizes at most one element: for all $\mathtt{r} \in A$, $x, y \in S$, if $\mathtt{r} \Vdash_S x$ and $\mathtt{r} \Vdash_S y$ then $x = y$.

An assembly $\mathbf{S}$ is equivalent to a multi-valued representation $\delta_S : \mathbb{A} \to \mathcal{P}(S)$ via the correspondence $\mathtt{x} \Vdash_S x \iff x \in \delta_S(\mathtt{x})$. A modest set is equivalent to a single-valued representation. Traditional schools of computable mathematics typically use (single-valued) representations, for example:

- When $\mathbb{A} = \mathbb{A}^\# = \mathbb{N}$ is the first Kleene algebra, the modest sets are equivalent to type 1 representations, or *numbered sets*, which are used in the study of recursive mathematics. In this model "effective" means "computable by (type 1) Turing machine".

[3] We write $x\ y$ instead of $x \cdot y$, and associate application to the left, $x\ y\ z = (x\ y)\ z$.
[4] Kleene equality $a \simeq b$ means that if one side is defined then so is the other and they are equal.
[5] The expression $t{\downarrow}$ means "t is defined".

- When $\mathbb{A} = \mathbb{N}^{\mathbb{N}}$ is the second Kleene algebra and $\mathbb{A}^{\#}$ the subalgebra of total computable functions we get *type 2 representations*. In this case "effective" means "computable by type 2 Turing machine".
- The case $\mathbb{A} = \mathbb{A}^{\#} = \mathbb{N}^{\mathbb{N}}$ is the continuous version of type 2 effectivity in which "effective" means "continuously realized".
- When $\mathbb{A}$ is a universal Scott domain and $\mathbb{A}^{\#}$ its computable analogue, the modest assemblies are equivalent to domain representations and computable maps between them. Of course, "effective" is now interpreted in the sense of domain representations.
- With Scott's graph model $\mathbb{A} = \mathcal{P}\omega$ and its r.e. counterpart $\mathbb{A}^{\#} = \mathsf{RE}$ we obtain effective equilogical spaces [3].

Single-valued representations seem to be preferred to general assemblies, perhaps because from a programmer's perspective it makes little sense to use one realizer for representing several things, although lately multi-valued type 2 representations have turned out to be useful [5]. We use assemblies because they contain the category of sets, which allows us to consider classical and effective algebras in a single framework. Realizability toposes could be used instead, but assemblies are easier to describe and work with.

2.1 The realizability interpretation of first-order logic

Assemblies supports an interpretation of first-order intuitionistic logic in which a formula is deemed valid when there is an element $\mathtt{r} \in \mathbb{A}^{\#}$ witnessing it. The interpretation is given in terms of a *realizability relation* $\mathtt{r} \Vdash \phi$ which is read as "$\mathtt{r}$ realizers ϕ", and is defined inductively on the structure of the sentence ϕ:

- always $\mathtt{r} \Vdash \top$, and never $\mathtt{r} \Vdash \bot$,
- $\langle \mathtt{p}, \mathtt{q} \rangle \Vdash \phi \wedge \psi$ iff $\mathtt{p} \Vdash \phi$ and $\mathtt{q} \Vdash \psi$,[6]
- $\langle \overline{0}, \mathtt{r} \rangle \Vdash \phi \vee \psi$ iff $\mathtt{r} \Vdash \phi$, and $\langle \overline{1}, \mathtt{r} \rangle \Vdash \phi \vee \psi$ iff $\mathtt{r} \Vdash \psi$,[7]
- $\mathtt{r} \Vdash \phi \Rightarrow \psi$ iff for all $\mathtt{q} \in \mathbb{A}$, if $\mathtt{q} \Vdash \phi$ then $\mathtt{r}\,\mathtt{q}{\downarrow}$ and $\mathtt{r}\,\mathtt{q} \Vdash \psi$,
- $\mathtt{r} \Vdash \forall x \in \mathbf{S}\,.\,\phi(x)$ iff for all $\mathtt{a} \in \mathbb{A}$, $a \in S$, if $\mathtt{a} \Vdash_S a$ then $\mathtt{r}\,\mathtt{a}{\downarrow}$ and $\mathtt{r}\,\mathtt{a} \Vdash \phi(a)$,
- $\langle \mathtt{a}, \mathtt{r} \rangle \Vdash \exists x \in \mathbf{S}\,.\,\phi(x)$ iff for some $a \in S$, $\mathtt{a} \Vdash_S a$ and $\mathtt{r} \Vdash \phi(a)$,
- $\mathtt{r} \Vdash a = b$ iff $a = b$.

A sentence ϕ is *valid*, written $\models \phi$, when there exists $\mathtt{r} \in A^{\#}$ such that $\mathtt{r} \Vdash \phi$. Note that $\mathtt{r}$ must be an element of the *sub*algebra $A^{\#}$. A formula with free variables is valid when its universal closure is valid. Intuitionistic logic is sound with respect to the realizability relation: if intuitionistic logic proves ϕ then ϕ is valid.

[6] $\langle \mathtt{p}, \mathtt{q} \rangle$ is the encoding of the pair whose components are $\mathtt{p}$ and $\mathtt{q}$.
[7] $\overline{n}$ is the encoding of the natural number n.

2.2 The role of double negation

Negation $\neg\phi$ is defined as $\phi \Rightarrow \bot$. This gives us

$$\mathtt{r} \Vdash \neg\phi \text{ iff for all } \mathtt{q} \in \mathbb{A}, \text{ not } \mathtt{q} \Vdash \phi,$$
$$\mathtt{r} \Vdash \neg\neg\phi \text{ iff there is } \mathtt{q} \in \mathbb{A} \text{ such that } \mathtt{q} \Vdash \phi.$$

A realizer $\mathtt{r}$ of a doubly negated formula $\neg\neg\phi$ does not carry any information about the computational content of ϕ, because we may replace it with any other. Thus double negation is a way of erasing the constructive or computational meaning of a formula.

A formula which is equivalent to its double negation is called $\neg\neg$-*stable*. Since $\phi \Rightarrow \neg\neg\phi$ is always intuitionistically provable, only the direction $\neg\neg\phi \Rightarrow \phi$ is relevant. An important family of stable formulas are the *negative* ones, which are those built from $\bot$, $\top$, $=$, $\neg$, $\wedge$, $\Rightarrow$, $\forall$, and possibly other $\neg\neg$-stable primitive relations. The realizers of a $\neg\neg$-stable formula ϕ are computationally irrelevant in the sense that any information that can be computed with the help of a realizer $\mathtt{r} \models \phi$ can be computed without $\mathtt{r}$, the extreme case of which is that $\mathtt{r}$ itself can be computed from nothing, as long as it exists.

A mono $i : \mathbf{S} \rightarrowtail \mathbf{T}$ is $\neg\neg$-*stable* when $\models \forall\, x \in \mathbf{T}\,.\,(\neg\neg(x \in \mathbf{S}) \Rightarrow x \in \mathbf{S})$, where "$x \in \mathbf{S}$" is a shorthand for $\exists\, y \in \mathbf{S}\,.\,i(y) = x$. Up to isomorphism, such a mono is a restriction of $\mathbf{T}$ to a subset $S \subseteq T$, and the realizability relation $\Vdash_S$ is $\Vdash_T$ restricted to S. Thus the $\neg\neg$-stable monos of $\mathbf{T}$ correspond to subsets of T.

A mono $i : \mathbf{S} \rightarrowtail \mathbf{T}$ is $\neg\neg$-*dense* when $\models \forall\, y \in \mathbf{T}\,.\,\neg\neg\exists\, x \in \mathbf{S}\,.\,y = i(x)$. Such a mono is always isomorphic to a mono $i : \mathbf{S} \rightarrowtail \mathbf{T}$ such that $S = T$ and i is the identity map. Thus the $\neg\neg$-dense monos play in Asm the role of reductions between representations.

2.3 Semidecidable predicates

To illustrate how the realizability interpretation is used, and for later use, we explain how to treat semidecidable predicates in Asm. We say that a mono $i : \mathbf{S} \rightarrowtail \mathbf{T}$, seen as a predicate on $\mathbf{T}$, is *semidecidable* when

$$\models \forall\, x \in \mathbf{T}\,.\,\exists\, f \in \{0,1\}^{\mathbf{N}}\,.\,(x \in \mathbf{S} \iff \exists\, n \in \mathbf{N}\,.\,f(n) = 1)\,.$$

Here $\mathbf{N}$ is the modest set of natural numbers, cf. Section 3.2, and the exponential $\{0,1\}^{\mathbf{N}}$ is the modest set of those maps $\mathbb{N} \to \{0,1\}$ which are tracked by an element of $\mathbb{A}$. Markov Principle, which is valid in Asm, states that a formula of the form $\exists\, n \in \mathbf{N}\,.\,f(n) = 1$ is $\neg\neg$-stable. Therefore only $\neg\neg$-stable predicates can be semidecidable. We assume without loss of generality that $i : \mathbf{S} \rightarrowtail \mathbf{T}$ is $\neg\neg$-stable and that i is a subset inclusion. Validity of the above formula is then equivalent to there being $\mathtt{r} \in \mathbb{A}^{\#}$ which works as follows: if $\mathtt{x} \Vdash_T x$ then, for all $n \in \mathbb{N}$, $\mathtt{r}\ \mathtt{x}\ \overline{n}\!\downarrow$ and $\mathtt{r}\ \mathtt{x}\ \overline{n} \in \{\overline{0}, \overline{1}\}$, and furthermore, $x \in S$ if, and only if, $\mathtt{r}\ \mathtt{x}\ \overline{n} = \overline{1}$ for some $n \in \mathbb{N}$. The semidecidable predicates have the expected properties: decidable predicates are semidecidable, and the semidecidable predicates are closed under conjunctions and existential quantification over $\mathbf{N}$.

In type 1 effectivity our notion of semidecidability coincides with the usual one, while in type 2 effectivity the notion is known as "r.e. open subset". In a purely topological model, such as the continuous version of type 2 effectivity "semidecidable" means "topologically open". The interpretation in Set is trivial because there every subset is semidecidable (even decidable) thanks to the law of excluded middle.

3 Algebras

A *signature* Σ for an algebra is given by a list of *function symbols* $\mathsf{f}_1, \ldots, \mathsf{f}_k$. Each f_i has an *arity*, which is a non-negative integer. The set $\mathsf{Term}(\Sigma)$ of *terms over* Σ is built inductively from variables $x, y, z, \ldots$, and terms $\mathsf{f}(t_1, \ldots, t_n)$, where f is a function symbol with arity n and $t_1, \ldots, t_n$ are terms. We assume that a standard Gödel numbering $\ulcorner - \urcorner : \mathbb{N} \to \{\star\} + \mathsf{Term}(\Sigma)$ of terms is given.[8]

A *Σ-algebra* $\mathcal{A}$ in a category C with finite products is given by an object $|\mathcal{A}|$ called the *carrier* of $\mathcal{A}$, and for each function symbol f with arity n a morphism $\mathsf{f}^{\mathcal{A}} : |\mathcal{A}|^n \to |\mathcal{A}|$, called an *operation*. Each term $t \in \mathsf{Term}(\Sigma)$ whose free variables are among $x_1, \ldots, x_k$ determines a morphism $|\mathcal{A}|^k \to |\mathcal{A}|$: a variable x_i is the i-th projection, while a term $\mathsf{f}(t_1, \ldots, t_n)$ is the composition of $\mathsf{f}^{\mathcal{A}}$ with the the morphisms determined by $t_1, \ldots, t_n$. A *subalgebra* of $\mathcal{A}$ is a Σ-algebra $\mathcal{B}$ with a mono $\mathcal{B} \rightarrowtail \mathcal{A}$ such that the operations in $\mathcal{A}$ restrict to operations in $\mathcal{B}$. We write $\mathcal{B} \leq \mathcal{A}$ when $\mathcal{B}$ is a subalgebra of $\mathcal{A}$.

If C and D are categories with finite products and $F : \mathsf{C} \to \mathsf{D}$ a functor which preserves finite products then a Σ-algebra $\mathcal{A}$ in C is mapped by F to a Σ-algebra $F(\mathcal{A})$ in $\mathcal{D}$, where $|F(\mathcal{A})| = F(|\mathcal{A}|)$ and $\mathsf{f}^{F(\mathcal{A})} = F(\mathsf{f}^{\mathcal{A}})$. The mapping preserves valid equations in $\mathcal{A}$, and also reflects them if F is faithful.

A *(first-order) formula ϕ over Σ* is a formula in first-order logic with terms over Σ. If $\mathcal{A}$ is a Σ-algebra in C, where C is either Set or Asm, then we may interpret such a ϕ as a statement about $\mathcal{A}$: the terms are interpreted according to $\mathcal{A}$, while the logic is interpreted either in the standard set-theoretic way, as given by Tarski, or using the realizability interpretation from Section 2.1. We write $\mathcal{A} \models_{\mathsf{C}} \phi$ when ϕ is valid when so interpreted. We refer to interpretations in Set as "classical" and those in Asm as "effective". More generally the adjectives "classical" and "effective" are used distinguish between the two settings. For example, a "classical algebra" is an algebra in Set, while an "effective algebra" is one in Asm. Similarly, a (classical) space is "classically complete" if the formula expressing completeness is valid in Set, and an (effective) space is "effectively complete" if the same formula is valid in Asm. Note however that the exact interpretation of "effective" depends on the choice of the underlying computational model.

[8] The special value $\ulcorner n \urcorner = \star$ signifies that n is not a valid Gödel code. This is not necessary for enumeration of all terms, but we do need it when we consider enumerations of closed terms, of which there may be none.

3.1 Subalgebras generated by subassemblies

Suppose $\mathcal{A}$ is classical Σ-algebra, and consider a subset $C \subseteq |\mathcal{A}|$ of the carrier. Then there exists the least subalgebra $\mathcal{I} \leq \mathcal{A}$ such that $C \subseteq |\mathcal{I}|$, namely the intersection of all subalgebras that contain C. We say that $\mathcal{I}$ is *generated by* C and denote it by $\langle C \rangle_{\mathcal{A}}$.

Now let $\mathcal{A}$ be an effective Σ-algebra and $\mathbf{C} \rightarrowtail |\mathcal{A}|$ a subassembly of $|\mathcal{A}|$. There exists the least effective subalgebra $\langle \mathbf{C} \rangle_{\mathcal{A}} \leq \mathcal{A}$ containing $\mathbf{C}$ as a subassembly. One way of proving this is to work in the internal language of the realizability topos $\mathsf{RT}(\mathbb{A}, \mathbb{A}^{\#})$, where $\langle \mathbf{C} \rangle_{\mathcal{A}}$ is the intersection of all subalgebras of $\mathcal{A}$ that contain the assembly $\mathbf{C}$, just like in Set. A special case is the *initial subalgebra* $\langle \emptyset \rangle_{\mathcal{A}}$ which is generated by the empty subassembly. It is always modest, even if $\mathcal{A}$ is not, and is effectively enumerated by a realized map $e : \mathbf{N} \to \{\star\} + \langle \emptyset \rangle_{\mathcal{A}}$ which is essentially the composition of the Gödel numbering of the closed terms over Σ with their interpretation in $\mathcal{A}$.

3.2 Algebras characterized by their universal properties

When a classical algebra is characterized up to isomorphism by a universal property, we may use the property to identify the corresponding effective algebra. It turns out that we usually get the generally accepted "correct" computability structure:

- The natural numbers $\mathbb{N}$ are the initial commutative semiring with unit. In Asm this is the modest set $\mathbf{N} = (\mathbb{N}, \Vdash_{\mathbb{N}})$ where $\overline{n} \Vdash_{\mathbb{N}} n$ for each $n \in \mathbb{N}$.
- The initial commutative ring in Set are the integers $\mathbb{Z}$, while in Asm it is the modest set $\mathbf{Z} = (\mathbb{Z}, \Vdash_{\mathbb{Z}})$ where, for each $m, n \in \mathbb{N}$ and $k \in \mathbb{Z}$, $\langle \overline{m}, \overline{n} \rangle \Vdash_{\mathbb{Z}} k$ when $k = m - n$.
- The field of fractions over the integers in Set are the rationals $\mathbb{Q}$. In Asm it is the modest set $\mathbf{Q} = (\mathbb{Q}, \Vdash_{\mathbb{Q}})$ where, for all $k, m, n \in \mathbb{N}$ and $q \in \mathbb{Q}$, $\langle \overline{k}, \overline{m}, \overline{n} \rangle \Vdash_{\mathbb{Q}} q$ when $q = (k - m)/n$.
- The reals $\mathbb{R}$ are the Cauchy-complete archimedean ordered field. The counterpart in assemblies is the modest set $\mathbf{R} = (R, \Vdash_{\mathbb{R}})$ where $\mathtt{x} \Vdash_{\mathbb{R}} x$ when $\mathtt{x} \in \mathbb{A}$ represents a fast Cauchy sequence[9] of rational numbers converging to x, and $R = \{x \in \mathbb{R} \mid \exists\, \mathtt{x} \in A \,.\, \mathtt{x} \Vdash_{\mathbb{R}} x\}$. Depending on the PCA $\mathbb{A}$ the set R could consist just of the computable reals, or all reals, or all reals computable with respect to an oracle, etc.

Unfortunately, such universal characterizations are not always available.

Apart from first-order formulas over a signature Σ we shall also consider more general first-order formulas which additionally refer to the natural numbers $\mathbb{N}$, the integers $\mathbb{Z}$, and the rationals $\mathbb{Q}$. We call them *extended formulas over the signature* Σ. When they are interpreted in Set, the symbols $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ receive their usual meaning, whereas in Asm we interpret them as the corresponding assemblies $\mathbf{N}$, $\mathbf{Z}$, and $\mathbf{Q}$, as described above. An extended formula may *not* refer

[9] A sequence $(a_n)_n$ is fast Cauchy if $|a_m - a_n| \leq 2^{-\min(m,n)}$ for all $m, n \in \mathbb{N}$.

directly to the real numbers because Propositions 1 and 2 fail for formulas that refer to the reals. An extended formula over Σ which is also negative is called *extended negative formula over* Σ.

3.3 Transfer of algebras from sets to assemblies

Every set S may be represented as a *constant assembly* $\nabla S = (S, \Vdash_{\nabla S})$ where $\mathtt{r} \Vdash_S x$ holds for all $\mathtt{r} \in A$ and $x \in S$. In other words, in ∇S every realizer realizes every element. Every function $f : S \to T$ between sets S and T is realized as a map $\nabla f : \nabla S \to \nabla T$, for example by the realizer $\mathsf{s}\,\mathsf{k}\,\mathsf{k}$. This gives us a full and faithful embedding $\nabla : \mathsf{Set} \to \mathsf{Asm}$.

The functor ∇ preserves finite limits, and finite products in particular. Therefore, ∇ maps a Σ-algebra $\mathcal{A}$ in Set to a Σ-algebra $\nabla\mathcal{A}$ in Asm. The mapping preserves and reflects equations because ∇ is faithful. Even more, it preserves all negative formulas:

Proposition 1. *Let $\mathcal{A}$ be a Σ-algebra in Set and ϕ an extended negative formula over Σ. Then $\mathcal{A} \models_{\mathsf{Set}} \phi$ if, and only if, $\nabla\mathcal{A} \models_{\mathsf{Asm}} \phi$.*

A $\neg\neg$-*dense* subalgebra $\mathcal{B} \leq \mathcal{A}$ in Asm is a subalgebra for which the mono $|\mathcal{B}| \rightarrowtail |\mathcal{A}|$ is $\neg\neg$-dense. We may assume that $|\mathcal{B}| = |\mathcal{A}|$ and that the mono $|\mathcal{B}| \rightarrowtail |\mathcal{A}|$ is the identity map.

Proposition 2. *Let $\mathcal{A}$ be an effective Σ-algebra and $\mathcal{B} \leq \mathcal{A}$ a $\neg\neg$-dense subalgebra. Then $\mathcal{A}$ and $\mathcal{B}$ satisfy the same extended negative formulas over Σ.*

The proofs of both propositions are standard exercises in performing an induction over the structure of ϕ. The deeper reason for their truth is the fact that sets are precisely the sheaves for the double negation topology on the realizability topos $\mathsf{RT}(\mathbb{A}, \mathbb{A}^{\#})$.

4 Premetric spaces

A *metric algebra* is a Σ-algebra $\mathcal{A}$ whose carrier is a metric space and the operations are continuous maps. A metric algebra is *complete* if its carrier is a complete metric space. We face a difficulty when we try to transfer metric algebras from sets to assemblies: ∇ maps a metric $d : S \times S \to \mathbb{R}$ to the realized map $\nabla d : \nabla S \times \nabla S \to \nabla\mathbb{R}$, which is not a metric anymore because its codomain $\nabla\mathbb{R}$ is not the object $\mathbf{R}$ of real numbers in Asm. To overcome the problem we use a formulation of metric spaces which does not directly refer to real numbers, is classically equivalent to the usual metric spaces,[10] and is constructively acceptable. Such a notion, namely *premetric spaces*, was defined by Fred Richman [6]. We use a slight variation:

[10] We allow infinite distances but that is inessential.

Definition 1. *A* premetric space (X, d) *is a set* X *with a ternary relation* $d \subseteq X \times X \times \mathbb{Q}$ *satisfying the following conditions, where we write* $d(x, y) \leq q$ *instead of* $(x, y, q) \in d$*:*

1. *if* $q < 0$ *then not* $d(x, y) \leq q$*,*
2. $d(x, y) \leq 0$ *if, and only if,* $x = y$*,*
3. *if* $d(x, y) \leq q$ *then* $d(y, x) \leq q$*,*
4. *if* $d(x, y) \leq q$ *and* $d(y, z) \leq r$ *then* $d(x, z) \leq q + r$*,*
5. $d(x, y) \leq q$ *if, and only if,* $d(x, y) \leq r$ *for all* $r > q$*.*

Richman's definition also requires that for all $x, y \in X$ there is a rational $q \geq 0$ such that $d(x, y) \leq q$. We omit the requirement because we do not need it, and because it is the only axiom which is not a negative formula.

Every metric space (M, d) is a premetric space (M, d') with $d' = \{(x, y, q) \in X \times X \times \mathbb{Q} \mid d(x, y) \leq q\}$. Classically, the converse holds if we allow infinite distances[11] because the metric d may be recovered from the premetric d' as $d(x, y) = \inf \{q \in \mathbb{Q} \mid d'(x, y) \leq q\}$. Constructively however the infimum need not exist.

The basic theory of premetric spaces parallels that of metric spaces. The notions of completeness, continuity, density, etc., are all easily expressed in terms of the premetric. In fact, the whole theory is constructively valid (even without choice), as was shown by Richman [6]. Despite our allowing infinite distances, the following theorem still holds constructively, and is therefore valid both in Set and Asm.

Proposition 3. *Let* X *be a premetric space and* $e : X \to Y$ *its* completion*, i.e., an isometry with a dense image into a complete premetric space* Y*. Then every locally uniformly continuous*[12] $f : X \to Z$ *to a complete premetric space* Z *has a unique locally uniformly continuous extension* $\overline{f} : Y \to Z$ *along* e*.*

An easy consequence of the theorem is that any two completions of a premetric space are isometrically isomorphic.

When a premetric space (X, d) is transferred from Set to Asm by ∇, the relation $d \subseteq X \times X \times \mathbb{Q}$ is mapped to the mono $\nabla d \rightarrowtail \nabla X \times \nabla X \times \nabla \mathbb{Q}$, which is $\neg\neg$-stable. The axioms for premetric structure are extended negative formulas, so by Proposition 1 they are preserved. This proves the following proposition:

Proposition 4. *If* (X, d) *is a classical premetric space then* $(\nabla X, \nabla d)$ *is an effective premetric space. Furthermore,* (X, d) *and* $(\nabla X, \nabla d)$ *satisfy the same extended negative formulas.*

Moreover, ∇ preserves the completeness property, which follows easily from the observation that the exponential assembly $(\nabla X)^{\mathbb{N}}$ is isomorphic to $\nabla(X^{\mathbb{N}})$:

Proposition 5. *A classical premetric space* (X, d) *is classically complete if, and only if,* $(\nabla X, \nabla d)$ *is effectively complete.*

[11] With Richman's extra axiom the correspondence between metric and premetric spaces is exact, classically.

[12] A map is locally uniformly continuous if it is uniformly continuous on every closed ball.

4.1 Complete subalgebras

When $\mathcal{A}$ is a classical complete premetric Σ-algebra we may ask whether every subalgebra $\mathcal{B} \leq \mathcal{A}$ is contained in the least *complete* subalgebra $\overline{\mathcal{B}} \leq \mathcal{A}$. The premetric closure $\overline{|\mathcal{B}|}$ in $|\mathcal{A}|$ is an obvious candidate. For it to be a subalgebra, each operation $\mathsf{f}^{\mathcal{B}} : |\mathcal{B}|^n \to |\mathcal{B}|$ must extend to a map $\mathsf{f}^{\overline{\mathcal{B}}} : \overline{|\mathcal{B}|}^n \to \overline{|\mathcal{B}|}$, which it does by Theorem 3 as long as the operations on $\mathcal{B}$ are locally uniformly continuous. We have proved the following proposition.

Proposition 6. *Let $\mathcal{A}$ be a classical complete Σ-algebra. The closure $\overline{|\mathcal{B}|}$ of the carrier of a subalgebra $\mathcal{B} \leq \mathcal{A}$ is the least complete subalgebra of $\mathcal{A}$ containing $\mathcal{B}$, provided the operations on $\mathcal{B}$ are locally uniformly continuous.*

The argument which proved Proposition 6 is constructively valid. Its interpretation in Asm gives the following effective version.

Proposition 7. *Let $\mathcal{A}$ be an effective*[13] *complete Σ-algebra. The effective closure $\overline{|\mathcal{B}|}$ of the carrier of a subalgebra $\mathcal{B} \leq \mathcal{A}$ is the least effective complete subalgebra of $\mathcal{A}$ containing $\mathcal{B}$, provided the operations on $\mathcal{B}$ are effectively locally uniformly continuous.*

We remark that the complete subalgebra $\overline{\mathcal{B}}$ generated by $\mathcal{B}$ is modest if $\mathcal{B}$ is modest, even if $\mathcal{A}$ is not.

5 Main Theorems

The results of the previous sections give us a method for finding canonical effective subalgebras of classical algebras. Let $\mathcal{A}$ be a classical premetric Σ-algebra. In general there will be many effective subalgebras $\mathcal{B} \leq \nabla\mathcal{A}$, each carving out a different piece of $\mathcal{A}$ with its own effective structure. Our first theorem gives conditions which severely cut down the number of possibilities. Define the relation $d(x, y) < q$ for $x, y \in |\mathcal{A}|$ and $q \in \mathbb{Q}$ by $d(x, y) < q \iff \exists\, r \in \mathbb{Q}\,.\, d(x, y) \leq r \wedge r < q$.

Theorem 1. *Suppose $\mathcal{A}$ is a classical premetric Σ-algebra in which the initial subalgebra $\langle\emptyset\rangle_{\mathcal{A}}$ is classically dense. Up to isomorphism, there is at most one effectively complete subalgebra $\mathcal{B} \leq \nabla\mathcal{A}$ on which the relation $d(x, y) < q$ is semidecidable.*

We omit the proof, and just note that $\mathcal{B}$, if it exists, is the effective completion of the initial subalgebra $\langle\emptyset\rangle_{\nabla\mathcal{A}}$.

When the initial subalgebra $\langle\emptyset\rangle_{\mathcal{A}}$ is not dense, Theorem 1 cannot be applied. Quite often this can be fixed with a judicious addition of new constants and operations. For example, the initial subring of the ring $\mathcal{C}[0, 1]$ of continuous real functions on the closed unit interval is the ring of integers (embedded as

[13] To be precise, we are talking about an "effectively complete effectively premetric effective Σ-algebra".

constant functions), which is not dense. If we adjoin the identity function and the constant function $\frac{1}{2}$ as primitive constants, the initial subalgebra will be the ring of polynomials whose coefficients are dyadic rationals,[14] which is dense by the (classical) Stone-Weierstraß theorem.

Another way to deal with non-dense initial subalgebra is to replace $\langle\emptyset\rangle_{\mathcal{A}}$ in Theorem 1 with a chosen dense subalgebra $\mathcal{D} \leq \mathcal{A}$, but then the statement is that there is at most one effectively complete subalgebra of $\nabla\mathcal{A}$ containing $\nabla\mathcal{D}$ for which $d(x, y) < q$ is semidecidable.

The next theorem complements Theorem 1 by giving conditions for existence of subalgebras.

Theorem 2. *Let $\mathcal{A}$ be a classical complete premetric Σ-algebra. Suppose the relation $d(x, y) < q$ is semidecidable on $\langle\emptyset\rangle_{\nabla\mathcal{A}}$ and the operations of $\langle\emptyset\rangle_{\nabla\mathcal{A}}$ are effectively locally uniformly continuous. Then $\nabla\mathcal{A}$ has an effective complete subalgebra on which the relation $d(x, y) < q$ is semidecidable.*

Again, we omit the proof. We know from the previous theorem that the desired subalgebra must be the completion of $\langle\emptyset\rangle_{\nabla\mathcal{A}}$, from which a concrete representation can be computed: because $\langle\emptyset\rangle_{\nabla\mathcal{A}}$ is essentially represented by a Gödel numbering of closed terms, its completion is represented by sequences of (Gödel codes of) closed terms that are fast Cauchy.

6 Applications

In this section we apply the results to two common scenarios.

6.1 Discrete premetric spaces

The simplest kind of complete premetric algebras are the discrete ones. Let $\mathcal{A}$ be a classical Σ-algebra and define the *discrete premetric* on $|\mathcal{A}|$ by

$$d(x, y) \leq q \iff (q < 1 \implies x = y),$$

which of course corresponds to the metric that takes on only values 0 and 1. In the discrete premetric every set is complete and every map is uniformly continuous. Therefore, half of the conditions in Theorems 1 and 2 are trivially satisfied. Furthermore, a discrete premetric is semidecidable on $\mathcal{B} \leq \nabla\mathcal{A}$ if, and only if, equality is semidecidable on $\mathcal{B}$, because $x = y \iff d(x, y) < 1$ and $d(x, y) < q \iff (q > 1 \vee x = y)$. Thus we obtain the following result.

Proposition 8. *Suppose $\mathcal{A}$ is a* finitely generated *classical Σ-algebra. Up to isomorphism, there is at most one effective structure on $\mathcal{A}$ for which the operations and the generators are effective, and equality is semidecidable. Furthermore, if there is such an effective structure, it is isomorphic to the effective subalgebra $\langle\{a_1, \ldots, a_n\}\rangle_{\nabla\mathcal{A}}$ of $\nabla\mathcal{A}$ generated by the generators $a_1, \ldots, a_n$ for $\mathcal{A}$.*

[14] A dyadic rational is one of the form $n/2^k$.

More precisely, the first part of the proposition states that there is at most one realizability relation $\Vdash_{\mathcal{A}}$ on the set $|\mathcal{A}|$ which turns the classical algebra $\mathcal{A}$ into an effective one[15] such that equality is semidecidable. The second part gives an explicit description of the effective structure, and also implies that the resulting assembly is modest.

In the context of type 1 effectivity Proposition 8 was first proved by Mal'cev, see [7] and [8, Theorem 4.1.2]. He actually considered two versions, one with general recursive functions and another with partial recursive functions. Our result corresponds to the partial recursive case because all partial recursive functions are representable in a PCA.

6.2 The real numbers

The real numbers form a classical ordered field, and a classical complete premetric space with the usual premetric $d(x,y) \leq q \iff |x-y| \leq q$. A slight complication is division because it is a partial operation. The journal version of this extended abstract will include a proper treatment of partial operations. For now, we circumvent division by viewing the real numbers as a ring $\mathcal{R} = (\mathbb{R}, 0, 1, \frac{1}{2}, +, -, \times)$ with a primitive constant $\frac{1}{2}$. The initial subalgebra is the ring of dyadic rationals, which is dense in $\mathbb{R}$. The relation $|x-y| < q$ is semidecidable, even decidable when x and y are dyadic rationals and q a rational. The operations are easily seen to be effectively locally uniformly continuous. Thus the conditions of both main theorems are satisfied. Up to isomorphism there is exactly one effectively complete effective subring $\mathbf{R} \leq \nabla\mathcal{R}$ on which the relation $d(x,y) < q$ is semidecidable. We may replace semidecidability of $d(x,y) < q$ with semidecidability of the strict order relation $x < y$ because $d(x,y) < q \iff -q < x - y < q$ and

$$x < y \iff \exists\, q, r \in \mathbb{Q}\,.\, \exists\, k \in \mathbb{N}\,.\, \big(d(x,q) < 2^{-k} \wedge d(y,r) < 2^{-k} \wedge q + 2^{-k+2} < r\big)\,.$$

The dyadic rationals have *approximate division*: for all $k \in \mathbb{N}$ and dyadic rationals a and $b \neq 0$ there exists a dyadic rational c such that $d(a, bc) < 2^{-k}$. The completion of a premetric ring with approximate division is always a field, constructively speaking. By putting all these observations together we get the following result.

Proposition 9. *Up to isomorphism, there is exactly one effectively complete effective subfield of the real numbers for which the strict linear order is is semidecidable.*

When the proposition is specialized to type 2 effectivity it gives Hertling's result [2] about type 2 representations of reals, while the interpretation in type 1 effectivity corresponds to a result of Moschovakis [1].

[15] This means that the operations and generators are realized by elements of $\mathbb{A}^{\#}$.

7 Conclusion

The relation d on a premetric space (X, d) induces a uniform structure on X whose (basic) entourages are $E_q = \{(x, y) \in X \times X \mid d(x, y) \leq q\}$, for rational $q > 0$. This suggests that one should look for a generalization to uniform spaces. We would first need a suitable constructive treatment of uniform spaces and their completions.

Another direction which might be worth investigating follows the work of Blanck et al. [9] who formulated general results about stability of effective algebras in type 1 effectivity. Their theorems do not translate into our settings easily, because they assume a structure which is not metric, but rather like that of sequential or limit spaces. Again, to incorporate such results we would require a constructive theory of limit spaces and their completions.

References

1. Moschovakis, Y.N.: Notation systems and recursive ordered fields. Compositio Mathematica **17** (1965) 40–71
2. Hertling, P.: A real number structure that is effectively categorical. Mathematical Logic Quarterly **45**(2) (1999) 147–182
3. Bauer, A.: The Realizability Approach to Computable Analysis and Topology. PhD thesis, Carnegie Mellon University (2000) Technical report CMU-CS-00-164.
4. van Oosten, J.: Realizability: An Introduction to its Categorical Side. Volume 152 of Studies in Logic and the Foundations of Mathematics. Elsevier (2008)
5. Weihrauch, K.: The computable multi-functions on multi-represented sets are closed under programming. J. UCS **14**(6) (2008) 801–844
6. Richman, F.: Real numbers and other completions. Mathematical Logic Quarterly **54** (2008) 98–108
7. Mal'cev, A.I.: Constructive algebras. I. Uspehi Mat. Nauk **16**(3 (99)) (1961) 3–60
8. Mal'cev, A.I.: The Metamathematics of Algebraic Systems, Collected Papers: 1936–1967. North-Holland (1971) Translated by Wells, B. F.
9. Blanck, J., Stoltenberg-Hansen, V., Tucker, J.V.: Stability for effective algebras. Electronic Notes in Theoretical Computer Science **221** (2008) 3 – 15 Proceedings of the Fifth International Conference on Computability and Complexity in Analysis (CCA 2008).

Realisability and Adequacy for (Co)induction

Ulrich Berger

Swansea University, Swansea, SA2 8PP, Wales UK
u.berger@swansea.ac.uk

Abstract. We prove the correctness of a formalised realisability interpretation of extensions of first-order theories by inductive and coinductive definitions in an untyped λ-calculus with fixed-points. We illustrate the use of this interpretation for program extraction by some simple examples in the area of exact real number computation and hint at further non-trivial applications in computable analysis.

1 Introduction

This paper studies a realisability interpretation of an extension of first-order predicate logic by least and greatest fixed points of strictly positive operators. The main results are the *Soundness Theorem* for this interpretation and the *Adequacy Theorem* for the realisers with respect to a call-by-name operational semantics and a domain-theoretic denotational semantics. Both results together imply the *Program Extraction Theorem* stating that from a constructive proof one can extract a program that is *provably correct* and *terminating*.

In order to get a flavour of the system we discuss some examples within the first-order theory of real closed fields with the real numbers as intended model. In the first example we define a set $\mathbb{N}$ of real numbers (inductively) as the *least* set satisfying

$$\mathbb{N}(0) \wedge \forall x\,(\mathbb{N}(x) \to \mathbb{N}(x+1))$$

More formally, $\mathbb{N} := \mu X.\{x \mid x = 0 \vee \exists y\,(x = y+1 \wedge X(y))\}$, i.e. $\mathbb{N}$ is the *least fixed point* of the operator mapping a set X to the set $\{x \mid x = 0 \vee \exists y\,(x = y+1 \wedge X(y))\}$. Clearly, in the intended model $\mathbb{N}$ is the set of natural numbers.

For the second example, set $\mathbb{I} := [-1,1] = \{x \mid -1 \le x \le 1\}$, $\mathrm{SD} := \{0,1,-1\}$, and $\mathrm{av}_i(x) := (x+i)/2$. Define C_0 (coinductively) as the *largest* set of real numbers satisfying

$$\forall x\,(\mathrm{C}_0(x) \to \exists i \in \mathrm{SD}, y \in \mathbb{I}\,(x = \mathrm{av}_i(y) \wedge \mathrm{C}_0(y)))$$

Formally, $\mathrm{C}_0 := \nu X.\{x \mid \exists i \in \mathrm{SD}, y \in \mathbb{I}\,(x = \mathrm{av}_i(y) \wedge X(y))\}$, i.e. C_0 is the *greatest fixed point* of the operator mapping X to $\{x \mid \exists i \in \mathrm{SD}, y \in \mathbb{I}\,(x = \mathrm{av}_i(y) \wedge X(y))\}$. Classically, one easily shows that $\mathrm{C}_0 = \mathbb{I}$. Hence the coinductive definition seems to be unnecessary. However, the point is that in order to prove *constructively* $\mathrm{C}_0(x)$ for $x \in \mathbb{I}$, one needs the extra assumption that there is a rational Cauchy sequence converging to x, and the (coinductive) proof gives us

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 49-60
http://drops.dagstuhl.de/opus/volltexte/2009/2258

a (coiterative) *program* transforming the Cauchy sequence into a signed digit representation of x.

Our third example extends the previous one to unary functions. Add a new sort for real functions, and let $\mathbb{I}^{\mathbb{I}}$ denote the set of real functions mapping $\mathbb{I}$ to $\mathbb{I}$. Define a set of real functions by

$$\mathrm{C}_1 := \nu F.\mu G.\{g \mid \exists i \in \mathrm{SD}, f \in \mathbb{I}^{\mathbb{I}}\,(g = \mathrm{av}_i \circ f \wedge F(f)) \vee \forall i \in \mathrm{SD}\, G(g \circ \mathrm{av}_i)\}$$

One can show that C_1 coincides with the set of functions in $\mathbb{I}^{\mathbb{I}}$ that are (constructively) uniformly continuous on $\mathbb{I}$. Moreover, a constructive proof of $\mathrm{C}_1(f)$ contains a program that implements f as a non-wellfounded tree acting as a (signed digit) stream transformer similar to the structures studied by Ghani, Hancock and Pattinson [GHP06]. More precisely, this interpretation is the computational content of a constructive proof the formula $\forall f\,(\mathrm{C}_1(f) \to \forall x\,(\mathrm{C}_0(x) \to \mathrm{C}_0(f(x))))$, which is a special case of a constructive composition theorem for analogous predicates C_n of n-ary functions. Details as well as concrete applications with extracted Haskell programs will be worked in a forthcoming publication.

The realisability interpretation we are going to study is related to interpretations given by Tatsuta [Tat98] and Miranda-Perea [MP05]. We try to point out the main similarities and differences. Like Tatsuta, we use *untyped* programs as realisers that allow for unrestricted recursion. The necessary termination proof for extracted programs (which seems to be missing in Tatsuta's paper) is obtained by a general Adequacy Theorem relating the operational with a (domain-theoretic) denotational semantics. Miranda extracts typed terms and uses the more general "Mendler-style" (co)inductive definitions [Men91] which extract strongly normalising terms in extensions of the second-order polymorphic λ-calculus or stronger systems [Mat01,AMU05]. Tatsuta studies realisability with truth while we omit the "truth" component. From a practical point of view the most important difference to Tatsuta's interpretation is that we treat quantifiers uniformly in the realisability interpretation (as Miranda-Perea does): $M\,\mathbf{r}\,\forall x\,A(x)$ is defined as $\forall x\,(M\,\mathbf{r}\,A(x))$, but not $\forall x\,(M\,x\,\mathbf{r}\,A(x))$, and $M\,\mathbf{r}\,\exists x\,A(x)$ is defined as $\exists x\,(M\,\mathbf{r}\,A(x))$, but not $\pi_2(M)\,\mathbf{r}\,A(\pi_1(M))$. In general, a realiser never depends on variables of the object language and does not produce output in that language, i.e. the object language and the language of realisers are kept strictly separate. Realisers are extracted exclusively from the "propositional skeleton" of a proof ignoring the first-order part which matters for the *correctness* of the realisers only. This widens the scope of applications because it is now possible to deal with abstract structures that are not necessarily "constructively" given. For example the real numbers in our examples above, were treated abstractly (i.e. axiomatically) without assuming them to be constructed in a particular way. The ignorance w.r.t. the first-order part can also be seen as a special case of the interpretations studied by Schwichtenberg [Sch09] and Hernest and Oliva [HO08] which allow for a fine control of the amount of computational information extracted from proofs.

We state most of the results without proof. Full proofs will be given in an extended version of this paper.

2 Induction and coinduction

We fix a first-order language $\mathcal{L}$. *Terms*, $r, s, t \ldots$, are built from constants, first-order variables and function symbols as usual. *Formulas*, $A, B, C \ldots$, are $s = t$, $\mathcal{P}(\boldsymbol{t})$ where $\mathcal{P}$ is a predicate (predicates are defined below), $A \wedge B$, $A \vee B$, $A \to B$, $\forall x\, A$, $\exists x\, A$. A *predicate* is either a predicate constant P, or a predicate variable X, or a comprehension term $\lambda \boldsymbol{x}.A$ (sometimes also written $\{\boldsymbol{x} \mid A\}$) where A is a formula and $\boldsymbol{x}$ is a vector of first-order variables, or an inductive predicate $\mu X.\mathcal{P}$, or a coinductive predicate $\nu X.\mathcal{P}$ where $\mathcal{P}$ is a predicate of the same arity as the predicate variable X and which is *strictly positive* in X, i.e. X does not occur free in any premise of a subformula of $\mathcal{P}$ which is an implication. The application, $\mathcal{P}(\boldsymbol{t})$, of a predicate $\mathcal{P}$ to a list of terms $\boldsymbol{t}$ is a primitive syntactic construct, except when $\mathcal{P}$ is a comprehension term, $\mathcal{P} = \{\boldsymbol{x} \mid A\}$, in which case $\mathcal{P}(\boldsymbol{t})$ stands for $A[\boldsymbol{t}/\boldsymbol{x}]$.

It will sometimes be convenient to write $\boldsymbol{x} \in \mathcal{P}$ instead of $\mathcal{P}(\boldsymbol{x})$ and also $\mathcal{P} \subseteq \mathcal{Q}$ for $\forall \boldsymbol{x}\, (\mathcal{P}(\boldsymbol{x}) \to \mathcal{Q}(\boldsymbol{x}))$ and $\mathcal{P} \cap \mathcal{Q}$ for $\{\boldsymbol{x} \mid \mathcal{P}(\boldsymbol{x}) \wedge \mathcal{Q}(\boldsymbol{x})\}$, etc. We also write $\{t \mid A\}$ as an abbreviation for $\{x \mid \exists \boldsymbol{y}\, (x = t \wedge A)\}$ where x is a fresh variable and $\boldsymbol{y} = \mathrm{FV}(t) \cap \mathrm{FV}(A)$. Furthermore, we introduce *operators* $\Phi := \lambda X.\mathcal{P}$ (or $\Phi(X) := \mathcal{P}$), where $\mathcal{P}$ is strictly positive in X, and then write $\Phi(\mathcal{Q})$ for the predicate $\mathcal{P}[\mathcal{Q}/X]$ where the latter is the usual substitution of the predicate $\mathcal{Q}$ for the predicate variable X. We also write $\mu\Phi$ and $\nu\Phi$ for $\mu X.\mathcal{P}$ and $\nu X.\mathcal{P}$. For convenience, we also write $A(X)$ to distinguish a particular predicate variable X in A, and $A(\mathcal{P})$ for the substitution of every free occurrence of X in A by $\mathcal{P}$. A formula, predicate, or operator is called *non-computational*, if it contains neither free predicate variables nor the propositional connective $\vee$. Otherwise it is called *computational*.

The *proof rules* are the usual ones for intuitionistic predicate calculus with equality. In addition, we have the axioms

$$\begin{array}{lll} \text{Closure} & \Phi(\mu\Phi) \subseteq \mu\Phi & \qquad \text{Induction} \quad \Phi(\mathcal{Q}) \subseteq \mathcal{Q} \to \mu\Phi \subseteq \mathcal{Q} \\ \text{Coclosure} & \nu\Phi \subseteq \Phi(\nu\Phi) & \qquad \text{Coinduction} \quad \mathcal{Q} \subseteq \Phi(\mathcal{Q}) \to \mathcal{Q} \subseteq \nu\Phi \end{array}$$

In addition we allow any axioms expressible by non-computational formulas that hold in the intended model. In particular, it is possible to add all classical non-computational tautologies as axioms such as, for example, $\exists x\, A \leftrightarrow \neg\forall x\, \neg A$ for non-computational A. We write $\Gamma \vdash A$ if A is derivable from assumptions in Γ in this system. If A is derivable without assumptions we write $\vdash A$, or even just A. We define falsity as $\bot := \mu X.X$ where X is a 0-ary predicate variable (i.e. a propositional variable). From the induction axiom for $\bot$ it follows immediately $\bot \to A$ for every formula A. The following basic facts are easy to prove.

Lemma 1. *(a) If $\Gamma(X) \vdash A(X)$, then $\Gamma(\mathcal{P}) \vdash A(\mathcal{P})$.*
(b) If $\Gamma \vdash \Phi(X) \subseteq \Psi(X)$, then $\Gamma \vdash \mu\Phi \subseteq \mu\Psi$ and $\Gamma \vdash \nu\Phi \subseteq \nu\Psi$.
(c) $\mathcal{P} \subseteq \mathcal{Q} \to \Phi(\mathcal{P}) \subseteq \Phi(\mathcal{Q})$.
(d) $\Phi(\mu\Phi) = \mu\Phi$ and $\Phi(\nu\Phi) = \nu\Phi$.

3 Realisability

The realisers of formulas are terms of a LISP-like untyped λ-calculus with pairing, injections and recursion (which in Sect. 5 will however receive a call-by-name operational semantics). *Program-terms*, $M, N, K, L, R \ldots$ (*terms* for short) are variables $x, y, z, \ldots$, the constant $()$, and the composite terms $\langle M, N\rangle$, $\mathrm{inl}(M)$, $\mathrm{inr}(M)$, $\lambda x.M$, $\pi_i(M)$ $(i = 1, 2)$, $\mathrm{case}\, M\, \mathrm{of}\{\mathrm{inl}(x) \to L\,;\, \mathrm{inr}(y) \to R\}$, $(M\, N)$, $\mathrm{rec}\, x\,.\, M$. The *free variables* of a term are defined as usual (the constructs λx, $\mathrm{rec}\, x$ and $\mathrm{inl}(x) \to$, $\mathrm{inr}(x) \to$ in a case term bind the variable x). The usual conventions concerning bound variables apply.

Of particular interest are closed terms that are built exclusively from $()$ by pairing $\langle \cdot, \cdot \rangle$ and the injections $\mathrm{inl}(\cdot)$ $\mathrm{inr}(\cdot)$. We call these terms *data* and denote them by $d, e, \ldots$. Roughly speaking, data stand for themselves and will in any reasonable denotational semantics coincide with their value. In Section 5 we will study such a denotational and also an operational semantics for arbitrary program terms and prove an Adequacy Theorem.

In order to formalise realisability we need a system that can talk about mathematical objects *and* realisers. Therefore we extend our first-order language $\mathcal{L}$ to a language $\mathbf{r}(\mathcal{L})$ by adding a new sort for program terms. All logical operations, including inductive and coinductive definitions, are extended as well. All axioms and rules for $\mathcal{L}$, including closure, induction, coclosure and coinduction and the rules for equality, are extended mutatis mutandis for $\mathbf{r}(\mathcal{L})$. In addition, we have as extra axioms the equations

$$\mathrm{case}\,\mathrm{inl}(M)\,\mathrm{of}\{\mathrm{inl}(x) \to L\,;\, \mathrm{inr}(y) \to R\} = L[M/x] \quad \text{similarly for } \mathrm{inr}(M),$$
$$\pi_i(\langle M_1, M_2\rangle) = M_i, \quad (\lambda x.M)N = M[N/x], \quad \mathrm{rec}\, x\,.\, M = M[\mathrm{rec}\, x\,.\, M/x]$$

The realisability interpretation assigns to every $\mathcal{L}$-formula A a unary $\mathbf{r}(\mathcal{L})$-predicate $\mathbf{r}(A)$. Intuitively, for any program term M the $\mathbf{r}(\mathcal{L})$-formula $\mathbf{r}(A)(M)$ (sometimes also written $M\,\mathbf{r}\,A$) states that M "realises" A. The definition of $\mathbf{r}(A)$ is relative to a fixed one-to-one mapping from $\mathcal{L}$-predicate variables X to $\mathbf{r}(\mathcal{L})$-predicate variables $\tilde{X}$ with one extra argument place for program terms. The definition of $\mathbf{r}(A)$ is such that if the formula A has the free predicate variables $X_1, \ldots, X_n$, then the predicate $\mathbf{r}(A)$ has the free predicate variables $\tilde{X}_1, \ldots, \tilde{X}_n$. Simultaneously with $\mathbf{r}(A)$ we define a predicate $\mathbf{r}(\mathcal{P})$ for every predicate $\mathcal{P}$, where $\mathbf{r}(\mathcal{P})$ has one extra argument place for program terms. In the definitions we take special care of non-computational formulas and predicates in order to get optimised realisers. If A is non-computational, then $\mathbf{r}(A) = \{() \mid A\}$. If $\mathcal{P}$ is non-computational, then $\mathbf{r}(\mathcal{P}) = \{((), \boldsymbol{x}) \mid \mathcal{P}(\boldsymbol{x})\}$. In all other cases:

$$\begin{aligned}
\mathbf{r}(\mathcal{P}(\boldsymbol{t})) &= \{x \mid \mathbf{r}(\mathcal{P})(x, \boldsymbol{t})\} & \mathbf{r}(A \to B) &= \{f \mid f(\mathbf{r}(A)) \subseteq \mathbf{r}(B)\}\\
\mathbf{r}(A \vee B) &= \mathrm{inl}(\mathbf{r}(A)) \cup \mathrm{inl}(\mathbf{r}(B)) & \mathbf{r}(A \wedge B) &= \langle \mathbf{r}(A), \mathbf{r}(B)\rangle\\
\mathbf{r}(\exists y\, A) &= \{x \mid \exists y\, (\mathbf{r}(A)(x))\} & \mathbf{r}(\forall y\, A) &= \{x \mid \forall y\, (\mathbf{r}(A)(x))\}\\
\mathbf{r}(X) &= \tilde{X} & \mathbf{r}(\{\boldsymbol{x} \mid A\}) &= \{(y, \boldsymbol{x}) \mid \mathbf{r}(A)(\boldsymbol{x})\}\\
\mathbf{r}(\mu X.\mathcal{P}) &= \mu\tilde{X}.\mathbf{r}(\mathcal{P}) & \mathbf{r}(\nu X.\mathcal{P}) &= \nu\tilde{X}.\mathbf{r}(\mathcal{P})
\end{aligned}$$

If one uses for operators $\Phi = \lambda X.\mathcal{P}$ the notation $\mathbf{r}(\Phi) := \lambda\tilde{X}.\mathbf{r}(\mathcal{P})$ one can shorten the last two clauses to $\mathbf{r}(\mu\Phi) = \mu\mathbf{r}(\Phi)$ and $\mathbf{r}(\nu\Phi) = \nu\mathbf{r}(\Phi)$.

We call a $\mathcal{L}$-formula a *data formula* if it contains no free predicate variables and every subformula which is an implication or of the form $\nu\Phi(\boldsymbol{t})$ is non-computational. We also define inductively a unary predicate Data by

$$\mathrm{Data} = \{()\} \cup \mathrm{inl}(\mathrm{Data}) \cup \mathrm{inr}(\mathrm{Data}) \cup \langle \mathrm{Data}, \mathrm{Data}\rangle$$

Lemma 2 (Data formulas). $\mathbf{r}(A) \subseteq \mathrm{Data}$ *for every data formula* A.

Proof. One shows more generally: if A is a formula such that every subformula which is an implication or of the form $\nu\Phi(\boldsymbol{t})$ is non-computational, then $\mathbf{r}(A)(\mathbf{Data}') \subseteq \mathrm{Data}$, where $\mathbf{r}(A)(\mathbf{Data}')$ is obtained from $\mathbf{r}(A)$ by replacing every $n+1$-ary $\mathbf{r}(\mathcal{L})$-predicate variable $\tilde{X}$ by the predicate $\mathrm{Data}' := \{(x, \boldsymbol{y}) \mid \mathrm{Data}(x)\}$. The easy proof is by induction on the structure of A.

Theorem 1 (Soundness). *From a closed derivation of a formula* A *one can extract a program term* M *and a derivation of* $\mathbf{r}(A)(M)$.

We prove the Soundness Theorem in the next chapter.

Let us see what we get when we apply realisability to our examples from the Introduction. In the first example, $\mathbf{r}(\mathbb{N})$ is the least relation such that

$$\mathbf{r}(\mathbb{N}) = \{(\mathrm{inl}(()), 0)\} \cup \{(\mathrm{inr}(n), x+1) \mid \mathbf{r}(\mathbb{N})(n, x)\}$$

Hence, we have for a data d and $x \in \mathbb{R}$ that $d\,\mathbf{r}\,\mathbb{N}(x)$ holds iff x is a natural number and $d = \underline{x} := \mathrm{inr}^x(\mathrm{inl}(()))$, i.e. d is a unary representation of x.

In the second example we first note that the formula $\mathrm{SD}(i)$ is shorthand for the formula $i = 0 \vee i = 1 \vee i = -1$. Hence for suitable data d_i ($i \in \mathrm{SD}$) we have that $\mathbf{r}(\mathrm{C}_0)$ is the largest predicate such that

$$\mathbf{r}(\mathrm{C}_0) = \{(\langle d_i, a\rangle, \mathrm{av}_i(y)) \mid i \in \mathrm{SD},\, y \in \mathbb{I},\, \mathbf{r}(\mathrm{C}_0)(a, y)\}$$

Hence, semantically, $\mathbf{r}(\mathrm{C}_0)(a, y)$ means that $a = a_0, a_1, \ldots$ is an infinite stream of digits $a_i \in \{0, 1, -1\}$ such that $y = \Sigma_{i=0}^{\infty} 2^{-(i+1)} * a_i$.

In the third example we have

$$\begin{aligned}\mathbf{r}(\mathrm{C}_1) = \nu\tilde{F}\,.\,\mu\tilde{G}\,.\;\; &\{(\langle d_i, t\rangle, \mathrm{av}_i \circ f) \mid i \in \mathrm{SD},\, f \in \mathbb{I}^{\mathbb{I}},\, \tilde{F}(t, f)\} \cup \\ &\{(\langle t_0, t_1, t_{-1}\rangle, f) \mid \forall i \in \mathrm{SD}\, \tilde{G}(t_i, g \circ \mathrm{av}_i)\}\end{aligned}$$

One sees that a realiser of $\mathrm{C}_1(f)$ is a non-wellfounded tree with two kinds of nodes: "writing nodes" labelled with (a representation of) a signed digit, which means the algorithm writes that digit to the output without reading the input stream, and "reading nodes" where the tree branches into three subtrees meaning that the algorithm reads the first digit of the input stream and continuous with the branch corresponding to the digit read and the tail of the input stream. Due to the inner "$\mu\tilde{G}$" infinitely many writing nodes occur on each path through the tree ensuring that in the limit an infinite output stream is produced.

4 Proof of the Soundness Theorem

The main task in proving the Soundness Theorem (Thm. 1) is to define the realisers of induction and coinduction and to prove their correctness.

We define program terms $\mathbf{map}_{X,A}$, $\mathbf{map}_{X,\mathcal{P}}$, $\mathbf{It}_{\mathrm{fix}\,X\,.\,\mathcal{P}}$, and $\mathbf{Coit}_{\mathrm{fix}\,X\,.\,\mathcal{P}}$, where X is a predicate variable, A is formula and $\mathcal{P}$ is a predicate, both strictly positive in X. In Lemma 6 we will show that $\mathbf{map}_{X,\mathcal{P}}$ realises the monotonicity of $\mathcal{P}$ w.r.t. X. The terms $\mathbf{It}_{\mathrm{fix}\,X\,.\,\mathcal{P}}$ and $\mathbf{Coit}_{\mathrm{fix}\,X\,.\,\mathcal{P}}$ will be used to realise induction and coinduction. In [MP05] the iterators and coiterators are given as constants which expect map-terms as extra arguments, and the property stated in Lemma 6 is an assumption in the Soundness Theorem.

Here, the terms $\mathbf{map}_{X,A}$, $\mathbf{map}_{X,\mathcal{P}}$, $\mathbf{It}_{\mathrm{fix}\,X\,.\,\mathcal{P}}$, and $\mathbf{Coit}_{\mathrm{fix}\,X\,.\,\mathcal{P}}$ are defined by recursion on the structure of A and $\mathcal{P}$. We write $M \circ N$ as an abbreviation for $\lambda x.M(N\,x)$ where x is fresh. $\mathbf{map}_{X,A} = \mathbf{map}_{X,\mathcal{P}} = \lambda f \lambda x\,.\,x$ if X is not free in A or $\mathcal{P}$. Otherwise:

$$
\begin{aligned}
\mathbf{map}_{X,\mathcal{P}(\boldsymbol{t})} &= \mathbf{map}_{X,\mathcal{P}}\\
\mathbf{map}_{X,A\wedge B} &= \lambda f \lambda x\,.\,\langle \mathbf{map}_{X,A} f\,(\pi_1(x)), \mathbf{map}_{X,B} f\,(\pi_2(x))\rangle\\
\mathbf{map}_{X,A\vee B} &= \lambda f \lambda x\,.\,\mathrm{case}\,x\,\mathrm{of}\{\mathrm{inl}(y) \to \mathbf{map}_{X,A}\,f\,y\,;\,\mathrm{inr}(z) \to \mathbf{map}_{X,B}\,f\,z\}\\
\mathbf{map}_{X,A\to B} &= \lambda f \lambda g\,.\,\mathbf{map}_{X,B}\,f \circ g\\
\mathbf{map}_{X,\{\boldsymbol{x}|A\}} &= \mathbf{map}_{X,A}\\
\mathbf{map}_{X,X} &= \lambda f\,.\,f\\
\mathbf{map}_{X,\mu Y.\mathcal{P}} &= \lambda f\,.\,\mathbf{It}_{\mathrm{fix}\,Y\,.\,\mathcal{P}}(\mathbf{map}_{X,\mathcal{P}} f)\\
\mathbf{map}_{X,\nu Y.\mathcal{P}} &= \lambda f\,.\,\mathbf{Coit}_{\mathrm{fix}\,Y\,.\,\mathcal{P}}(\mathbf{map}_{X,\mathcal{P}} f)\\
\mathbf{It}_{\mathrm{fix}\,X\,.\,\mathcal{P}} &= \lambda s\,.\,\mathrm{rec}\,g\,.\,s \circ \mathbf{map}_{X,\mathcal{P}} g\\
\mathbf{Coit}_{\mathrm{fix}\,X\,.\,\mathcal{P}} &= \lambda s\,.\,\mathrm{rec}\,g\,.\,\mathbf{map}_{X,\mathcal{P}} g \circ s
\end{aligned}
$$

Lemma 3. *(a)* $\mathbf{It}_{\mathrm{fix}\,X\,.\,\mathcal{P}} s = s \circ \mathbf{map}_{X,\mathcal{P}}(\mathbf{It}_{\mathrm{fix}\,X\,.\,\mathcal{P}} s)$
(b) $\mathbf{Coit}_{\mathrm{fix}\,X\,.\,\mathcal{P}} s = \mathbf{map}_{X,\mathcal{P}}(\mathbf{Coit}_{\mathrm{fix}\,X\,.\,\mathcal{P}} s) \circ s$
(c) $\mathbf{map}_{X,\mu Y.\mathcal{P}} g = \mathbf{map}_{X,\mathcal{P}} g \circ \mathbf{map}_{Y,\mathcal{P}}(\mathbf{map}_{X,\mu Y.\mathcal{P}} g)$
(d) $\mathbf{map}_{X,\nu Y.\mathcal{P}} g = \mathbf{map}_{Y,\mathcal{P}}(\mathbf{map}_{X,\nu Y.\mathcal{P}} g) \circ \mathbf{map}_{X,\mathcal{P}} g$

Proof. Easy calculation using the equational axioms for the calculus.

Lemma 4 (Substitution). $\mathbf{r}(\Phi)(\mathbf{r}(\mathcal{Q})) = \mathbf{r}(\Phi(\mathcal{Q}))$ *for every operator* Φ *and predicate* $\mathcal{Q}$*.*

Proof. Straightforward induction on the (syntactic) size of Φ.

In the next lemmas we consider predicates in the language $\mathbf{r}(\mathcal{L})$ whose first arguments range over predicate terms. The following definitions will be used:

$$\mathcal{P} \circ f := \{(x, \boldsymbol{y}) \mid (f\,x, \boldsymbol{y}) \in \mathcal{P}\} \qquad f * \mathcal{P} := \{(f\,x, \boldsymbol{y}) \mid (x, \boldsymbol{y}) \in \mathcal{P}\}$$

Clearly, $(\mathcal{P} \circ f) \circ g = \mathcal{P} \circ (f \circ g)$ and $f * (g * \mathcal{P}) = (f * g) * \mathcal{P}$. The rationale for the first of the two definitions is that $\mathbf{r}(\mathcal{P} \subseteq \mathcal{Q}) = \{f \mid \mathbf{r}(\mathcal{P}) \subseteq \mathbf{r}(\mathcal{Q}) \circ f\}$.

and the Induction Axiom is an implication between inclusions of predicates. The following easy lemma shows that the two definitions are adjoints. This will allow us to treat induction and coinduction in a similar way.

Lemma 5 (Adjunction). $\mathcal{Q} \subseteq \mathcal{P} \circ f \;\Leftrightarrow\; f * \mathcal{Q} \subseteq \mathcal{P}$

Lemma 6 (Map). *Let Φ be an operator in the language $\mathcal{L}$. and X a fresh predicate variable. Then $\mathbf{map}_{X,\Phi(X)}$ realises the monotonicity of Φ, that is*

$$\mathbf{map}_{X,\Phi(X)}\,\mathbf{r}\,(\mathcal{P} \subseteq \mathcal{Q} \to \Phi(\mathcal{P}) \subseteq \Phi(\mathcal{Q}))$$

for all $\mathcal{L}$-predicates $\mathcal{P}$ and $\mathcal{Q}$. By the definition of realisability and the Adjunction Lemma this is equivalent to each of the following two statements about arbitrary $\mathbf{r}(\mathcal{L})$-predicates $\mathcal{P}$ and $\mathcal{Q}$ of appropriate arity and all f:

(a) $\mathcal{P} \subseteq \mathcal{Q} \circ f \to \mathbf{r}(\Phi)(\mathcal{P}) \subseteq \mathbf{r}(\Phi)(\mathcal{Q}) \circ \mathbf{map}_{X,\Phi(X)} f$
(b) $f * \mathcal{P} \subseteq \mathcal{Q} \to \mathbf{map}_{X,\Phi(X)} f * \mathbf{r}(\Phi)(\mathcal{P}) \subseteq \mathbf{r}(\Phi)(\mathcal{Q})$

Furthermore, setting in (a) $\mathcal{P} := \mathcal{Q} \circ f$ and in (b) $\mathcal{Q} := f * \mathcal{P}$ one obtains

(c) $\mathbf{r}(\Phi)(\mathcal{Q} \circ f) \subseteq \mathbf{r}(\Phi)(\mathcal{Q}) \circ \mathbf{map}_{X,\Phi(X)} f$
(d) $\mathbf{map}_{X,\Phi(X)} f * \mathbf{r}(\Phi)(\mathcal{P}) \subseteq \mathbf{r}(\Phi)(f * \mathcal{P})$

Proof. We show a slight generalisation of (a). Let Φ be an operator of $n+1$ arguments, and $X, \boldsymbol{Y}$ fresh predicate variables. Let $\boldsymbol{\mathcal{Q}} = \mathcal{Q}_1, \ldots, \mathcal{Q}_n$ be predicates in the language $\mathbf{r}(\mathcal{L})$. Then for all f, $\mathcal{P}$, $\mathcal{Q}$

$$\mathcal{P} \subseteq \mathcal{Q} \circ f \to \mathbf{r}(\Phi)(\mathcal{P}, \boldsymbol{\mathcal{Q}}) \subseteq \mathbf{r}(\Phi)(\mathcal{Q}, \boldsymbol{\mathcal{Q}}) \circ \mathbf{map}_{X,\Phi(X)} f$$

The proof is by induction on the structure of $\Phi(X, \boldsymbol{Y})$. In the proof we allow ourselves to switch between (a) and (b) whenever convenient. We only carry out in detail the difficult cases, namely when Φ is defined by induction or coinduction.

Case $\Phi(X, \boldsymbol{Y}) = \mu Z.\Phi_0(X, \boldsymbol{Y}, Z)$. Then $\mathbf{r}(\Phi)(\tilde{X}, \tilde{\boldsymbol{Y}}) = \mu\tilde{Z}.\mathbf{r}(\Phi_0)(\tilde{X}, \tilde{\boldsymbol{Y}}, \tilde{Z})$. Assume $\mathcal{P} \subseteq \mathcal{Q} \circ f$. Setting $\mathcal{R} := \mathbf{r}(\Phi)(\mathcal{Q}, \boldsymbol{\mathcal{Q}}) = \mu\tilde{Z}.\mathbf{r}(\Phi_0)(\mathcal{Q}, \boldsymbol{\mathcal{Q}}, \tilde{Z})$, we have to show $\mu\tilde{Z}.\mathbf{r}(\Phi_0)(\mathcal{P}, \boldsymbol{\mathcal{Q}}, \tilde{Z}) \subseteq \mathcal{R} \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f$. We induct on $\mu\tilde{Z}.\mathbf{r}(\Phi_0)(\mathcal{P}, \boldsymbol{\mathcal{Q}}, \tilde{Z})$. Hence, we have to show $\mathbf{r}(\Phi_0)(\mathcal{P}, \boldsymbol{\mathcal{Q}}, \mathcal{R} \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f) \subseteq \mathcal{R} \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f$.

$$\begin{array}{rl}
 & \mathbf{r}(\Phi_0)(\mathcal{P}, \boldsymbol{\mathcal{Q}}, \mathcal{R} \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f) \\
\overset{\text{i.h.(c)}}{\subseteq} & \mathbf{r}(\Phi_0)(\mathcal{P}, \boldsymbol{\mathcal{Q}}, \mathcal{R}) \circ \mathbf{map}_{Z,\Phi_0(X,\boldsymbol{Y},Z)}(\mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f) \\
\overset{\text{i.h.(a)}}{\subseteq} & \mathbf{r}(\Phi_0)(\mathcal{Q}, \boldsymbol{\mathcal{Q}}, \mathcal{R}) \circ \mathbf{map}_{X,\Phi_0(X,\boldsymbol{Y},Z)} f \circ \mathbf{map}_{Z,\Phi_0(X,\boldsymbol{Y},Z)}(\mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f) \\
\overset{\text{L. 3 (c)}}{=} & \mathbf{r}(\Phi_0)(\mathcal{Q}, \boldsymbol{\mathcal{Q}}, \mathcal{R}) \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f \\
= & \mathbf{r}(\Phi_0)(\mathcal{Q}, \boldsymbol{\mathcal{Q}}, \mu\tilde{Z}.\mathbf{r}(\Phi_0)(\mathcal{P}, \boldsymbol{\mathcal{Q}}, \tilde{Z})) \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f \\
\overset{\text{Fixed P.}}{=} & \mu\tilde{Z}.\mathbf{r}(\Phi_0)(\mathcal{Q}, \boldsymbol{\mathcal{Q}}, \tilde{Z}) \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f \\
= & \mathcal{R} \circ \mathbf{map}_{X,\Phi(X,\boldsymbol{Y})} f
\end{array}$$

Case $\Phi(X,\boldsymbol{Y}) = \nu Z.\Phi_0(X,\boldsymbol{Y},Z)$. Then $\mathbf{r}(\Phi)(\tilde{X},\tilde{\boldsymbol{Y}}) = \nu\tilde{Z}.\mathbf{r}(\Phi_0)(\tilde{X},\tilde{\boldsymbol{Y}},\tilde{Z})$. Obviously, it is now more convenient to show (b). Assume $f{*}\mathcal{P} \subseteq \mathcal{Q}$. Setting $\mathcal{R} := \mathbf{r}(\Phi)(\mathcal{P},\boldsymbol{\mathcal{Q}}) = \nu\tilde{Z}.\mathbf{r}(\Phi_0)(\mathcal{P},\boldsymbol{\mathcal{Q}},\tilde{Z})$ we use coinduction to show $\mathbf{map}_{X,\Phi(X,\boldsymbol{Y})}f * \mathcal{R} \subseteq \nu\tilde{Z}.\mathbf{r}(\Phi_0)(\mathcal{Q},\boldsymbol{\mathcal{Q}},\tilde{Z})$. The proof is exactly dual to the inductive proof above (using the i.h. in the form (d) and (b)).

Proof of the Soundness Theorem (Thm. 1). As usual, one shows by induction on derivations the following more general statement: From a derivation $B_1,\dots,B_n \vdash A$ one can extract a program term M with free variables among $x_1,\dots,x_n$ such that $\mathbf{r}(B_1)(x_1),\dots,\mathbf{r}(B_n)(x_n) \vdash \mathbf{r}(A)(M)$. The only interesting cases are induction and coinduction.

Induction. By the Substitution Lemma, we have

$$\mathbf{r}(\Phi(\mathcal{Q}) \subseteq \mathcal{Q} \to \mu\Phi \subseteq \mathcal{Q}) = \{f \mid \forall s\,(\mathbf{r}(\Phi)(\mathbf{r}(\mathcal{Q})) \subseteq \mathbf{r}(\mathcal{Q}){\circ}s \to \mu\mathbf{r}(\Phi) \subseteq \mathbf{r}(\mathcal{Q}){\circ}fs)\}$$

Hence, in order to show that $\mathbf{It}_{\varphi(\alpha)}$ (=: M) realises induction, we assume

$$\mathbf{r}(\Phi)(\mathbf{r}(\mathcal{Q})) \subseteq \mathbf{r}(\mathcal{Q}) \circ s$$

and show $\mu\mathbf{r}(\Phi) \subseteq \mathbf{r}(\mathcal{Q}) \circ \mathbf{It}_{\varphi(\alpha)}s$. We use induction on $\mu\mathbf{r}(\Phi)$, which reduces the problem to showing $\mathbf{r}(\Phi)(\mathbf{r}(\mathcal{Q}) \circ \mathbf{It}_{\varphi(\alpha)}s) \subseteq \mathbf{r}(\mathcal{Q}) \circ \mathbf{It}_{\varphi(\alpha)}s$.

$$\begin{array}{lcl}
\mathbf{r}(\Phi)(\mathbf{r}(\mathcal{Q}) \circ \mathbf{It}_{\varphi(\alpha)})s & \overset{\text{Map Lemma (c)}}{\subseteq} & \mathbf{r}(\Phi)(\mathbf{r}(\mathcal{Q})) \circ \mathbf{map}_{\varphi(\alpha)}(\mathbf{It}_{\varphi(\alpha)}s) \\
& \overset{\text{assumption}}{\subseteq} & \mathbf{r}(\mathcal{Q}) \circ s \circ \mathbf{map}_{\varphi(\alpha)}(\mathbf{It}_{\varphi(\alpha)}s) \\
& \overset{\text{Lemma 3 (a)}}{=} & \mathbf{r}(\mathcal{Q}) \circ \mathbf{It}_{\varphi(\alpha)}s
\end{array}$$

Coinduction. Similar, using the Map Lemma (d) and Lemma 3 (b).

5 Semantics of program terms

Now we study a call-by-name operational semantics of program terms which allows us to use the program terms extracted from a formal proof to compute data. As an intermediate step we employ a domain-theoretic *denotational* semantics. The denotational semantics is of independent interest since it directly reflects the intuitive mathematical meaning of program terms.

By a *domain* a *Scott-domain*, i.e. an algebraic, countably based, bounded complete, dcpo [GHK$^+$03]. Note that every domain has a least element $\bot$ w.r.t. the domain ordering $\sqsubseteq$. Let D be the least solution of the domain equation

$$D = \mathbf{1} + D + D + D \times D + [D \to D]$$

where $\mathbf{1}$ is the one-point domain $\{()\}$, and $+, \times, [\cdot \to \cdot]$ denote the usual domain operations, separated sum, cartesian product, and continuous function space (of course, the domain equation holds only "up to isomorphism"). Hence, every

element of D is of exactly one of the following forms: $\bot$, $()$, $\mathrm{inl}(a)$, $\mathrm{inr}(a)$, $\langle a, b\rangle$, $\mathrm{abst}(f)$, where $a, b \in D$ and $f \in [D \to D]$. It follows from standard facts in domain theory that every program term M defines in a natural way a continuous function $[\![M]\!] : D^{\mathrm{Var}} \to D$. For example, $[\![\lambda x.M]\!]\xi = \mathrm{abst}(f)$ where $f(a) = [\![M]\!]\xi[x \mapsto a]$ and $[\![\mathrm{rec}\, x\,.\, M]\!]\xi$ is the least fixed point of f. Furthermore, if $[\![M]\!]\xi = \mathrm{abst}(f)$, then $[\![M\, N]\!]\xi = f([\![N]\!]\xi)$, otherwise the result is $\bot$.

If Ax is a set of non-computational $\mathcal{L}$-axioms we denote by $\mathbf{r}(\mathrm{Ax})$ the system of $\mathbf{r}(\mathcal{L})$-axioms consisting of the axioms in Ax together with the extra axioms introduced in Sect. 3. If $\mathcal{M}$ is a model of Ax, then we denote by $\mathbf{r}(\mathcal{M})$ the obvious expansion of $\mathcal{M}$ to a model of $\mathbf{r}(\mathrm{Ax})$ using the definition above of the value of a program term. Again, it follows from standard results in domain theory that $\mathbf{r}(\mathcal{M})$ satisfies the axioms for program terms and hence is indeed a model of $\mathbf{r}(\mathrm{Ax})$. Note that in this model the interpretation of the predicate Data defined in Sect. 3 is the least subset $[\![\mathrm{Data}]\!]$ of D such that

$$[\![\mathrm{Data}]\!] = \{()\} \cup \mathrm{inl}([\![\mathrm{Data}]\!]) \cup \mathrm{inr}([\![\mathrm{Data}]\!]) \cup \langle [\![\mathrm{Data}]\!], [\![\mathrm{Data}]\!]\rangle$$

Hence, if $\mathrm{Data}(M)$ is provable, then $[\![M]\!] \in [\![\mathrm{Data}]\!]$.

Now we introduce the operational semantics of program terms. A *closure* is a pair (M, η) where M is a program term and η is an *environment*, i.e. a finite mapping from variables to closures, such that all free variables of M are in the domain of η. Note that this is an inductive definition on the meta-level. A *value* is a closure (M, η) where M is an *intro term*, i.e. a term of the form $()$, or $\mathrm{inl}(M_0)$, or $\mathrm{inr}(M_0)$, or $\langle M_1, M_2\rangle$, or $\lambda x.M_0$. We let $c, c', \ldots$ range over closures and $v, v', \ldots$ range over values. We inductively define the relation $c \longrightarrow v$ (big-step reduction):

$$v \longrightarrow v \qquad \frac{\eta(x) \longrightarrow v}{(x, \eta) \longrightarrow v}$$

$$\frac{(M, \eta) \longrightarrow (\mathrm{inl}(M_0), \eta') \qquad (L, \eta[x \mapsto (M_0, \eta')]) \longrightarrow v}{(\mathrm{case}\, M\, \mathrm{of}\{\mathrm{inl}(x) \to L\,;\, \mathrm{inr}(y) \to R\}, \eta) \longrightarrow v}\ \text{sim. inr}(M_0).$$

$$\frac{(M, \eta) \longrightarrow (\langle M_1, M_2\rangle, \eta') \qquad (M_i, \eta) \longrightarrow v}{\pi_i(M) \longrightarrow v}$$

$$\frac{(M, \eta) \longrightarrow (\lambda x.M_0, \eta') \qquad (M_0, \eta'[x \mapsto (N, \eta)]) \longrightarrow v}{(M\, N, \eta) \longrightarrow v}$$

$$\frac{(M, \eta[x \mapsto (\mathrm{rec}\, x\,.\, M, \eta)]) \longrightarrow v}{(\mathrm{rec}\, x\,.\, M, \eta) \longrightarrow v}$$

Finally, in order to compute data we need a 'print' relation $c \Longrightarrow d$ between closures c and data terms d.

$$\frac{c \longrightarrow ((), \eta)}{c \Longrightarrow ()} \qquad \frac{c \longrightarrow (\mathrm{inl}(M), \eta) \qquad (M, \eta) \Longrightarrow d}{c \Longrightarrow \mathrm{inl}(d)}\ \text{sim. inr}(M)$$

$$\frac{c \longrightarrow (\langle M_1, M_2\rangle, \eta) \qquad (M_1, \eta) \Longrightarrow d_1 \qquad (M_2, \eta) \Longrightarrow d_2}{c \Longrightarrow \langle d_1, d_2\rangle}$$

The inductive definition of $c \Longrightarrow d$ gives rise to an algorithm computing d from c in a call-by-name fashion. It follows that whenever $M \Longrightarrow d$, then in a call-by-name language such as Haskell the evaluation of the program corresponding will terminate with a result corresponding to d.

To every closure c we assign a term $\overline{c}$ by 'flattening', i.e. removing the structure provided by the nested environments: $\overline{(M,\eta)} = M[\overline{\eta(x)}/x \mid x \in \mathrm{dom}(\eta)]$.

Lemma 7 (Correctness).

(a) If $c \longrightarrow v$, then $\overline{c} = \overline{v}$ is provable.
(b) If $c \Longrightarrow d$, then $\overline{c} = d$ is provable.

Theorem 2 (Adequacy). *If $[\![M]\!] = d$, then $(M, \emptyset) \Longrightarrow d$.*

The proof of the Adequacy Theorem is uses a technique that has been used for a similar purpose in [Win93] and [CS06]. It can be viewed as transformation of Plotkin's Adequacy Theorem for PCF [Plo77] to the untyped setting. To carry out the proof, we first exploit the algebraicity of the domain D. Every element of D is the directed supremum of *compact* elements, which are generated at some finite stage in the construction of D. Let D_0 be the set of compact elements of D. There is a rank function $\mathbf{rk}(\cdot) : D_0 \to \mathbb{N}$ satisfying:

(rk1) The images of the injections $\mathrm{inl}(\cdot), \mathrm{inr}(\cdot)$, and the pairing function $\langle\cdot,\cdot\rangle$ are compact iff their arguments are. Injections and pairing increase rank.
(rk2) If $\mathrm{abst}(f)$ is compact, then for every $a \in D$, $f(a)$ is compact with $\mathbf{rk}(f(a)) < \mathbf{rk}(\mathrm{abst}(f))$, and there exists a compact $a_0 \sqsubseteq a$ with $\mathbf{rk}(a_0) < \mathbf{rk}(\mathrm{abst}(f))$ and $f(a_0) = f(a)$.

These properties allow us to define for every compact a a set $\mathbf{Cl}(a)$ of closures, by recursion on $\mathbf{rk}(a)$: $\mathbf{Cl}(\bot)$ is the set of all closures, otherwise

$$\begin{aligned}
\mathbf{Cl}(()) &= \{c \mid \exists \eta\,(c \longrightarrow ((), \eta))\}\\
\mathbf{Cl}(\mathrm{inl}(a)) &= \{c \mid \exists (M,\eta) \in \mathbf{Cl}(a)\,(c \longrightarrow (\mathrm{inl}(M), \eta))\}\\
\mathbf{Cl}(\mathrm{inr}(a)) &= \{c \mid \exists (M,\eta) \in \mathbf{Cl}(a)\,(c \longrightarrow (\mathrm{inr}(M), \eta))\}\\
\mathbf{Cl}(\langle a_1, a_2\rangle) &= \{c \mid \exists M_1, M_2, \eta\,((M_1,\eta) \in \mathbf{Cl}(a_1) \wedge (M_2,\eta) \in \mathbf{Cl}(a_2) \wedge\\
&\qquad c \longrightarrow (\langle M_1, M_2\rangle, \eta))\}\\
\mathbf{Cl}(\mathrm{abst}(f)) &= \{c \mid \exists x, M, \eta\,(c \longrightarrow (\lambda x.M, \eta) \wedge \forall a \in D_0\,(\mathbf{rk}(a) < \mathbf{rk}(\mathrm{abst}(f))\\
&\qquad \to \forall c' \in \mathbf{Cl}(a)\,(M, \eta[x \mapsto c']) \in \mathbf{Cl}(f(a))))\}
\end{aligned}$$

Alternatively, one could use Pitt's method [Pit94] to define similar "candidate" sets. Using (rk1) and (rk2) one can prove:

Lemma 8. *(a) If a, b are compact with $a \sqsubseteq b$, then $\mathbf{Cl}(a) \supseteq \mathbf{Cl}(b)$.*
(b) $c \in \mathbf{Cl}(a)$ iff there exists a value v with $c \longrightarrow v$ and $v \in \mathbf{Cl}(a)$.
(c) If $c \in \mathbf{Cl}(d)$, where d is a data, then $c \Longrightarrow d$.

In the following we write $\eta \in \mathbf{Cl}(\xi)$ if for all $x \in \mathrm{dom}(\eta)$, $\xi(x)$ is compact and $\eta(x) \in \mathbf{Cl}(\xi(x))$.

Lemma 9 (Approximation). *If $\eta \in \mathbf{Cl}(\xi)$ and a is compact with $a \sqsubseteq [\![M]\!]\xi$, then $(M, \eta) \in \mathbf{Cl}(a)$.*

Proof. Let $[\![M]\!]^n\xi$ denote the n-th stage in the definition of $[\![M]\!]\xi$. Hence, $[\![M]\!]^0\xi = \bot$ and e.g. $[\![\lambda x.M]\!]^{n+1}\xi(a) = [\![M]\!]^n\xi[\mapsto a]$, e.t.c. Since the $[\![M]\!]^n\xi$ form an increasing chain in D with $[\![M]\!]\xi$ as its supremum, it follows that if a is compact and $a \sqsubseteq [\![M]\!]\xi$, then $a \sqsubseteq [\![M]\!]^n\xi$ for some n. Hence, it is enough to show by induction on n that if $\eta \in \mathbf{Cl}(\xi)$ and a is compact with $a \sqsubseteq [\![M]\!]^n\xi$, then $(M, \eta) \in \mathbf{Cl}(a)$.

Proof of the Adequacy Theorem (Thm. 2). Assume $[\![M]\!] = d$ for some data d. Since d is compact, it follows, by the Approximation Lemma, $(M, \emptyset) \in \mathbf{Cl}(d)$. Hence $(M, \emptyset) \Longrightarrow d$, by Lemma 8 (c).

Theorem 3 (Program extraction). *From a proof of a data formula A one can extract a program term M with the property that $(M, \emptyset) \Longrightarrow d$ for some data d provably realising A, i.e. $\mathbf{r}(A)(d)$ is provable.*

Proof. By the Soundness Theorem, we obtain from a proof of A a program term M and a proof of $\mathbf{r}(A)(M)$. By Lemma 2, $\mathrm{Data}(M)$ is provable and therefore true in D, i.e. $[\![M]\!] = d$ for some data d. By the Adequacy Theorem, $(M, \emptyset) \Longrightarrow d$, and by Lemma 7, $M = d$ is provable. It follows that $\mathbf{r}(A)(d)$ is provable.

6 Conclusion and further work

In this paper we laid the logical and semantical foundations for the extraction of programs from proofs involving inductive and coinductive definitions. The main results where the *Soundness Theorem* for a realisability interpretation stating that the extracted program provably realises the proven formula, and the *Adequacy Theorem* stating that for *data formulas* the realisers can be computed into canonical form via a call-by-name operational semantics.

We restricted ourselves to simple examples illustrating the method. More substantial applications are to be published in forthcoming papers. Strictly speaking our results do not apply to loc. cit. because there realisers are typed (with Haskell or ML style polymorphic types) while our realisers are untyped. We plan to recast our results with typed realisers, which will probably technically more complicated, but will have the advantage that the category-theoretic justification of induction and coinduction can be used to "derive" the realisability interpretation. Moreover, this will allow for a direct interpretation of realisers as programs in a call-by-name typed programming language such as Haskell.

A major piece of work that remains to be done is to implement the realisability interpretation in an interactive theorem prover and carry out case studies. We expect this to tie in nicely with recent work on implementations of inductive and coinductive definitions and proofs [CDG06,Ber07], exact real arithmetic [MRE07,GNSW07,EH02,Sch09], realisability [BS07], and functional interpretation [HO08].

References

[AMU05] A. Abel, R. Matthes, and T. Uustalu. Iteration and coiteration schemes for higher-order and nested datatypes. *Theor. Comput. Sci.*, 333:3–66, 2005.

[Ber07] Y. Bertot. Affine functions and series with co-inductive real numbers. *Math. Struct. Comput. Sci.*, 17:37–63, 2007.

[BS07] A. Bauer and A.C. Stone. RZ: A tool for bringing constructive and computable mathematics closer to programming practice. In S.B. Cooper, B. Löwe, and A. Sorbi, editors, *CiE 2007*, volume 4497 of *LNCS*, pages 28–42, 2007.

[CDG06] A. Ciaffaglione and P. Di Gianantonio. A certified, corecursive implementation of exact real numbers. *Theor. Comput. Sci.*, 351:39–51, 2006.

[CS06] T. Coquand and A. Spiwack. A proof of strong normalisation using domain theory. In *LICS'06*, pages 307–316. IEEE Computer Society Press, 2006.

[EH02] A. Edalat and R. Heckmann. Computing with real numbers: I. The LFT approach to real number computation; II. A domain framework for computational geometry. In G. Barthe, P. Dybjer, L. Pinto, and J. Saraiva, editors, *Applied Semantics - Lecture Notes from the International Summer School, Caminha, Portugal*, pages 193–267. Springer, 2002.

[GHK+03] G. Gierz, K.H. Hofmann, K. Keimel, J.D. Lawson, M. Mislove, and D.S. Scott. *Continuous Lattices and Domains*, volume 93 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 2003.

[GHP06] N. Ghani, P. Hancock, and D. Pattinson. Continuous functions on final coalgebras. *Electr. Notes in Theoret. Comp. Sci.*, 164, 2006.

[GNSW07] H. Geuvers, M. Niqui, B. Spitters, and F. Wiedijk. Constructive analysis, types and exact real numbers. *Math. Struct. Comput. Sci.*, 17(1):3–36, 2007.

[HO08] M. D. Hernest and P. Oliva. Hybrid functional interpretations. In A. Beckmann, C. Dimitracopoulos, and B. Löwe, editors, *CiE 2008: Logic and Theory of Algorithms*, volume 5028 of *LNCS*, pages 251–260. Springer, 2008.

[Mat01] R. Matthes. Monotone inductive and coinductive constructors of rank 2. In L Fribourg, editor, *Computer Science Logic (Proceedings of the Fifteenth CSL Conference)*, number 2142 in LNCS, pages 600–615. Springer, 2001.

[Men91] N.P. Mendler. Inductive types and type constraints in the second-order lambda calculus. *Ann. Pure Appl. Logic*, 51:159–172, 1991.

[MP05] F. Miranda-Perea. Realizability for monotone clausular (co)inductive definitions. *Electron. Notes Theoret. Comput. Sci.*, 123:179–193, 2005.

[MRE07] J. R. Marcial-Romero and M. H. Escardo. Semantics of a sequential language for exact real-number computation. *Theor. Comput. Sci.*, 379(1-2):120–141, 2007.

[Pit94] A.M. Pitts. Computational adequacy via "mixed" inductive definitions. In *Proc. 9th International Conference on Mathematical Foundations of Programming Semantics*, pages 72–82, London, UK, 1994. Springer-Verlag.

[Plo77] G.D. Plotkin. LCF considered as a programming language. *Theor. Comput. Sci.*, 5:223–255, 1977.

[Sch09] H. Schwichtenberg. Realizability interpretation of proofs in constructive analysis. Theory Comput. Sys., to appear, 2009.

[Tat98] M. Tatsuta. Realizability of monotone coinductive definitions and its application to program synthesis. In R. Parikh, editor, *Mathematics of Program Construction*, volume 1422 of *LNM*, pages 338–364. Springer, 1998.

[Win93] G. Winskel. *The Formal Semantics of Programming Languages*. Found. of Comput. Series. The MIT Press, Cambridge, Massachusetts, 1993.

A Constructive Study of Landau's Summability Theorem

Josef Berger[1] and Douglas Bridges[2]

[1] Mathematisches Institut
Universität München
Theresienstraße 39, D-80333 München, Germany
jberger@mathematik.uni-muenchen.de
[2] (Corresponding author)
Department of Mathematics & Statistics
University of Canterbury
Private Bag 4800
Christchurch 8140, New Zealand
d.bridges@math.canterbury.ac.nz

Abstract. A summability theorem of Landau, which classically is a simple consequence of the uniform boundedness theorem, is examined constructively.

Edmund Landau (1877–1938) is known for many contributions to mathematics. In this paper we examine his summability theorem,

> *If p, q are* ***conjugate exponents****—positive integers such that $\frac{1}{p} + \frac{1}{q} = 1$—and if $\mathbf{a} = (a_n)_{n\geqslant 1}$ is a sequence in $\mathbf{C}$ such that $\sum_{n=1}^{\infty} a_n x_n$ converges for each $\mathbf{x} = (x_n)_{n\geqslant 1}$ in the Banach space l_p, then $\mathbf{a} \in l_q$,*

from the viewpoint of Bishop's constructive mathematics **(BISH)**—that is, mathematics developed with intuitionistic logic and a suitable set-theoretic foundation such as the Aczel-Rathjen-Myhill CST [1, 13].

The standard functional-analytic proof goes as follows. For each $\mathbf{x} = (x_n)_{n\geqslant 1}$ in l_p and each k define

$$s_k(\mathbf{x}) = \sum_{n=1}^{k} a_n x_n.$$

Then

$$|s_k(\mathbf{x})| \leqslant \left(\sum_{n=1}^{k} |a_n|^q\right)^{1/q} \left(\sum_{n=1}^{k} |x_p|^p\right)^{1/p} \leqslant \left(\sum_{n=1}^{k} |a_n|^q\right)^{1/q} \|\mathbf{x}\|_p,$$

from which it follows that s_k is a bounded linear functional on l_p with norm

$$\|s_k\| = \left(\sum_{n=1}^{k} |a_n|^q\right)^{1/q}.$$

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 61-70
http://drops.dagstuhl.de/opus/volltexte/2009/2259

Also, the sequence $(s_k(\mathbf{x}))_{k\geqslant 1}$ converges in $\mathbf{C}$ and so is bounded. Applying the uniform boundedness theorem to the sequence $(s_k)_{k\geqslant 1}$, we now obtain $M > 0$ such that $\|s_k\| \leqslant M$ for each k. The partial sums of the series $\sum_{n=1}^{\infty} |a_n|^q$ are therefore bounded, so the series converges in $\mathbf{R}$.

From a constructive viewpoint, there are two problems with this proof. First, the uniform boundedness theorem in the form applied there is not the constructive one. Secondly, boundedness of the partial sums of a series of positive terms is not enough to ensure its convergence (see pages 60–64 of [5]). In fact, a Brouwerian example shows that Landau's summability theorem in its classical form is not constructively valid: under its hypotheses we cannot even prove, in general, that $a_n \to 0$ as $n \to \infty$. To see this, take $\mathbf{a}$ as a binary sequence with at most one term equal to 1, and consider the case $p = q = 2$. The series $\sum_{n=1}^{\infty} a_n x_n$ certainly converges for each $\mathbf{x}$ in l_2. But if $a_n \to 0$ as $n \to \infty$, we can find N such that $a_n = 0$ for all $n > N$; by testing $a_1, \ldots, a_N$, we can decide whether $a_n = 0$ for all n or there exists n such that $a_n = 1$. Thus the statement

> For each sequence $\mathbf{a}$ of complex numbers, if $\sum_{n=1}^{\infty} a_n x_n$ converges for all $\mathbf{x} \in l_2$, then $\mathbf{a} \in l_2$

implies the essentially nonconstructive **limited principle of omniscience**,

> **LPO**: For each binary sequence $\mathbf{a}$, either $a_n = 0$ for all n or else there exists n such that $a_n = 1$.

At this stage, it remains a possibility that, under the hypotheses of Landau's theorem, the series $\sum_{n=1}^{\infty} |a_n|^q$ has bounded partial sums. To explore this possibility, we need some background information from constructive functional analysis.

A linear functional ϕ on a normed space X is said to be **normed** (or **normable**) if its norm

$$\|\phi\| = \sup\{\|\phi(x) : x \in X,\ \ \|x\| \leqslant 1\|\}$$

exists. Every linear functional on a finite-dimensional Banach space is normed; but if the same holds for an infinite-dimensional Hilbert space, then we can prove **LPO**. The following is the constructive version of the representation theorem for l_p spaces ([3], Chapter 7, Theorem (3.25)).

Theorem 1. *If p, q are conjugate exponents, then a bounded linear functional ϕ on l_p is normed if and only if there exists a (perforce unique) vector $\mathbf{a} \in l_q$ such that $\phi(\mathbf{x}) = \sum_{n=1}^{\infty} a_n x_n$ for each $\mathbf{x} \in l_p$, in which case $\|\phi\| = \|\mathbf{a}\|_q$.*

We shall also need the constructive **uniform boundedness theorem:**

Theorem 2. *Let $(T_n)_{n\geqslant 1}$ be a sequence of bounded linear mappings from a Banach space X into a normed space Y, such that $\|T_n\| \to \infty$ as $n \to \infty$. Then there exists $x \in X$ such that the sequence $(\|T_n x\|)_{n\geqslant 1}$ is unbounded.*

Proof. See [6] (Corollary 6.2.12) or [14].

The next result follows from Theorem 7 of [8]. We include the proof here to clarify the role played by the uniform boundedness theorem in our work, is a general one with a corollary classically equivalent to Landau's summability theorem.

Theorem 3. *Let $(T_n)_{n\geqslant 1}$ be a sequence of bounded linear mappings of a separable Banach space X into a normed space Y, converging pointwise to a linear mapping $T : X \to Y$. Then T is sequentially continuous.*

Proof. Let $(x_n)_{n\geqslant 1}$ be a sequence converging to 0 in X, and let $\varepsilon > 0$. By Ishihara's tricks [8] (Lemma 2), either $\|Tx_n\| < \varepsilon$ for all sufficiently large n or else $\|Tx_n\| > \varepsilon/2$ for infinitely many n. It suffices to rule out the latter alternative. To that end, we may suppose that $\|Tx_n\| > \varepsilon/2$ and $\|x_n\| < 1/n$ for each n. Then $y_n = \|x_n\|^{-1} x_n$ is a unit vector such that $\|Ty_n\| > n\varepsilon/2$. Since $T_n x \to Tx$ for each $x \in X$, we can construct inductively a strictly increasing sequence $(n_k)_{k\geqslant 1}$ of positive integers such that $\|T_{n_k} y_k\| > k\varepsilon/2$ for each k. Applying the uniform boundedness theorem, we obtain a unit vector $y \in X$ such that the sequence $(\|T_{n_k} y\|)_{k\geqslant 1}$ is unbounded. This is absurd, since $T_{n_k} y \to Ty$ as $k \to \infty$.

Corollary 1. *Let p be a positive integer, and* **a** *a sequence of complex numbers such that*

$$f(\mathbf{x}) = \sum_{n=1}^{\infty} a_n x_n \tag{1}$$

converges for each $\mathbf{x} \in l_p$. Then f is a sequentially continuous linear functional on l_p.

Proof. Noting that

$$f_k(\mathbf{x}) = \sum_{n=1}^{k} a_n x_n$$

defines a normed, and hence sequentially continuous, linear functional on X with

$$\|f_k\| = \left(\sum_{n=1}^{k} |a_k|^q \right)^{1/q},$$

we apply Theorem 3 with $X = l_p$.

Observe that the linear functional f in this corollary is continuous/bounded if and only the partial sums of the series $\sum_{i=1}^{\infty} |a_i|^q$ are bounded. Indeed, if f has a bound $c > 0$ and k is any positive integer, then, taking

$$\mathbf{x} = \left(a_1^* |a_1|^{q-2}, \ldots, a_k^* |a_k|^{q-2}, 0, 0, \ldots \right),$$

we obtain

$$\begin{aligned}\sum_{n=1}^{k}|a_n|^q &= \sum_{n=1}^{k}a_nx_n = f(\mathbf{x})\\ &\leqslant c\,\|\mathbf{x}\|_p = c\left(\sum_{n=1}^{k}\left|a_n^*\,|a_n|^{q-2}\right|^p\right)^{1/p}\\ &= c\left(\sum_{n=1}^{k}|a_n|^{p(q-1)}\right)^{1/p} = c\left(\sum_{n=1}^{k}|a_n|^q\right)^{1/p}\end{aligned}$$

and therefore

$$\left(\sum_{n=1}^{k}|a_n|^q\right)^{1/q} = \left(\sum_{n=1}^{k}|a_n|^q\right)^{1-1/p} \leqslant c.$$

Conversely, if c is a positive number such that c^q is a bound for the partial sums of $\sum_{n=1}^{\infty}|a_n|^q$, then for each $\mathbf{x}\in l_2$ and each k we have

$$\begin{aligned}|f(x_1,x_2,\ldots,x_k,0,0,\ldots)| &= \left|\sum_{n=1}^{k}a_nx_n\right|\\ &\leqslant \left(\sum_{n=1}^{k}|a_n|^q\right)^{1/q}\left(\sum_{n=1}^{k}|x_n|^p\right)^{1/p} \leqslant c\,\|\mathbf{x}\|_p\,.\end{aligned}$$

Since (by Corollary 1) f is sequentially continuous and

$$\mathbf{x} = \lim_{k\to\infty}(x_1,x_2,\ldots,x_k,0,0,\ldots)$$

in l_p, it follows that $|f(\mathbf{x})|\leqslant c\,\|\mathbf{x}\|_p$. Thus our suggestion that, under the hypotheses of Landau's theorem, the series $\sum_{n=1}^{\infty}|a_n|^q$ has bounded partial sums is equivalent to the corresponding linear functional, defined at (1), being continuous. This equivalence, taken with work of Ishihara [7], suggests that we bring into play the following notions.

We say that a subset S of $\mathbf{N}$ is **pseudobounded** if $\lim_{n\to\infty}n^{-1}s_n = 0$ for each sequence $(s_n)_{n\geqslant 1}$ in S. Following Ishihara [7], we consider the principle

BD-$\mathbb{N}$ *Every inhabited, countable, pseudobounded subset of the set $\mathbf{N}^+$ of positive integers is bounded,*

which holds in the intuitionistic and recursive models of BISH, but, being independent of Heyting arithmetic [12], is not provable within BISH. In [7], Ishihara proved that the statement 'Every sequentially continuous linear mapping from a separable metric space into a metric space is pointwise continuous' is equivalent to **BD-N**.

Our next result (whose proof has, unsurprisingly, some similarities to that of Lemma 20 in [9]) belongs to constructive reverse mathematics, a relatively new

field in which theorems are classified according to their equivalence, over some formal or (in this case) informal system for constructive mathematics, to certain principles such as **BD-N**. For more on this topic, see [10].

Theorem 4. *The following statement is equivalent to* ***BD*-N**.

(*) If p, q are conjugate exponents, and $\mathbf{a}$ *is a sequence of complex numbers such that*

$$f(\mathbf{x}) = \sum_{n=1}^{\infty} a_n x_n$$

converges for each $\mathbf{x} \in l_p$, then $\sum_{n=1}^{\infty} |a_n|^q$ has bounded partial sums.

Proof. The implication from **BD-N** to (*) is a consequence of Corollary 1 and the result of Ishihara mentioned immediately before the statement of this proposition. For the reverse implication, assume (*) and let

$$S \equiv \{s_1, s_2, \ldots\}$$

be an inhabited, countable, pseudobounded subset of $\mathbf{N}$. Without loss of generality, we may assume that $s_1 \leqslant s_2 \leqslant \cdots$. Setting

$$b_1 \equiv \sqrt[q]{s_1},\ b_{n+1} \equiv \sqrt[q]{s_{n+1} - s_n},$$

we need only prove that $\sum_{n=1}^{\infty} b_n x_n$ converges for each $\mathbf{x} \in l_p$: for then the partial sums of the series $\sum_{n=1}^{\infty} |b_n|^q$ are bounded, which implies the boundedness of the set S. Accordingly, fix $\mathbf{x} \in l_p$; we may assume that $x_n \geqslant 0$ for each n. Let $(n_k)_{k \geqslant 1}$ be a strictly increasing sequence of positive integers such that

$$\sum_{n=n_k}^{\infty} |x_n|^p < \left(\frac{1}{2^{k+1}k}\right)^p \tag{2}$$

for each k. Define

$$I_k \equiv \{n_k, n_k + 1, \ldots, n_{k+1} - 1\}\ .$$

Since S is pseudobounded, there exists κ such that $s_{n_{k+1}} < k$ for all $k \geqslant \kappa$. For $k' > k \geqslant \kappa$ we have

$$\begin{aligned}\left|\sum_{n=n_k}^{n_{k'}} b_n x_n\right| &\leqslant \sum_{j=k}^{k'} \left(\sum_{i \in I_j} b_i x_i\right) \leqslant \sum_{j=k}^{k'} \left(\sqrt[q]{\sum_{i \in I_j} |b_i|^q}\ \sqrt[p]{\sum_{i \in I_j} |x_i|^p}\right) \\ &\leqslant \sum_{j=k}^{k'} \frac{s_{j+1}}{2^{j+1}j} \leqslant \sum_{j=k}^{k'} 2^{-j-1} < 2^{-k}.\end{aligned}$$

It readily follows that the partial sums of $\sum_{n=1}^{\infty} b_n x_n$ form a Cauchy sequence, and hence that the series converges in $\mathbf{C}$.

Perhaps the most significant aspect of Theorem 4 is this: in contrast to Ishihara's original result relating **BD-N** and the passage from sequential to pointwise continuity, a result proved using a relatively strange space as the domain of the sequentially continuous mapping, Theorem 4 uses one of the standard spaces in functional analysis.

Our next result confirms that the use of the classical uniform boundedness theorem in proving Landau's theorem is not just a matter of convenience.

Proposition 1. *Statement (*) of Theorem 4 is equivalent to the classical uniform boundedness theorem in the form*

***UBT**$_c$* *If $(T_n)_{n\geqslant 1}$ is a sequence of bounded linear mappings of a Banach space X into a Banach space Y such that*

$$\{T_n x : n \geqslant 1\}$$

is bounded for each $x \in X$, then $\{\|T_n\| : N \geqslant 1\}$ is bounded.

Proof. Ishihara [11] has shown that **UBT**$_c$ is equivalent to **BD-N**. The result now follows from Theorem 4.

The question now arises: what can we say about Landau's theorem without assuming **BD-N**? The next three lemmas take some distance in the direction of an answer.

Lemma 1. *Let p, q be conjugate exponents, let $\mathbf{a}$ be a sequence of complex numbers such that $\sum_{n=1}^{\infty} a_n x_n$ converges for each $\mathbf{x}$ in l_p, and let $\phi : \mathbf{N}^+ \to \mathbf{R}^+$ be a strictly increasing mapping such that $\phi(k) \to \infty$ as $k \to \infty$. Let $(\lambda_k)_{k\geqslant 1}$ be an increasing binary sequence such that if $\lambda_k = 1 - \lambda_{k-1}$, then there exists $\nu \geqslant k$ such that $\sum_{n=1}^{\nu} |a_n|^q > \phi(k)$. Then either $\lambda_k = 0$ for all k or else there exists K such that $\lambda_K = 1$.*

Proof. Let $\mathbf{u}$ be a unit vector in l_q, set $\lambda_0 = 0$, and define a sequence $(f_k)_{k\geqslant 1}$ of normed linear functionals on l_p as follows. For each positive integer k if $\lambda_k = \lambda_{k-1}$, define

$$f_k(\mathbf{x}) = k \sum_{n=1}^{\infty} u_n x_n \quad (\mathbf{x} \in l_p)$$

and note that $\|f_k\| = k$. If $\lambda_k = 1 - \lambda_{k-1}$, then, choosing $\nu \geqslant k$ such that $\sum_{n=1}^{\nu} |a_n|^q > \phi(k)$, define

$$f_k(\mathbf{x}) = \sum_{n=1}^{\nu} a_n x_n \quad (\mathbf{x} \in l_p)$$

and note that $\|f_k\| > (\phi(k))^{1/q}$. Clearly, $\|f_k\| \to \infty$ as $k \to \infty$; so, by Theorem 2, there exists a unit vector $\mathbf{x} \in l_p$ such that $|f_k(\mathbf{x})| \to \infty$ as $k \to \infty$. Since $\sum_{n=1}^{\infty} a_n x_n$ converges, there exists K such that

$$|f_k(\mathbf{x})| > 1 + \left| \sum_{n=1}^{k} a_n x_n \right| \qquad (k \geqslant K)\,. \tag{3}$$

Suppose that $\lambda_k = 1 - \lambda_{k-1}$ for some $k > K$. Then $f_k(\mathbf{x}) = \sum_{n=1}^{\nu} a_n x_n$ for some $\nu \geqslant k$, which is absurd in view of (3). Hence $\lambda_k = \lambda_K$ for all $k \geqslant K$, from which the desired conclusion follows.

Lemma 2. *Let p, q be conjugate exponents, let $\mathbf{a}$ be a sequence of complex numbers such that $\sum_{n=1}^{\infty} a_n x_n$ converges for each $\mathbf{x}$ in l_p, and let $\phi : \mathbf{N}^+ \to \mathbf{R}^+$ be a strictly increasing mapping such that $\phi(n) \to \infty$ as $n \to \infty$. Let $(\lambda_k)_{k \geqslant 1}$ be an increasing binary sequence, and $(n_k)_{k \geqslant 1}$ an increasing sequence of positive integers, such that if $\lambda_k = 0$, then $\sum_{n=1}^{n_k} |a_n|^q > \phi(k) - 1$. Then there exists K such that $\lambda_K = 1$.*

Proof. Again let $\mathbf{u}$ be a unit vector in l_p and set $\lambda_0 = 0$. This time, for each $\mathbf{x}$ in l_p we define $f_k(\mathbf{x}) = \sum_{n=1}^{n_k} a_n x_n$ if $\lambda_k = 0$, and $f_k(\mathbf{x}) = k \sum_{n=1}^{\infty} u_n x_n$ if $\lambda_k = 1$. This produces a sequence $(f_k)_{k \geqslant 1}$ of normed linear functionals on l_p such that $\|f_k\| \to \infty$ as $k \to \infty$. Using Theorem 2, we produce a unit vector $\mathbf{x}$ in l_p such that $|f_k(\mathbf{x})| \to \infty$ as $k \to \infty$. Since $\sum_{n=1}^{\infty} a_n x_n$ converges, there exists K such that (3) holds. If $\lambda_K = 0$, then $f_K(\mathbf{x}) = \sum_{n=1}^{n_K} a_n x_n$, which is absurd in view of our choice of K. Hence $\lambda_K = 1$.

Lemma 3. *Let p, q be conjugate exponents, let $\mathbf{a}$ be a sequence of complex numbers such that $\sum_{n=1}^{\infty} a_n x_n$ converges for each $\mathbf{x}$ in l_p, and let $\phi : \mathbf{N}^+ \to \mathbf{R}^+$ be a strictly increasing mapping such that $\phi(k) \to \infty$ as $k \to \infty$. Then either $\sum_{n=1}^{k} |a_n|^q < \phi(k)$ for all k or else there exists k such that $\sum_{n=1}^{k} |a_n|^q > \phi(k) - 1$.*

Proof. Construct an increasing binary sequence $(\lambda_k)_{k \geqslant 1}$ such that

$$\lambda_k = 0 \Rightarrow \forall_{j \leqslant k} \left(\sum_{n=1}^{j} |a_n|^q < \phi(j) \right),$$

$$\lambda_k = 1 - \lambda_{k-1} \Rightarrow \sum_{n=1}^{k} |a_n|^q > \phi(k) - 1.$$

Now apply Lemma 1.

Proposition 2. *Let p, q be conjugate exponents, let $\mathbf{a}$ be a sequence of complex numbers such that $\sum_{n=1}^{\infty} a_n x_n$ converges for each $\mathbf{x}$ in l_p, and let $\phi : \mathbf{N}^+ \to \mathbf{R}^+$ be a strictly increasing mapping such that $\phi(k) \to \infty$ as $k \to \infty$. Then there exists $m > K$ such that $\sum_{n=K+1}^{m} |a_n|^q < \phi(m)$ for all $m \geqslant 1$.*

Proof. In view of the previous lemma, we may suppose that there exists n_1 such that $\sum_{n=1}^{n_1} |a_n|^q > \phi(n_1) - 1$. Setting $\lambda_1 = 0$ and applying Lemma 3 to the sequence $(0, 0, \ldots, 0, a_{n_1+1}, a_{n_1+2}, \ldots)$, we see that either $\sum_{n=n_1+1}^{m} |a_n|^q < \phi(m)$ for all $m > n_1$ or else there exists $n_2 > n_1$ such that $\sum_{n=n_1+1}^{n_2} |a_n|^q > \phi(n_2) - 1$. In the first case we set $\lambda_k = 1$ and $n_k = n_1$ for all $k \geqslant 2$; in the second we set $\lambda_2 = 0$. Carrying on in this way, we construct an increasing binary sequence $(\lambda_k)_{k \geqslant 1}$ and an increasing sequence $(n_k)_{k \geqslant 1}$ of positive integers such that

- if $\lambda_{k+1} = 0$, then $n_{k+1} > n_k$ and $\sum_{n=n_k+1}^{n_{k+1}} |a_n|^q > \phi(n_{k+1}) - 1$;

– if $\lambda_{k+1} = 1 - \lambda_k$, then $\sum_{n=n_k+1}^{m} |a_n|^q < \phi(m)$ for all $m > n_k$, and $n_j = n_k$ for all $j \geqslant k$.

Applying Lemma 2, we obtain the desired conclusion.

It follows for example, that, under the hypotheses of Landau's theorem, for each positive integer m there exists N such that

$$\sum_{i=N}^{n} |a_i|^q < \underbrace{\log(\log(\cdots(\log n)\cdots))}_{m \text{ instances of ``log''}}$$

for all $n > N$. This is a long way from showing that the partial sums of $\sum_{i=1}^{\infty} |a_i|^q$ are bounded, but it is progress of a kind.

We now have a constructive substitute for the convergence of a_n to 0 in Landau's theorem.

Proposition 3. *Let p, q be conjugate exponents, and let $\mathbf{a}$ be a sequence of complex numbers such that the series $\sum_{n=1}^{\infty} a_n x_n$ converges for each $\mathbf{x}$ in l_p. Then for each $\varepsilon > 0$ and each positive integer ν there exists k such that $\sum_{n=(k-1)\nu}^{k\nu} |a_n|^q < \varepsilon$.*

Proof. Fix a unit vector $\mathbf{u}$ in l_q. For each positive integer k, construct an increasing binary sequence $(\lambda_k)_{k\geqslant 1}$ such that

$$\lambda_k = 0 \Rightarrow \forall_{j\leqslant k} \left(\sum_{n=(j-1)\nu}^{j\nu} |a_n|^q > \frac{\varepsilon}{2} \right),$$

$$\lambda_k = 1 - \lambda_{k-1} \Rightarrow \sum_{n=(j-1)\nu}^{j\nu} |a_n|^q < \varepsilon.$$

Applying Lemma 2 with $\phi(k) = 1 + \frac{k\varepsilon}{2}$, we see that there exists N such that $\lambda_N = 1$; whence $\sum_{n=(k-1)\nu}^{k\nu} |a_n|^q < \varepsilon$ for some $k \leqslant N$.

Corollary 2. *Let p, q be conjugate exponents, and let $\mathbf{a}$ be a sequence of complex numbers such that the series $\sum_{n=1}^{\infty} a_n x_n$ converges for each $\mathbf{x}$ in l_p. Then there exists a sequence $(n_k)_{k\geqslant 1}$ of positive integers such that for each k, $n_k + k < n_{k+1}$ and*

$$\sum_{n=n_k+1}^{n_k+k} |a_n|^q < 2^{-k}.$$

Proof. By Proposition 3, there exists n_1 such that $|a_{n_1}|^q < 2^{-1}$. Having computed n_k with the desired properties, apply Proposition 3 to the sequence $(a_n)_{n>n_k+k}$, to produce $n_{k+1} > n_k + k$ such that $\sum_{n=n_{k+1}+1}^{n_{k+1}+k+1} |a_n|^q < 2^{-k-1}$. This completes the inductive construction of the sequence $(n_k)_{k\geqslant 1}$.

The conclusion of Corollary 2 holds for any binary sequence with at most one term equal to 1, and so is not enough to yield constructively the result that, under the hypotheses of that corollary and with $p = q = 2$, $a_n \to 0$ as $n \to \infty$.

We conclude the paper by proving a constructive version of Landau's summability theorem that is classically equivalent to the classical version but has stronger hypotheses and conclusion than Corollary 1. For this we recall the constructive **least-upper-bound principle**:

> *In order that an inhabited set S of real numbers that is bounded above have a supremum, it is necessary and sufficient that S be* ***order located****, in the sense that for all positive α, β with $\alpha < \beta$, either β is an upper bound for S or else there exists $x \in S$ such that $x > \alpha$* ([3], page 37, Proposition (4.3)).

Theorem 5. *Let p, q be conjugate exponents, and let* $\mathbf{a}$ *be a sequence of complex numbers such that $\sum_{n=1}^{\infty} a_n x_n$ converges for each* $\mathbf{x}$ *in l_p. Then the following are equivalent.*

(i) *The series $\sum_{n=1}^{\infty} |a_n|^q$ is convergent.*
(ii) *For all α, β with $0 < \alpha < \beta$, either $\sum_{n=1}^{k} |a_n|^q < \beta$ for all k or else there exists k such that $\sum_{n=1}^{k} |a_n|^q > \alpha$.*

Proof. It is clear that if $\sum_{n=1}^{\infty} |a_n|^q$ converges, then (ii) holds. Conversely, assuming (ii), construct an increasing binary sequence $(\lambda_k)_{k \geqslant 1}$ and an increasing sequence $(n_k)_{k \geqslant 0}$ of positive integers with $n_0 = 0$, such that

- ▷ if $\lambda_k = 0$, then $n_k > n_{k-1}$ and $\sum_{i=1}^{n_k} |a_i|^q > k$, and
- ▷ if $\lambda_k = 1$, then $n_k = n_{k-1}$ and $\sum_{i=1}^{j} |a_i|^q < k + 1$ for all j.

To do so, first observe that either $\sum_{i=1}^{j} |a_i|^q < 2$ for all j or else there exists $n_1 \geqslant 1$ such that $\sum_{i=1}^{n_1} |a_i|^q > 1$. In the first case set $\lambda_1 = n_1 = 1$; in the second, set $\lambda_1 = 0$. Now suppose we have found λ_{k-1} and n_{k-1} with the applicable properties. If $\lambda_{k-1} = 1$, set $\lambda_k = 1$ and $n_k = n_{k-1}$. If $\lambda_{k-1} = 0$, then by (ii), either $\sum_{i=1}^{j} |a_i|^q < k + 1$ for all j, in which case we set $\lambda_k = 1$ and $n_k = n_{k-1}$; or else there exists n_k such that $\sum_{i=1}^{n_k} |a_i|^q > k$. In the latter case, replacing n_k by a sufficiently large positive integer, we may assume that $n_k > n_{k-1}$; we then set $\lambda_k = 0$ to complete the inductive construction. Taking $\phi(k) = k + 1$ in Lemma 2, we obtain K such that $\lambda_K = 1$. The partial sums of $\sum_{i=1}^{\infty} |a_i|^q$ are therefore bounded above by $K + 1$. It follows from (ii) and the constructive least-upper-bound principle that $\sum_{i=1}^{\infty} |a_i|^q$ converges in $\mathbf{R}$.

In view of the constructive least-upper-bound principle, it is curious that condition (ii) is used to prove that the partial sums of $\sum_{n=1}^{\infty} |a_n|^2$ are bounded before it is again invoked to prove that their supremum exists.

For related work within the framework of Weihrauch's theory of Type Two Effectivity [15], see [4]. For connections between that theory and Bishop-style constructive mathematics, see [2].

Acknowledgements: The authors are grateful to (i) the Royal Society of New Zealand, for supporting Bridges, through Marsden Award UOC0502, on visits to Munich during which this work was carried out, and (ii) the Alexander von Humboldt Stiftung for supporting Berger by a Feodor Lynen Return Fellowship.

References

1. P. Aczel and M. Rathjen, *Notes on Constructive Set Theory*, Report No. 40, Institut Mittag-Leffler, Royal Swedish Academy of Sciences, 2001.
2. A. Bauer, 'Realizability as the connection between computable and constructive mathematics', Proceedings of CCA 2005, Kyoto, Japan, 25–29 August 2005; to appear.
3. E.A. Bishop and D.S. Bridges, *Constructive Analysis*, Grundlehren der Math. Wiss. **279**, Springer-Verlag, Heidelberg, 1985.
4. V. Brattka, 'Computability on non-separable Banach spaces and Landau's theorem', in: *From Sets and Types to Topology and Analysis* (L. Crosilla and P.M. Schuster, eds), 316–333, Oxford Logic Guides **48**, Clarendon Press, Oxford, 2005.
5. D.S. Bridges and F. Richman, *Varieties of Constructive Mathematics*, London Math. Soc. Lecture Notes **97**, Cambridge Univ. Press, Cambridge, 1985.
6. D.S. Bridges and L.S. Vîţă, *Techniques of Constructive Analysis*, Universitext, Springer-New-York, 2006.
7. H. Ishihara, 'Continuity properties in constructive mathematics', J. Symbolic Logic **57**, 557–565, 1992.
8. H. Ishihara, 'Sequential continuity of linear mappings in constructive analysis', J. Univ. Comp. Sci. **3**(11), 1250–1254, 1997.
9. H. Ishihara, 'Sequential continuity in constructive mathematics', in: *Combinatorics, Computability and Logic* (C.S. Calude, M.J. Dinneen and S. Sburlan, eds), 5–12, Springer-Verlag, London, 2001.
10. H. Ishihara, 'Reverse mathematics in Bishop's constructive mathematics', Phil. Scientiae (Cahier Spécial), **6**, 43–59, 2006.
11. H. Ishihara, 'The uniform boundedness theorem and a boundedness principle', Japan Adv. Inst. Science & Technology, Ishikawa, Japan, 2007.
12. P. Lietz, *From Constructive Mathematics to Computable Analysis via the Realizability Interpretation,* Dr. rer. nat. dissertation, Technische Universität Darmstadt, Germany, 2004.
13. J. Myhill, 'Constructive set theory', J. Symbolic Logic **40**(3), 1975, 347–382.
14. H. L. Royden, 'Aspects of constructive analysis', in: *Errett Bishop: Reflections on Him and His Research* (M. Rosenblatt, ed.), Contemporary Math. **39**, American Math. Soc., Providence RI, 1985.
15. K. Weihrauch, *Computable Analysis*, Springer-Verlag, Heidelberg, 2000.

Separations of Non-monotonic Randomness Notions

Laurent Bienvenu, Rupert Hölzl, Thorsten Kräling, and Wolfgang Merkle

Institut für Informatik, Ruprecht-Karls-Universität,
Heidelberg, Germany

Abstract. In the theory of algorithmic randomness, several notions of random sequence are defined via a game-theoretic approach, and the notions that received most attention are perhaps Martin-Löf randomness and computable randomness. The latter notion was introduced by Schnorr and is rather natural: an infinite binary sequence is computably random if no total computable strategy succeeds on it by betting on bits in order. However, computably random sequences can have properties that one may consider to be incompatible with being random, in particular, there are computably random sequences that are highly compressible. The concept of Martin-Löf randomness is much better behaved in this and other respects, on the other hand its definition in terms of martingales is considerably less natural.
Muchnik, elaborating on ideas of Kolmogorov and Loveland, refined Schnorr's model by also allowing non-monotonic strategies, i.e. strategies that do not bet on bits in order. The subsequent "non-monotonic" notion of randomness, now called Kolmogorov-Loveland-randomness, has been shown to be quite close to Martin-Löf randomness, but whether these two classes coincide remains a fundamental open question.
In order to get a better understanding of non-monotonic randomness notions, Miller and Nies introduced some interesting intermediate concepts, where one only allows non-adaptive strategies, i.e., strategies that can still bet non-monotonically, but such that the sequence of betting positions is known in advance (and computable). Recently, these notions were shown by Kastermans and Lempp to differ from Martin-Löf randomness. We continue the study of the non-monotonic randomness notions introduced by Miller and Nies and obtain results about the Kolmogorov complexities of initial segments that may and may not occur for such sequences, where these results then imply a complete classification of these randomness notions by order of strength.

1 Introduction

Random sequences are the central object of study in algorithmic randomness and have been investigated intensively over the last decade, which led to a wealth of interesting results clarifying the relations between the various notions of randomness and revealing interesting interactions with notions such as computational power [2, 5, 11].

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 71-82
http://drops.dagstuhl.de/opus/volltexte/2009/2260

Intuitively speaking, a binary sequence is random if the bits of the sequence do not have effectively detectable regularities. This idea can be formalized in terms of betting strategies, that is, a sequence will be called random in case the capital gained by successive bets on the bits of the sequence according to a fixed betting strategy must remain bounded, with fair payoff and a fixed set of admissible betting strategies understood.

The notions of random sequences that have received most attention are Martin-Löf randomness and computable randomness. Here a sequence is called computably random if no total computable betting strategy can achieve unbounded capital by betting on the bits of the sequence in the natural order, a definition that indeed is natural and suggests itself. However, computably random sequences may lack certain properties associated with the intuitive understanding of randomness, for example there are such sequences that are highly compressible, i.e., show a large amount of redundancy, see Theorem 4 below. Martin-Löf randomness behaves much better in this and other respects. Indeed, the Martin-Löf random sequences can be characterized as the sequences that are incompressible in the sense that all their initial segments have essentially maximal Kolmogorov complexity, and in fact this holds for several versions of Kolmogorov complexity according to celebrated results by Schnorr, by Levin and, recently, by Miller and Yu [2]. On the other hand, it has been held against the concept of Martin-Löf randomness that its definition involves effective approximations, i.e., a very powerful, hence rather unnatural model of computation, and indeed the usual definition of Martin-Löf randomness in terms of left-computable martingales, that is, in terms of betting strategies where the gained capital can be effectively approximated from below, is not very intuitive.

It can be shown that Martin-Löf randomness strictly implies computable randomness. According to the preceding discussion the latter notion is too inclusive while the former may be considered unnatural. Ideally, we would therefore like to find a more natural characterization of ML-randomness; or, if that is impossible, we are alternatively interested in a notion that is close in strength to ML-randomness, but has a more natural definition. One promising way of achieving such a more natural characterization or definition could be to use computable betting strategies that are more powerful than those used to define computable randomness.

Muchnik [10] proposed to consider computable betting strategies that are non-monotonic in the sense that the bets on the bits need not be done in the natural order, but such that the bit to bet on next can be computed from the already scanned bits. The corresponding notion of randomness is called Kolmogorov-Loveland randomness because Kolmogorov and Loveland independently had proposed concepts of randomness defined via non-monotonic selecting of bits.

Kolmogorov-Loveland randomness is implied by and in fact is quite close to Martin-Löf randomness, see Theorem 14 below, but whether the two notions are distinct is one of the major open problems of algorithmic randomness. In order to get a better understanding of this open problem and of non-monotonic

randomness in general, Miller and Nies [9] introduced restricted variants of Kolmogorov-Loveland randomness, where the sequence of betting positions must be non-adaptive, i.e., can be computed in advance without knowing the sequence on which one bets.

The randomness notions mentioned so far are determined by two parameters that correspond to the columns and rows, respectively, of the table in Figure 1. First, the sequence of places that are scanned and on which bets may be placed, while always being given effectively, can just be monotonic, can be equal to $\pi(0), \pi(1), \ldots$ for a permutation or an injection π from $\mathbb{N}$ to $\mathbb{N}$, or can be adaptive, i.e., the next bit depends on the bits already scanned. Second, once the sequence of scanned bits is determined, betting on these bits can be according to a betting strategy where the corresponding martingale is total or partial computable, or is left-computable. The known inclusions between the corresponding classes of random sequences are shown in Figure 1, see Section 2 for technical details and for the definitions of the class acronyms that occur in the figure.

	monotonic		permutation		injection		adaptive
total	**TMR**	$=$	**TPR**	$\supseteq$	**TIR**	$\supseteq$	**KLR**
	$\cup\mid$		$\cup\mid$		$\cup\mid$		$\parallel$
partial	**PMR**	$\supseteq$	**PPR**	$\supseteq$	**PIR**	$\supseteq$	**KLR**
	$\cup\mid$		$\cup\mid$		$\cup\mid$		$\cup\mid$
left-computable	**MLR**	$=$	**MLR**	$=$	**MLR**	$=$	**MLR**

Fig. 1. Known class inclusions

The classes in the last row of the table in Figure 1 all coincide with the class of Martin-Löf random sequences by the folklore result that left-computable martingales always yield the concept of Martin-Löf randomness, no matter whether the sequence of bits to bet on is monotonic or is determined adaptively, because even in the latter, more powerful model one can uniformly in k enumerate an open cover of measure at most $1/k$ for all the sequences on which some universal martingale exceeds k. Furthermore, the classes in the first and second row of the last column coincide with the class of Kolmogorov-Loveland random sequences, because it can be shown that total and partial adaptive betting strategies yield the same concept of random sequence [6]. Finally, it follows easily from results of Buhrman et al. [1] that the class **TMR** of computably random sequences coincides with the class **TPR** of sequences that are random with respect to total permutation martingales, i.e., the ability to scan the bits of a sequence according to a computable permutation does not increase the power of total martingales.

Concerning non-inclusions, it is well-known that it holds that

$$\mathbf{KLR} \subsetneq \mathbf{PMR} \subsetneq \mathbf{TMR}.$$

Furthermore, Kastermans and Lempp [3] have recently shown that the Martin-Löf random sequences form a proper subclass of the class **PIR** of partial injective random sequences, i.e., $\mathbf{MLR} \subsetneq \mathbf{PIR}$.

Apart from trivial consequences of the definitions and the results just mentioned, nothing has been known about the relations of the randomness notions between computable randomness and Martin-Löf randomness in Figure 1. In what follows, we investigate the six randomness notions that are shown in Figure 1 in the range between **PIR** and **TMR**, i.e., between partial injective randomness as introduced below and computable randomness. We obtain a complete picture of the inclusion structure of these notions, more precisely we show that the notions are mutually distinct and indeed are mutually incomparable with respect to set theoretical inclusion, except for the inclusion relations that follow trivially by definition and by the known relation $\mathbf{TMR} \subseteq \mathbf{TPR}$, see Figure 2 at the end of this paper. Interestingly these separation results are obtained by investigating the possible values of the Kolomogorov complexity of initial segments of random sequences for the different strategy types, and for some randomness notions we obtain essentially sharp bounds on how low these complexities can be.

Notation. We conclude the introduction by fixing some notation. The set of finite strings (or finite binary sequences, or words) is denoted by $2^{<\omega}$, ϵ being the empty word. We denote the set of infinite binary sequences by 2^ω. Given two finite strings w, w', we write $w \sqsubseteq w'$ if w is a prefix of w'. Given an element x of 2^ω or $2^{<\omega}$, $x(i)$ denotes the i-th bit of x (where by convention there is a 0-th bit and $x(i)$ is undefined if x is a word of length less than $i+1$). If $A \in 2^\omega$ and $X = \{x_0 < x_1 < x_2 < \ldots\}$ is a subset of $\mathbb{N}$ then $A \upharpoonright X$ is the finite or infinite binary sequence $A(x_0)A(x_1)\ldots$. We abbreviate $A \upharpoonright \{0, \ldots, n-1\}$ by $A \upharpoonright n$ (i.e., the prefix of A of length n).

C and K denote plain and prefix-free Kolmogorov complexity, respectively [2, 5]. The function log designates the logarithm of base 2. An **order** is a function $h : \mathbb{N} \to \mathbb{N}$ that is non-decreasing and tends to infinity.

2 Permutation and injection randomness

We now review the concept of martingale and betting strategy that are central for the unpredictability approach to define notions of an infinite random sequence.

Definition 1. *A **martingale** is a nonnegative, possibly partial, function $d : 2^{<\omega} \to \mathbb{Q}$ such that for all $w \in 2^{<\omega}$, $d(w0)$ is defined if and only if $d(w1)$ is, and if these are defined, then so is $d(w)$, and the relation $2d(w) = d(w0) + d(w1)$ holds. A martingale **succeeds** on a sequence $A \in 2^\omega$ if $d(A \upharpoonright n)$ is defined for all n, and $\limsup d(A \upharpoonright n) = +\infty$. We denote by $\mathrm{Succ}(d)$ the **success set** of d, i.e., the set of sequences on which d succeeds.*

Intuitively, a martingale represents the capital of a player who bets on the bits of a sequence $A \in 2^\omega$ in order, where at every round she bets some amount of money

on the value of the next bit of A. If her guess is correct, she doubles her stake. If not, she loses her stake. The quantity $d(w)$, with w a string of length n, represents the capital of the player before the n-th round of the game (by convention there is a 0-th round) when the first n bits revealed so far are those of w.

We say that a sequence A is **computably random** if no total computable martingale succeeds on it. One can extend this in a natural way to partial computable martingales: a sequence A is **partial computably random** if no partial computable martingale succeeds on it. No matter whether we consider partial or total computable martingales, this game model can be seen as too restrictive by the discussion in the introduction. Indeed, one could allow the player to bet on bits in any order she likes (as long as she can visit each bit at most once). This leads us to extend the notion of martingale to the notion of strategy.

Definition 2. *A **betting strategy** is a pair $b = (d, \sigma)$ where d is a martingale and $\sigma : 2^{<\omega} \to \mathbb{N}$ is a function.*

For a strategy $b = (d, \sigma)$, the term σ is called the *scan rule*. For a string w, $\sigma(w)$ represents the position of the next bit to be visited if the player has read the sequence of bits w during the previous moves. And as before, d specifies how much money is bet at each move. Formally, given an $A \in 2^\omega$, we define by induction a sequence of positions $n_0, n_1, \ldots$ by

$$\begin{cases} n_0 = \sigma(\epsilon), \\ n_{k+1} = \sigma\left(A(n_0)A(n_1)\ldots A(n_k)\right) \text{ for all } k \geq 0 \end{cases}$$

and we say that $b = (d, \sigma)$ **succeeds** on A if the n_i are all defined and pairwise distinct (i.e., no bit is visited twice) and

$$\limsup_{k \to +\infty} d\left(A(n_0)\ldots A(n_k)\right) = +\infty$$

Here again, a betting strategy $b = (d, \sigma)$ can be total or partial. In fact, its partiality can be due either to the partiality of d or to the partiality of σ. We say that a sequence is **Kolmogorov-Loveland random** if no total computable betting strategy succeeds on it. As noted in [8], the concept of Kolmogorov-Loveland randomness remains the same if one replaces "total computable" by "partial computable" in the definition.

Kolmogorov-Loveland randomness is implied by Martin-Löf randomness and whether the two notions can be separated is one of the most important open problems on algorithmic randomness. As we discussed above, Miller and Nies [9] proposed to look at intermediate notions of randomness, where the power of non-monotonic betting strategies is limited. In the definition of a betting strategy, the scan rule is adaptive, i.e., the position of the next visited bit depends on the bits previously seen. It is interesting to look at non-adaptive games.

Definition 3. *In the above definition of a strategy, when $\sigma(w)$ only depends on the length of w for all w (i.e., the decision of which bit should be chosen at each move is independent of the values of the bits seen in previous moves), we identify σ with the (injective) function $\pi : \mathbb{N} \to \mathbb{N}$, where for all n $\pi(n)$ is the value of σ on words of length n ($\pi(n)$ indicates the position of the bit visited during the n-th move), and we say that $b = (d, \pi)$ is an **injection strategy**. If moreover π is bijective, we say that b is a **permutation strategy**. If π is the identity, the strategy $b = (d, \pi)$ is said to be **monotonic**, and can clearly be identified with the martingale d.*

All this gives a number of possible non-adaptive, non-monotonic, randomness notions: one can consider either monotonic, permutation, or injection strategies, and either total computable or partial computable ones. This gives a total of six randomness classes, which we denote by

$$\mathbf{TMR},\ \mathbf{TPR},\ \mathbf{TIR},\ \mathbf{PMR},\ \mathbf{PPR},\ \text{and}\ \mathbf{PIR}, \tag{1}$$

where the first letter indicates whether we consider total (T) or partial (P) strategies, and the second indicates whether we look at monotonic (M), permutation (P) or injection (I) strategies. For example, the class **TMR** is the class of computably random sequences, while the class **PIR** is the class of sequences A such that no partial injection strategy succeeds on A. Recall in this connection that the known inclusions between the six classes in (1) and the classes **KLR** and **MLR** of Kolmogorov-Loveland random and Martin-Löf random sequences have been shown in Figure 1 above.

3 Randomness notions based on total computable strategies

We begin our study by the randomness notions arising from the game model where strategies are total computable. As we will see, in this model, it is possible to construct sequences that are random and yet have very low Kolmogorov complexity (i.e. all their initial segments are of low Kolmogorov complexity). We will see in the next section that this is no longer the case when we allow partial computable strategies in the model.

3.1 Sequences in TMR and TPR may have low complexity

The following theorem is a first illustration of the phenomenon we just described.

Theorem 4 (Lathrop and Lutz [4], Muchnik [10]). *For every computable order h, there is a sequence $A \in \mathbf{TMR}$ such that, for all $n \in \mathbb{N}$,*

$$\mathrm{C}\left(A \upharpoonright n \mid n\right) \leq h(n) + \mathrm{O}(1).$$

Proof (Idea). Defeating one total computable martingale is easy and can be done computably, i.e., for every total computable martingale d there exists a sequence A, uniformly computable in d, such that $A \notin \mathrm{Succ}(d)$. Indeed, given a martingale d. For any given w, one has either $d(w0) \leq d(w)$ or $d(w1) \leq d(w)$. Thus, one can easily construct a computable sequence A by setting $A \upharpoonright 0 = \epsilon$ and by induction, having defined $A \upharpoonright n$, we choose $A \upharpoonright n+1 = (A \upharpoonright n)i$ where $i \in \{0,1\}$ is such that $d((A \upharpoonright n)i) \leq d(A \upharpoonright n)$. This can of course be done computably since d is total computable, and by construction of A, $d(A \upharpoonright n)$ is non-increasing, meaning in particular that d does not succeed against A.

Defeating a finite number of total computable martingales is equally easy. Indeed, given a finite number $d_1, \ldots, d_k$ of such martingales, their sum $D = d_1 + \ldots + d_k$ is itself a total computable martingale (this follows directly from the definition). Thus, we can construct as above a computable sequence A that defeats D. And since $D \geq d_i$ for all $1 \leq i \leq k$, this implies that A defeats all the d_i. Note that this argument would work just as well if we had taken D to be any weighted sum $\alpha_1 d_1 + \ldots + \alpha_k d_k$, with positive rational constants α_i.

We now need to deal with the general case where we have to defeat *all* total computable martingales simultaneously. What we do is simply add total martingales one by one: we start by diagonalizing against the first total martingale d_1, then (maybe after a long time) we may introduce the second martingale d_2, with a small coefficient α_2 (to ensure that introducing d_2 does not cost us too much) and then consider the martingale $d_1 + \alpha_2 d_2$. Much later we can introduce the third martingale d_3 with an even smaller coefficient α_3, and diagonalize against $d_1 + \alpha_2 d_2 + \alpha_3 d_3$, and so on. So in each step of the construction we have to consider just a finite number of martingales, and if we add the martingales sufficiently slowly, it is easy to see than we can keep the complexity low (indeed, as long as we know what martingales we are diagonalizing against, the construction is computable; note however that the sequence as a whole will not be computable since whenever we add a martingale we need to store some information, consisting of its code, when it was added and with which coefficient). □

It turns out that, perhaps surprisingly, the classes **TMR** and **TPR** coincide. This fact was stated explicitely in Merkle et al [8], but is easily derived from the ideas introduced in Buhrman et al [1]. We present the main ideas of their proof as we will later need them. We shall prove:

Theorem 5. *Let $b = (d, \pi)$ be a total computable permutation strategy. There exists a total computable martingale d such that* $\mathrm{Succ}(b) \subseteq \mathrm{Succ}(d)$.

This theorem states that total permutation strategies are no more powerful than total monotonic strategies, which obviously entails **TMR** = **TPR**. Before we can prove it, we first need a definition.

Definition 6. *Let $b = (d, \pi)$ be a total injective strategy. Let $w \in 2^{<\omega}$. We can run the strategy b on w as if it were an element of 2^ω, stopping the game when b*

asks to bet on a bit of position outside w. This game is of course finite (for a given w) since at most $|w|$ bets can be made. We define $\hat{b}(w)$ to be the capital of b at the end of this game. Formally: $\hat{b}(w) = d\left(w_{\pi(0)} \dots w_{\pi(N-1)}\right)$ where N is the smallest integer such that $\pi(N) \geq |w|$.

Note that if $b = (d, \pi)$ is a total computable injection martingale, $\hat{b}$ is total computable. If $\hat{b}$ was itself a monotonic martingale, Theorem 5 would be proven. This is however not the case in general. The trick is, given a betting strategy b and a word w, to look at the *expected value* of b on w, i.e., look at the mathematical expectation of $b(w')$ for large enough extensions w' of w. Specifically, given a total betting strategy $b = (d, \pi)$ and a word w of length n, we take an integer M large enough to have

$$\pi([0, \dots, M-1]) \cap [0, \dots, n-1] = \pi(\mathbb{N}) \cap [0, \dots, n-1]$$

(i.e. the strategy b will never bet on a bit of position less than n after the M-th move), and define:

$$\mathrm{Av}_b(w) = \frac{1}{2^M} \sum_{\substack{w \sqsubseteq w' \\ |w'|=M}} \hat{b}(w')$$

Proposition 7 (Buhrman et al [1], Kastermans-Lempp [3]).

(i) The quantity $\mathrm{Av}_b(w)$ (defined above) is well-defined i.e. does not depend on M as long as it satisfies the required condition.
(ii) For a total injective strategy b, Av_b is a martingale.
(iii) For a given injective strategy b and a given word w of length n, $\mathrm{Av}_b(w)$ can be computed if we know the set $\pi(\mathbb{N}) \cap [0, \dots, n-1]$. In particular, if b is a total computable permutation strategy, then Av_b is total computable.

As Buhrman et al. [1] explained, it is not true in general that if a total computable injective strategy b succeeds on a sequence A, then Av_b also succeeds on A. However, this can be dealt with using the well-known "saving trick". Suppose we are given a martingale d with initial capital, say, 1. Consider the variant d' of d that does the following: when run on a given sequence A, d' initially plays exactly as d. If at some stage of the game d' reaches a capital of 2 or more, it then puts half of its capital on a "bank account", which will never be used again. From that point on, d' bets half of what d does, i.e. starts behaving like $d/2$ (plus the saved capital). If later in the game the "non-saved" part of its capital reaches 2 or more, then half of it is placed on the bank account and then d' starts behaving like $d/4$, and so on. For every martingale d' that behaves as above (i.e. saves half of its capital as soon as it exceeds twice its starting capital), we say that d' has the "saving property".

Lemma 8. *Let $b = (d, \pi)$ be a total injective strategy such that d has the saving property. Let $d' = \mathrm{Av}_b$. Then $\mathrm{Succ}(b) \subseteq \mathrm{Succ}(d')$.*

Now the proof of Theorem 5 is as follows. Let $b = (d, \pi)$ be a total computable permutation strategy. By the above discussion, let d' be the saving version of d, so that $\mathrm{Succ}(d) \subseteq \mathrm{Succ}(d')$. Setting $b' = (d', \pi)$, we have $\mathrm{Succ}(b) \subseteq \mathrm{Succ}(b')$. By Proposition 7 and Lemma 8, $d'' = \mathrm{Av}_{b'}$ is a total computable martingale, and

$$\mathrm{Succ}(b) \subseteq \mathrm{Succ}(b') \subseteq \mathrm{Succ}(d'')$$

as wanted. □

3.2 Understanding the strength of injective strategies: the class TIR

While the class of computably random sequences (i.e. the class **TMR**) is closed under computable permutations of the bits, we now see that this result does not extend to computable injections. To wit, the following theorem is true.

Theorem 9. *Let $A \in 2^\omega$. Let $\{n_k\}_{k\in\mathbb{N}}$ be a computable sequence of integers such that $n_{k+1} \geq 2n_k$ for all k. Suppose that A is such that:*

$$\mathrm{C}\left(A \upharpoonright n_k \mid k\right) \leq \log(n_k) - 3\log(\log(n_k))$$

*for infinitely many k. Then $A \notin$ **TIR***

As an immediate corollary, we get the following.

Corollary 10. *If for a sequence A we have for all n that $\mathrm{C}\left(A \upharpoonright n \mid n\right) < \log n - 4\log\log n + \mathrm{O}(1)$, then $A \notin$ **TIR**.*

The lower bound on Kolmogorov complexity given in Theorem 9 is quite tight, as witnessed by the following theorem.

Theorem 11. *For every computable order h there is a sequence $A \in$ **TIR** such that $\mathrm{C}(A \upharpoonright n \mid n) \leq \log(n) + h(n) + \mathrm{O}(1)$ (in particular, $\mathrm{C}(A \upharpoonright n) \leq 2\log(n) + h(n) + \mathrm{O}(1)$).*

4 Randomness notions based on partial computable strategies

We now turn our attention to the second line of Figure 1, i.e., to those randomness notions that are based on partial computable betting strategies.

4.1 The class PMR: partial computable martingales are stronger than total ones

We have seen in the previous section that some sequences in **TIR** (and a fortiori **TPR** and **TMR**) may be of very low complexity, namely logarithmic. This is not the case anymore when one allows partial computable strategies, even monotonic ones.

Theorem 12 (Merkle [7]). *If* $\mathrm{C}(A \upharpoonright n) = \mathrm{O}(\log n)$ *then* $A \notin \mathbf{PMR}$.

By this theorem, together with Theorem 4, we immediately see that the class **PMR** is strictly contained in **TMR**. However, the next theorem, proven by An. A. Muchnik, shows that allowing slightly super-logarithmic growth of the Kolmogorov complexity is enough to construct a sequence in **PMR**.

Theorem 13 (Muchnik et al. [10]). *For every computable order* h *there is a sequence* $A \in \mathbf{PMR}$ *such that, for all* $n \in \mathbb{N}$,

$$\mathrm{C}\left(A \upharpoonright n \mid n\right) \leq h(n)\log(n) + \mathrm{O}(1).$$

4.2 The class PPR

In the case of total strategies, allowing permutation gives no real additional power, as $\mathbf{TMR} = \mathbf{TPR}$. Very surprisingly, Muchnik showed that in the case of partial computable strategies, permutation strategies are a real improvement over monotonic ones. To wit, the following theorem (quite a contrast to Theorem 13!) holds.

Theorem 14 (Muchnik [10]). *If there is a computable order* h *such that for all* n *we have* $\mathrm{K}(A \upharpoonright n) \leq n - h(n) - \mathrm{O}(1)$, *then* $A \notin \mathbf{PPR}$.

Note that the proof used by Muchnik in [10] works if we replace K by C in the above statement. So we now know that any sequence in **PPR** must have infinitely many initial segments of high Kolmogorov complexity. The following theorem shows that some sequences in **PPR** also have infinitely many initial segments of very low complexity.

Theorem 15. *For every computable order* h *there is a sequence* $A \in \mathbf{PPR}$, *such that there are infinitely many* n *where* $\mathrm{C}\left(A \upharpoonright n \mid n\right) < h(n)$.

Furthermore, if we have an infinite computable set $S \subseteq \mathbb{N}$, *we can choose the infinitely many lengths* n *such that they all are contained in* S.

The proof of this theorem requires the following "totalization" technique.

Proposition 16. *Let* $b = (d, \pi)$ *be a partial computable permutation strategy (resp. injective strategy). Let* $\mathcal{C}$ *be an effectively closed subset of* 2^ω. *Suppose that* b *is total on every element of* $\mathcal{C}$. *Then there exists a total computable permutation strategy (resp. injective strategy)* b' *such that* $\mathrm{Succ}(b) \cap \mathcal{C} = \mathrm{Succ}(b') \cap \mathcal{C}$.

Proof (of Theorem 15, sketch). Like for Theorem 4, we proceed by diagonalization against all partial permutation martingales, "transforming" them into total monotonic martingales in order to carry out the construction. Suppose that during the construction we have constructed an initial segment w of our infinite sequence by diagonalization against a weighted sum of total monotonic martingales $\alpha_1 d_1 + \ldots + \alpha_k d_k$ (say that the value of this sum at w is less than 2). Suppose that we now want to add a new partial permutation martingale D. There are two cases:

1. Either there exists an extension v of w such that D diverges on v, and such that the value of $\alpha_1 d_1 + \ldots + \alpha_k d_k$ still does not exceed 2 at v. In that case, we choose v as an initial segment of our sequence, as it both defeats D (no matter how we further extend v later) and keeps $\alpha_1 d_1 + \ldots + \alpha_k d_k$ low.
2. Or, if we are not in the first case, then D is total on the set T of sequences v such that $\alpha_1 d_1 + \ldots + \alpha_k d_k$ has value less than 2 at v and at all its prefixes. Notice that T is a computable tree, and thus defines an effectively closed set $\mathcal{C}$. By Proposition 16, we can therefore replace D by a total permutation strategy. Then, by Theorem 5, we can additionally make D monotonic. The martingale d_{k+1} we obtain can now be added to the other ones, with a coefficient α_{k+1} sufficiently small to ensure that $\alpha_1 d_1 + \ldots + \alpha_k d_k + \alpha_{k+1} d_{k+1}$ has value less than 2 at w, and go on with the diagonalization.

Here again, if we wait for a long time before introducing a new strategy, we can keep the Kolmogorov complexity low. However, we cannot keep it low *all the time* as in the above case 1, the string v cannot be found effectively (we cannot check that a strategy diverges on a string), and in particular it may have high Kolmogorov complexity. □

Now that we have assembled all our tools, we can easily prove the desired results.

Theorem 17. *The following statements hold.*

1. $\mathbf{PPR} \not\subseteq \mathbf{TIR}$
2. $\mathbf{TIR} \not\subseteq \mathbf{PMR}$
3. $\mathbf{PMR} \not\subseteq \mathbf{PPR}$

From these results it easily follows that in Figure 2 no inclusion holds except those indicated and those implied by transitivity.

Proof. 1. Choose a computable sequence $\{n_k\}_k$ fulfilling the requirements of Theorem 9 such that $\mathrm{C}(k) \leq \log\log n_k$ for all k. The members of this set then form a computable set S. Use Theorem 15 to construct a sequence $A \in \mathbf{PPR}$ such that $\mathrm{C}(A \upharpoonright n \mid n) < \log\log n$ at infinitely many places in S. We then have for infinitely many k

$$\mathrm{C}(A \upharpoonright n_k \mid k) \leq \mathrm{C}(A \upharpoonright n_k) \leq \mathrm{C}(A \upharpoonright n_k \mid n_k) + 2\log\log n_k \leq 3\log\log n_k,$$

so A cannot be in **TIR** according to Theorem 9.

	monotonic		permutation		injection
total	**TMR**	=	**TPR**	$\supsetneq$	**TIR**
	$\cup\!\!\upharpoonright$		$\cup\!\!\upharpoonright$		$\cup\!\!\upharpoonright$
partial	**PMR**	$\supsetneq$	**PPR**	$\supsetneq$	**PIR**

Fig. 2. Assembled class inclusion results

2. Follows immediately from Theorems 11 and 12.
3. Follows immediately from Theorems 13 and 14. □

Note: An extended version of this paper (with full proofs of theorems) can be found at `http://arxiv.org/pdf/0907.2324`.

References

1. Harry Buhrman, Dieter van Melkebeek, Kenneth Regan, D. Sivakumar, and Martin Strauss. A generalization of resource-bounded measure, with application to the BPP vs. EXP problem. *SIAM Journal of Computing*, 30(2):576–601, 2000.
2. Rod Downey and Denis Hirschfeldt. *Algorithmic Randomness*. Springer, to appear.
3. Bart Kastermans and Steffen Lempp. Comparing notions of randomness. Manuscript, 2008.
4. James Lathrop and Jack Lutz. Recursive computational depth. *Information and Computation*, 153(1):139–172, 1999.
5. Ming Li and Paul Vitányi. *Kolmogorov Complexity and Its Applications*. Springer, 2008.
6. Wolfgang Merkle. The Kolmogorov-Loveland stochastic sequences are not closed under selecting subsequences. *Journal of Symbolic Logic*, 68:1362–1376, 2003.
7. Wolfgang Merkle. The complexity of stochastic sequences. *Journal of Computer and System Sciences*, 74(3):350–357, 2008.
8. Wolfgang Merkle, Joseph S. Miller, André Nies, Jan Reimann, and Frank Stephan. Kolmogorov-Loveland randomness and stochasticity. *Annals of Pure and Applied Logic*, 138(1-3):183–210, 2006.
9. Joseph Miller and André Nies. Randomness and computability: open questions. *Bulletin of Symbolic Logic*, 12(3):390–410, 2006.
10. Andrei A. Muchnik, Alexei Semenov, and Vladimir Uspensky. Mathematical metaphysics of randomness. *Theoretical Computer Science*, 207(2):263–317, 1998.
11. André Nies. *Computability and Randomness*. Oxford University Press, 2009.

Weihrauch Degrees, Omniscience Principles and Weak Computability*

Vasco Brattka[1] and Guido Gherardi[2]

[1] Laboratory of Foundational Aspects of Computer Science, Department of Mathematics & Applied Mathematics, University of Cape Town, South Africa
Vasco.Brattka@uct.ac.za
[2] Dipartimento di Filosofia, Università di Bologna, Italy
Guido.Gherardi@unibo.it

Abstract. In this paper we study a reducibility that has been introduced by Klaus Weihrauch or, more precisely, a natural extension of this reducibility for multi-valued functions on represented spaces. We call the corresponding equivalence classes Weihrauch degrees and we show that the corresponding partial order induces a lower semi-lattice with the disjoint union of multi-valued functions as greatest lower bound operation. We show that parallelization is a closure operator for this semi-lattice and the parallelized Weihrauch degrees even form a lattice with the product of multi-valued functions as greatest lower bound operation. We show that the Medvedev lattice and hence the Turing upper semi-lattice can both be embedded into the parallelized Weihrauch lattice in a natural way. The importance of Weihrauch degrees is based on the fact that multi-valued functions on represented spaces can be considered as realizers of mathematical theorems in a very natural way and studying the Weihrauch reductions between theorems in this sense means to ask which theorems can be transformed continuously or computably into each other. This allows a new purely topological or computational approach to metamathematics that sheds new light on the nature of theorems. As crucial corner points of this classification scheme we study the limited principle of omniscience LPO, the lesser limited principle of omniscience LLPO and their parallelizations. We show that parallelized LLPO is equivalent to Weak Kőnig's Lemma and hence to the Hahn-Banach Theorem in this new and very strong sense. We call a multi-valued function weakly computable if it is reducible to the Weihrauch degree of parallelized LLPO and we present a new proof that the class of weakly computable operations is closed under composition. This proof is based on a computational version of Kleene's ternary logic. Moreover, we characterize weakly computable operations on computable metric spaces as operations that admit upper semi-computable compact-valued selectors and we show that any single-valued weakly computable operation is already computable in the ordinary sense.

* This project has been supported by the Italian Ministero degli Affari Esteri and the National Research Foundation of South Africa (NRF)

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 83-94
http://drops.dagstuhl.de/opus/volltexte/2009/2261

1 Introduction

The purpose of this paper is to propose a new computational approach to metamathematics that is based on the classification of mathematical theorems according to their computational content. Such an approach started with a classification of the Weihrauch degree of the Hahn-Banach Theorem in [1] and the intention here is to lay some careful foundations for further studies. In a following paper [2] we analyze certain choice principles and we present a case study with a classification on many theorems from analysis. This paper is only an extended abstract, but a full version with all definitions and proofs is available for the interested reader [3].

Essentially, the idea is to ask which theorems can be continuously or even computably transferred into each other. In order to give a meaningful interpretation to this idea we consider mathematical theorems as multi-valued operations $F : X \rightrightarrows Y$ that map certain input data X into certain output data Y. Such a perspective is very natural, since many theorems in mathematics are actually Π_2 theorems, hence they have the logical form

$$(\forall x \in X)(\exists y \in Y)\ (x, y) \in A$$

and one can just consider $F : X \rightrightarrows Y$ as a realizer or multi-valued Skolem function for this statement.

The appropriate technical tool to study whether two such potentially partial multi-valued functions $F :\subseteq X \rightrightarrows Y$ and $G :\subseteq X \rightrightarrows Y$ can be continuously or computably transferred into each other is Weihrauch reducibility. This is a reducibility that has been introduced by Klaus Weihrauch around 1990 in two unpublished papers [4, 5] and since then it has been studied by several others (see for instance [6–11, 1, 2, 12]).

Originally, this reducibility has been introduced for single-valued functions on Baire space. Basically, the idea is to say that F is *strongly Weihrauch reducible* to G, in symbols $F \leq_{\mathrm{sW}} G$, if there are computable (or alternatively continuous) functions H and K such that

$$F = H \circ G \circ K.$$

Thus, K acts as an input modification and H acts as an output modification. We will mainly consider the computable version of this reduction here since the positive reduction results are stronger. For negative results the topological version of the reduction is stronger and indeed reductions typically fail for continuity reasons. However, such topological results can usually be derived from computational results by relativization.

It turns out that the strong version of Weihrauch reducibility is slightly too strong for many purposes, since it distinguishes too many functions. For instance the identity cannot be reduced to a constant function in this way, since there is no way to feed the input through a constant function. This is the reason why the more important reducibility is the one where we say that F is *Weihrauch*

reducible to G, in symbols $F \leq_{\mathrm{W}} G$, if there are computable functions H and K such that

$$F = H \circ \langle \mathrm{id}, G \circ K \rangle.$$

Thus, the difference is that the input is fed through to the outer function H independently of G.

Weihrauch [4, 5] has already studied an extended version of his reducibility to sets $\mathcal{F}$ and $\mathcal{G}$ of functions on Baire space and $\mathcal{F}$ is called *Weihrauch reducible* to $\mathcal{G}$, in symbols $\mathcal{F} \leq_{\mathrm{W}} \mathcal{G}$, if there are computable functions H and K such that

$$(\forall G \in \mathcal{G})(\exists F \in \mathcal{F})\ F = H\langle \mathrm{id}, GK \rangle.$$

Here $\langle\ \rangle : \mathbb{N}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ denotes a computable standard pairing function [13]. That is, any function $G \in \mathcal{G}$ computes some function $F \in \mathcal{F}$ and the computation is performed uniformly with two fixed computable H and K. This extension of Weihrauch reducibility is related to ordinary Weihrauch reducibility exactly as Medvedev reducibility is related to Turing reducibility.

We use this concept to extend Weihrauch reducibility even further to multi-valued operations $f :\subseteq X \rightrightarrows Y$ on represented spaces X and Y. Roughly speaking, such an f is *Weihrauch reducible* to an analogous g, in symbols $f \leq_{\mathrm{W}} g$, if the set of realizers of f is reducible to the set of realizers of g in the above mentioned sense of Weihrauch reducibility for sets, i.e.

$$\{F : F \vdash f\} \leq_{\mathrm{W}} \{G : G \vdash g\}.$$

Here a single-valued F on Baire space is called a *realizer* of f, in symbols $F \vdash f$, if F computes a name $F(p)$ of some output value in $f(x)$, given some name p of x. This generalization of Weihrauch reducibility was introduced for single-valued functions in [10] and for multi-valued functions in [1]. We call the corresponding equivalence classes *Weihrauch degrees*.

Compared to strong Weihrauch reducibility, the ordinary version of Weihrauch reducibility has exactly the right degree of precision, it distinguishes exactly what should be distinguished computationally, but not more. Among all functions (with at least one computable point in the domain) the computable ones form the least degree. For the continuous version of Weihrauch reducibility exactly the continuous functions form the least degree (among all functions with non-empty domain).

2 Products, sums and parallelization

In this section we briefly summarize some of our results on some basic properties of Weihrauch reducibility and of Weihrauch degrees. In particular we investigate the product operation $f \times g$ and the direct sum $f \oplus g$ of multi-valued operations and we show that both operations are monotone with respect to Weihrauch reducibility. While the product preserves single-valuedness, the disjoint union does not and hence it requires multi-valuedness in order to be meaningful. Among other things the partial order on Weihrauch degrees induces a lower semi-lattice

with direct sums as greatest lower bounds. While the product operation is just the ordinary product operation of multi-valued functions, we define the direct sum as follows. For any two sets Y, Z we define the *direct sum* or *disjoint union* by $Y \oplus Z := (\{0\} \times Y) \cup (\{1\} \times Z)$.

Definition 1 (Direct sum). Let $f :\subseteq X \rightrightarrows Y$ and $g :\subseteq U \rightrightarrows V$ be multi-valued maps on represented spaces. Then the *direct sum* of these maps $f \oplus g :\subseteq X \times U \rightrightarrows Y \oplus V$ is defined by $(f \oplus g)(x, u) := (\{0\} \times f(x)) \cup (\{1\} \times g(u))$ for all $(x, u) \in \mathrm{dom}(f \oplus g) := \mathrm{dom}(f) \times \mathrm{dom}(g)$.

The first observation is that product and sum are both monotone operations in the sense that

- $f \leq_{\mathrm{W}} g$ and $f' \leq_{\mathrm{W}} g' \Longrightarrow f \times f' \leq_{\mathrm{W}} g \times g'$ and $f \oplus f' \leq_{\mathrm{W}} g \oplus g'$.

This monotonicity result guarantees that we can safely extend the product and direct sum operation to Weihrauch degrees. Other common properties of products and sums are that they are both associative and commutative on degrees. The identity is a neutral element with respect to products. An important difference between product and sum is that functions are not necessarily idempotent with respect to products, i.e. there are f such that $f \not\equiv_{\mathrm{W}} f \times f$, while idempotency is always given for sums. A crucial property of sums is that they yield the greatest lower bound with respect to Weihrauch reducibility.

Proposition 1 (Greatest lower bound). *Let f and g be multi-valued functions on represented spaces. Then $f \oplus g$ is the greatest lower bound of f and g with respect to Weihrauch reducibility $\leq_{\mathrm{W}}$ and strong Weihrauch reducibility $\leq_{\mathrm{sW}}$.*

Is there any multi-valued map that plays the role of a neutral element with respect to the sum operation? Naturally, this would have to be a multi-valued function with an empty set of realizers. One should note that this is not the nowhere defined function $f :\subseteq X \rightrightarrows Y$, since $\{F : F \vdash f\}$ is the set of all function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$. If we accept the Axiom of Choice, then clearly, a function without realizers does not exist and hence we add an extra object 0 to our structure with $\{F : F \vdash 0\} = \emptyset$. Weihrauch reducibility can straightforwardly be extended to multi-valued functions enriched by 0, just by using $\emptyset$ as the set of realizers of 0. We denote the Weihrauch degree of 0 by $\mathbf{0}$. We assume that we have a fixed underlying set of represented spaces $\mathcal{R}$ and we also assume that this set includes $(\mathbb{N}^{\mathbb{N}}, \mathrm{id})$ and that $\mathcal{R}$ is closed under products and direct sums.

Definition 2 (Set of Weihrauch degrees). Let $\mathcal{W}$ denote the set that contains the degree $\mathbf{0}$ and all Weihrauch degrees of all multi-valued operations $f :\subseteq X \rightrightarrows Y$ with at least one computable point in $\mathrm{dom}(f)$ and with represented spaces $X, Y \in \mathcal{R}$. By $\mathbf{1}$ we denote the degree of the computable functions in $\mathcal{W}$.

In the following theorem we collect all the structural properties of Weihrauch degrees that we have studied so far.

Theorem 1 (Weihrauch degrees). *The space $(\mathcal{W}, \leq_{\mathrm{W}})$ of Weihrauch degrees is a lower semi-lattice with least element $\mathbf{1}$ and greatest element $\mathbf{0}$ and with $\oplus$ as the greatest lower bound operation. In particular, $(\mathcal{W}, \oplus)$ is an idempotent monoid with neutral element $\mathbf{0}$. Moreover, $(\mathcal{W}, \times)$ is a monoid with neutral element $\mathbf{1}$.*

An important operation on functions is *parallelization* $\widehat{f}$, which means to take countably many copies of the function f in parallel, i.e.

$$\widehat{f}(x_0, x_1, x_2, ...) := f(x_0) \times f(x_1) \times f(x_2) \times ...$$

This operation forms a closure operator with respect to Weihrauch reducibility.

Proposition 2 (Parallelization). *Let f and g be multi-valued functions on represented spaces. Then*

1. $f \leq_{\mathrm{W}} \widehat{f}$ *(extensive)*
2. $f \leq_{\mathrm{W}} g \Longrightarrow \widehat{f} \leq_{\mathrm{W}} \widehat{g}$ *(increasing)*
3. $\widehat{f} \equiv_{\mathrm{W}} \widehat{\widehat{f}}$ *(idempotent)*

An analogous result holds for strong Weihrauch reducibility.

The fact that Weihrauch reducibility is a closure operator allows us to define a parallelized version of Weihrauch reducibility.

Definition 3 (Parallel reducibility). Let f and g be multi-valued operations on represented spaces. Then we say that f is *parallely Weihrauch reducible* to g, in symbols $f \leq_{\widehat{\mathrm{W}}} g$, if $\widehat{f} \leq_{\mathrm{W}} \widehat{g}$. We say that f is *parallely Weihrauch equivalent* to g, in symbols $f \equiv_{\widehat{\mathrm{W}}} g$, if $f \leq_{\widehat{\mathrm{W}}} g$ and $g \leq_{\widehat{\mathrm{W}}} f$ holds. We call the corresponding equivalence classes *parallel Weihrauch degrees*.

Parallel reducibility is compatible with products and sums in the following sense:

- $\widehat{f \times g} \equiv_{\mathrm{sW}} \widehat{f} \times \widehat{g}$ and $\widehat{f \oplus g} \leq_{\mathrm{sW}} \widehat{\widehat{f} \oplus \widehat{g}} \equiv_{\mathrm{sW}} \widehat{f} \oplus \widehat{g}$.

Moreover, parallel Weihrauch degrees are idempotent with respect to products, i.e. $\widehat{f} \equiv_{\mathrm{sW}} \widehat{f} \times \widehat{f}$. The idempotency of parallel Weihrauch degrees has the consequence that the product actually is the least upper bound operation for parallel Weihrauch degrees[3].

Proposition 3 (Least upper bound). *Let f and g be multi-valued functions on represented spaces. Then $f \times g$ is the least upper bound of f and g with respect to parallel Weihrauch reducibility $\leq_{\widehat{\mathrm{W}}}$.*

[3] Independently, Arno Pauly [12] has recently proved that another operation on functions that takes direct sums on the input and output side yields a supremum even in the non-parallelized case. He has also proved that the corresponding upper semilattice is distributive.

The parallelized Weihrauch degrees together with their partial order even form a lattice with the product as least upper bound operation. By $\widehat{\mathcal{W}}$ we denote the set of parallel Weihrauch degrees, which is defined as $\mathcal{W}$ but using parallel Weihrauch reducibility. As a corollary of our results we obtain that the parallel Weihrauch degrees of multi-valued functions form a lattice.

Theorem 2 (Parallel Weihrauch degrees). *The space $(\widehat{\mathcal{W}}, \leq_{\widehat{\mathrm{W}}})$ of parallel Weihrauch degrees is a lattice with least element $\mathbf{1}$ and greatest element $\mathbf{0}$, with $\oplus$ as the greatest lower bound operation and with $\times$ as the least upper bound operation. In particular, $(\widehat{\mathcal{W}}, \oplus)$ and $(\widehat{\mathcal{W}}, \times)$ are idempotent monoids with neutral elements $\mathbf{0}$ and $\mathbf{1}$, respectively.*

3 Embedding of Turing degrees and Medvedev degrees

Now we mention that the Medvedev lattice can be embedded into the Weihrauch lattice such that least upper bounds and greatest lower bounds are preserved. This embedding only requires total and continuous multi-valued operations on Baire space. As a consequence, we obtain that Turing degrees can be embedded such that least upper bounds are preserved and this embedding only requires total and continuous single-valued functions on Baire space.

We recall that a set $\mathcal{A} \subseteq \mathbb{N}^{\mathbb{N}}$ is said to be *Medvedev reducible* to $\mathcal{B} \subseteq \mathbb{N}^{\mathbb{N}}$, in symbols $\mathcal{A} \leq_{\mathrm{M}} \mathcal{B}$, if there exists a computable $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ with $\mathcal{B} \subseteq \mathrm{dom}(F)$ and $F(\mathcal{B}) \subseteq \mathcal{A}$. In fact, Turing reducibility is a special case, since $p \in \mathbb{N}^{\mathbb{N}}$ is said to be *Turing reducible* to $q \in \mathbb{N}^{\mathbb{N}}$, in symbols $p \leq_{\mathrm{T}} q$, if $\{p\} \leq_{\mathrm{M}} \{q\}$ (see [14]).

Now we associate to any $q \in \mathbb{N}^{\mathbb{N}}$ the *constant function* $c_q : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}, p \mapsto q$ for all $p \in \mathbb{N}^{\mathbb{N}}$. In the next step we associate a multi-valued function to any non-empty $\mathcal{A} \subseteq \mathbb{N}^{\mathbb{N}}$ by $c_{\mathcal{A}} : \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}, p \mapsto \mathcal{A}$ for all $p \in \mathbb{N}^{\mathbb{N}}$. Then $c_{\mathcal{A}}$ has a computable realizer if and only if $\mathcal{A}$ contains a computable member. To the empty set $\emptyset \subseteq \mathbb{N}^{\mathbb{N}}$ we associate $c_{\emptyset} := 0$, the special "multi-valued function" without realizer. We note that the function $c_{\mathcal{A}}$ is parallelizable, i.e. $c_{\mathcal{A}} \equiv_{\mathrm{W}} \widehat{c_{\mathcal{A}}}$. Our main result of this section is now the following theorem.

Theorem 3 (Embedding of Medvedev degrees). *Let $\mathcal{A}, \mathcal{B} \subseteq \mathbb{N}^{\mathbb{N}}$. Then $\mathcal{A} \leq_{\mathrm{M}} \mathcal{B} \iff c_{\mathcal{A}} \leq_{\mathrm{W}} c_{\mathcal{B}}$.*

It is clear that a corresponding embedding of Turing degrees follows, i.e. $p \leq_{\mathrm{T}} q \iff c_p \leq_{\mathrm{W}} c_q$. Now we want to show that our embedding of the Medvedev lattice preserves also greatest lower and least upper bounds. For sets $\mathcal{A}, \mathcal{B} \subseteq \mathbb{N}^{\mathbb{N}}$ one usually defines $\mathcal{A} \oplus \mathcal{B} := \{\langle p, q\rangle : p \in \mathcal{A} \text{ and } q \in \mathcal{B}\}$ and $\mathcal{A} \otimes \mathcal{B} := 0\mathcal{A} \cup 1\mathcal{B}$. The reader should note that product and sum are just swapped compared to the way we use these operations. Now one can easily show the following result.

Proposition 4. *Let $\mathcal{A}, \mathcal{B} \subseteq \mathbb{N}^{\mathbb{N}}$. Then $c_{\mathcal{A}\oplus\mathcal{B}} \equiv_{\mathrm{sW}} c_{\mathcal{A}} \times c_{\mathcal{B}}$ and $c_{\mathcal{A}\otimes\mathcal{B}} \equiv_{\mathrm{sW}} c_{\mathcal{A}} \oplus c_{\mathcal{B}}$.*

We mention that this result implies that our embedding of the Medvedev lattice preserves least upper bounds and greatest lower bounds.

Corollary 1 (Embedding of the Medvedev lattice). *The Medvedev lattice is embeddable into the parallel Weihrauch lattice (restricted to total and continuous multi-valued functions on Baire space and 0) with an embedding that preserves least upper bounds and greatest lower bounds.*

We also formulate the analogous result for Turing degrees.

Corollary 2 (Embedding of the Turing upper semi-lattice). *The upper semi-lattice of Turing degrees is embeddable into the parallel Weihrauch lattice (restricted to total and continuous single-valued functions on Baire space) with an embedding that preserves least upper bounds.*

Using these results some structural properties of the parallel Weihrauch lattice can be transferred from the Turing uppers semi-lattice and the Medvedev lattice. This observation also gives raise to plenty of further research questions.

4 Omniscience principles

In this section we study the the *limited principle of omniscience* LPO and the *lesser limited principle of omniscience* LLPO in the upper semi-lattice of Weihrauch reducibility. Such a study has also already been initiated by Weihrauch [5]. The principles themselves have originally been introduced by Brouwer and Bishop in constructive mathematics [15, 16]. Roughly speaking, LPO corresponds to the law of the excluded middle $(A \vee \neg A)$ and LLPO to de Morgan's law $\neg(A \wedge B) \iff (\neg A \vee \neg B)$, both restricted to simple existential statements. More precisely, they are stated as follows:

Definition 4 (Omniscience principles). We define:

- $\mathsf{LPO} : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}, \quad \mathsf{LPO}(p) = \begin{cases} 0 \text{ if } (\exists n \in \mathbb{N})\ p(n) = 0 \\ 1 \text{ otherwise} \end{cases},$
- $\mathsf{LLPO} :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}, \mathsf{LLPO}(p) \ni \begin{cases} 0 \text{ if } (\forall n \in \mathbb{N})\ p(2n) = 0 \\ 1 \text{ if } (\forall n \in \mathbb{N})\ p(2n+1) = 0 \end{cases},$

where $\mathrm{dom}(\mathsf{LLPO}) := \{p \in \mathbb{N}^{\mathbb{N}} : p(k) \neq 0 \text{ for at most one } k\}$.

One should notice that the definition of LLPO implies that $\mathsf{LLPO}(0^{\mathbb{N}}) = \{0, 1\}$. The two principles LPO and LLPO have already been studied in computable analysis [4–7]. For instance, it is well-known that LPO is reducible to any other discontinuous single-valued function on Baire space (see Lemma 8.2.6 in [13]).

Proposition 5. *Let $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ be discontinuous. Then we obtain $\mathsf{LPO} \leq_{\mathrm{sW}} F$, relatively to some oracle.*

While LPO is the "simplest" single-valued discontinuous function, its parallelization $\widehat{\mathsf{LPO}}$ is at the other end of the spectrum, it is complete among all $\mathbf{\Sigma}^0_2$–measurable functions with respect to the Borel hierarchy.

Similarly, as $\widehat{\mathsf{LPO}}$ is complete for the class of limit computable operations, we will show that $\widehat{\mathsf{LLPO}}$ is also complete for a very natural class of operations that we will call *weakly computable*. For technical simplicity by $\widehat{\mathsf{LLPO}}$ we actually mean

$$\widehat{\mathsf{LLPO}}\langle p_0, p_1, ...\rangle(k) \ni \begin{cases} 0 \text{ if } (\forall n)\ p_k(2n) = 0 \\ 1 \text{ if } (\forall n)\ p_k(2n+1) = 0 \end{cases}$$

One benefit of this understanding of $\widehat{\mathsf{LLPO}}$ is that it is composable with itself and the next observation is that the composition of $\widehat{\mathsf{LLPO}}$ with itself is strongly below itself. Roughly speaking this is because LLPO is defined only in terms of universal quantifiers and two consecutive universal quantifiers can be absorbed in one.

Lemma 1. $\widehat{\mathsf{LLPO}} \circ \widehat{\mathsf{LLPO}} \leq_{\mathrm{sW}} \widehat{\mathsf{LLPO}}$.

Using the NAND operation and Kleene's ternary logic we can show another interesting property of $\widehat{\mathsf{LLPO}}$, namely that it has some quasi-continuity property although it is discontinuous and we will exploit this property for our main result in this section. This result can also be interpreted as a completeness result for parallelized LLPO.

Theorem 4 (Completeness of parallelized LLPO). *For any computable function $F :\subseteq \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ there exists a computable function $G :\subseteq \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ such that $F \circ \widehat{\mathsf{LLPO}} = \widehat{\mathsf{LLPO}} \circ G$.*

If we combine the results showed so far, then we obtain that the multi-valued operations below $\widehat{\mathsf{LLPO}}$ are closed under composition. This has first been observed in [1], where it was expressed in terms of Weak Kőnig's Lemma (see also Corollary 3).

Proposition 6 (Composition). *Let $f :\subseteq X \rightrightarrows Y$ and $g :\subseteq Y \rightrightarrows Z$ be multi-valued operations on represented spaces. Then*

$$f \leq_{\mathrm{W}} \widehat{\mathsf{LLPO}} \text{ and } g \leq_{\mathrm{W}} \widehat{\mathsf{LLPO}} \Longrightarrow g \circ f \leq_{\mathrm{W}} \widehat{\mathsf{LLPO}}.$$

The same holds true with respect to some oracle (i.e. we can replace Weihrauch reducibility by its continuous counterpart in all occurrences here).

We believe that this result justifies to give a new name to the operations below $\widehat{\mathsf{LLPO}}$.

Definition 5 (Weakly computable). A function $f :\subseteq X \rightrightarrows Y$ on represented spaces X, Y is called *weakly computable*, if $f \leq_{\mathrm{W}} \widehat{\mathsf{LLPO}}$. Similarly, such a function is called *weakly continuous*, if $f \leq_{\mathrm{W}} \widehat{\mathsf{LLPO}}$ holds with respect to some oracle.

One main goal of this section is to show the following theorem on the omniscience principles. This theorem completely characterizes the relation of the omniscience principles and their parallelizations with respect to Weihrauch reducibility.

Theorem 5 (Omniscience principles). $\mathsf{LLPO} <_{\mathrm{W}} \mathsf{LPO} |_{\mathrm{W}} \widehat{\mathsf{LLPO}} <_{\mathrm{W}} \widehat{\mathsf{LPO}}$. *All negative results also hold true with respect to some arbitrary oracle.*

Note that the proof even shows the strong reduction $\mathsf{LLPO} \leq_{\mathrm{sW}} \mathsf{LPO}$. A different direct proof of $\mathsf{LPO} \not\leq_{\mathrm{W}} \mathsf{LLPO}$ is presented in Theorem 4.2 in [5].

Since any discontinuous single-valued function is already above LPO, it is clear that no such single-valued function can be below $\widehat{\mathsf{LLPO}}$. In other words, the parallel Weihrauch degree of LLPO has no single-valued member. In particular, this means that multi-valuedness does not appear accidentally in our theory, but in some sense it is unavoidable. Indeed we will show in Corollary 4 that any single-valued weakly computable function is already computable in the ordinary sense.

This has surprising algorithmic consequences. Any "algorithm" that uses weakly computable operations such as $x \leq 0$ or $x \geq 0$ leads to a computable result, as long as the result is uniquely determined, i.e. single-valued. And this is so, although these operations are typically discontinuous and non-computable.

5 Compact choice and Weak Kőnig's Lemma

In this section we will show that the parallel version of LLPO is equivalent to Weak Kőnig's Lemma. We first formalize Weak Kőnig's Lemma for this purpose. We recall that a *binary tree* is a subset $T \subseteq \{0,1\}^*$ that is closed under the prefix relation, i.e. if $w \in T$ and $v \sqsubseteq w$, then $v \in T$. We use some standard bijective enumeration $(w_n)_{n\in\mathbb{N}}$ of all the binary words. By Tr we denote the set of all binary trees and we use a representation δ_{Tr} of Tr that is defined by

$$\delta_{\mathrm{Tr}}(p) = T :\iff \chi_T(w_n) = p(n),$$

where $\chi_T : \{0,1\}^* \to \{0,1\}$ denotes the characteristic function of the binary tree T. The classical statement of Kőnig's Lemma is that any infinite binary tree has an infinite path. An infinite path of T is a sequence $p \in \{0,1\}^{\mathbb{N}}$, such that $p[n] \in T$ for all $n \in \mathbb{N}$. Here $p[n] = p(0)...p(n-1)$ is the prefix of p of length n. By $[T]$ the set of infinite paths of T is denoted. Now we can formalize Weak Kőnig's Lemma as follows.

Definition 6 (Weak Kőnig's Lemma). We define a multi-valued operation $\mathsf{WKL} :\subseteq \mathrm{Tr} \rightrightarrows \{0,1\}^{\mathbb{N}}, T \mapsto [T]$ with $\mathrm{dom}(\mathsf{WKL}) = \{T \subseteq \{0,1\}^* : T \text{ is an infinite binary tree}\}$.

Weak Kőnig's Lemma has already been studied in this form in [1]. Our main result here is that the parallel version of LLPO is strongly equivalent to Weak Kőnig's Lemma. For the proof we use Weak Kőnig's Lemma itself.

Theorem 6 (Weak Kőnig's Lemma). $\mathsf{WKL} \equiv_{\mathrm{sW}} \widehat{\mathsf{LLPO}}$.

In [1] it has been proved that the Hahn-Banach Theorem HBT has the same Weihrauch degree as WKL and hence the same Weihrauch degree as $\widehat{\mathsf{LLPO}}$. We formulate this as a corollary without exactly specifying HBT (the reader is referred to [1] for details).

Corollary 3. $\mathsf{HBT} \equiv_{\mathrm{W}} \mathsf{WKL} \equiv_{\mathrm{W}} \widehat{\mathsf{LLPO}}$.

Another equivalence that has been proved in [1] is that all the aforementioned theorems are equivalent to compact choice in rich spaces. We will use this observation and we adapt the formulation to our context.

Definition 7 (Compact choice). Let X be a computable metric space. The multi-valued operation $\mathsf{C}_{\mathcal{K}(X)} :\subseteq \mathcal{K}_-(X) \rightrightarrows X, A \mapsto A$ with $\mathrm{dom}(\mathsf{C}_{\mathcal{K}(X)}) := \{A \subseteq X : A \neq \emptyset\}$ is called *compact choice* of X.

Here $\mathcal{K}_-(X)$ denotes the set of compact subsets of X, which is equipped with the negative information representation κ_- (here a name of a compact set K is a list of all finite open rational covers of K, see [17] for details). In some sense, WKL is compact choice for the Cantor space $\{0,1\}^{\mathbb{N}}$ and, in fact, in [1] it has been proved that compact choice for a large class of computable metric space is equivalent to $\mathsf{C}_{\mathcal{K}(\{0,1\}^{\mathbb{N}})} \equiv_{\mathrm{W}} \mathsf{WKL}$. Using this result we mention a slightly different result here adapted to our operations.

Theorem 7 (Compact choice). *Let X be a computable metric space. Then $\mathsf{C}_{\mathcal{K}(X)} \leq_{\mathrm{sW}} \widehat{\mathsf{LLPO}}$. If X is rich, i.e. if there is a computable embedding $\iota : \{0,1\}^{\mathbb{N}} \hookrightarrow X$, then $\mathsf{C}_{\mathcal{K}(X)} \equiv_{\mathrm{sW}} \widehat{\mathsf{LLPO}}$.*

The characterization of $\widehat{\mathsf{LLPO}}$ as compact choice allows us to derive a characterization of weakly computable operations. Now we are prepared to show the characterization of weakly computable operations. We say that a function $s :\subseteq X \to \mathcal{K}_-(Y)$ is a *selector* of a function $f :\subseteq X \rightrightarrows Y$, if $\mathrm{dom}(s) = \mathrm{dom}(f)$ and $s(x) \subseteq f(x)$ for all $x \in \mathrm{dom}(f)$. Continuous functions $s :\subseteq X \to \mathcal{K}_-(Y)$ are also called *upper semi-continuous*.

Theorem 8 (Selection). *Let X be a represented space and let Y be a computable metric space. A function $f :\subseteq X \rightrightarrows Y$ is weakly computable if and only if f admits a computable selector $s :\subseteq X \to \mathcal{K}_-(Y)$.*

It is known that for computable metric spaces (Y, δ_Y) the singleton operation $Y \to \mathcal{K}_-(Y), y \mapsto \{y\}$ that maps a point to the corresponding singleton set is (δ_Y, κ_-)–computable and it admits a (κ_-, δ_Y)–computable right inverse (see for instance Lemma 6.4 in [18]). Thus we obtain the following corollary of the Selection Theorem 8.

Corollary 4 (Weakly computability). *Let X be a represented space and Y a computable metric space. Any weakly computable single-valued operation $f :\subseteq X \to Y$ is computable.*

Similarly, it follows that any weakly continuous single-valued function is already continuous in the ordinary sense.

6 Conclusions

In this paper we have studied Weihrauch reducibility of multi-valued functions on represented spaces. Among other things, we have proved that Weihrauch degrees form a lower semi-lattice with the direct sum operation as greatest lower bound operation. Moreover, we have studied parallelization as closure operator and we have shown that the parallelized Weihrauch degrees even form a lattice with the product as least upper bound operation. The Medvedev lattice and the upper semi-lattice of Turing degrees can be embedded into the parallelized Weihrauch lattice. Moreover, we have proved that the parallelized versions $\widehat{\mathsf{LPO}}$ and $\widehat{\mathsf{LLPO}}$ of the limited principle of omniscience and the lesser limited principle of omniscience, respectively, play a crucial role in our lattice. While $\widehat{\mathsf{LPO}}$ is complete for the class of limit computable operations, we have shown that $\widehat{\mathsf{LLPO}}$ can be used to define a meaningful class of weakly computable operations that is closed under composition. Single-valued weakly computable operations are already computable in the ordinary sense. This fact could be related to conservativeness properties of WKL_0 in reverse mathematics [19, 20] and to known uniqueness properties in constructive mathematics [21–24].

In a forthcoming paper [2] we discuss the classification of the Weihrauch degree of many theorems from analysis, such as the Intermediate Value Theorem, the Baire Category Theorem, the Banach Inverse Mapping Theorem and many others. It turns out that certain choice principles are crucial cornerstones for that classification and we believe that our classification sheds new light on the computational properties of these theorems. In particular, our classification seems to be in a well-defined sense finer than other known classifications in constructive and reverse mathematics.

References

1. Gherardi, G., Marcone, A.: How incomputable is the separable Hahn-Banach theorem? Notre Dame Journal of Formal Logic (to appear) extended abstract version in CCA 2008 proceedings ENTCS 221 (2008) 85–102.
2. Brattka, V., Gherardi, G.: Effective choice and boundedness principles in computable analysis. http://arxiv.org/abs/0905.4685 (preliminary version)
3. Brattka, V., Gherardi, G.: Weihrauch degrees, omniscience principles and weak computability. http://arxiv.org/abs/0905.4679 (preliminary version)
4. Weihrauch, K.: The degrees of discontinuity of some translators between representations of the real numbers. Technical Report TR-92-050, International Computer Science Institute, Berkeley (1992)
5. Weihrauch, K.: The TTE-interpretation of three hierarchies of omniscience principles. Informatik Berichte 130, FernUniversität Hagen, Hagen (1992)
6. Stein, T.v.: Vergleich nicht konstruktiv lösbarer Probleme in der Analysis. Diplomarbeit, Fachbereich Informatik, FernUniversität Hagen (1989)
7. Mylatz, U.: Vergleich unstetiger Funktionen in der Analysis. Diplomarbeit, Fachbereich Informatik, FernUniversität Hagen (1992)
8. Hertling, P.: Unstetigkeitsgrade von Funktionen in der effektiven Analysis. Informatik Berichte 208, FernUniversität Hagen, Hagen (1996) Dissertation.

9. Brattka, V.: Computable invariance. Theoretical Computer Science **210** (1999) 3–20
10. Brattka, V.: Effective Borel measurability and reducibility of functions. Mathematical Logic Quarterly **51**(1) (2005) 19–44
11. Mylatz, U.: Vergleich unstetiger Funktionen: "Principle of Omniscience" und Vollständigkeit in der C–hierarchie. PhD Thesis, Faculty for Mathematics and Computer Science, University Hagen, Germany (2006)
12. Pauly, A.: On the (semi)lattices induced by continuous reducibilities. http://arxiv.org/abs/0903.2177 (preliminary version)
13. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
14. Rogers, H.: Theory of Recursive Functions and Effective Computability. McGraw-Hill, New York (1967)
15. Bishop, E., Bridges, D.S.: Constructive Analysis. Volume 279 of Grundlehren der Mathematischen Wissenschaften. Springer, Berlin (1985)
16. Bridges, D., Richman, F.: Varieties of Constructive Mathematics. Volume 97 of London Mathematical Society Lecture Note Series. Cambridge University Press, Cambridge (1987)
17. Brattka, V., Presser, G.: Computability on subsets of metric spaces. Theoretical Computer Science **305** (2003) 43–76
18. Brattka, V.: Plottable real number functions and the computable graph theorem. SIAM Journal on Computing **38**(1) (2008) 303–328
19. Simpson, S.G.: Subsystems of Second Order Arithmetic. Perspectives in Mathematical Logic. Springer, Berlin (1999)
20. Simpson, S., Tanaka, K., Yamazaki, T.: Some conservation results on weak König's lemma. Annals of Pure and Applied Logic **118**(1-2) (2002) 87–114
21. Kohlenbach, U.: Effective moduli from ineffective uniqueness proofs. an unwinding of de La Vallé Poussin's proof for Chebycheff approximation. Annals of Pure and Applied Logic **64** (1993) 27–94
22. Schuster, P.: Unique solutions. Mathematical Logic Quarterly **52**(6) (2006) 534–539
23. Schuster, P.: Corrigendum to "unique solutions". Mathematical Logic Quarterly **53**(2) (2007) 214–214
24. Ishihara, H.: Unique existence and computability in constructive reverse mathematics. In Cooper, S.B., Löwe, B., Sorbi, A., eds.: Computation and Logic in the Real World. Volume 4497 of Lecture Notes in Computer Science., Berlin, Springer (2007) 368–377 Third Conference on Computability in Europe, CiE 2007, Siena, Italy, June 18-23, 2007.

Effective Choice and Boundedness Principles in Computable Analysis*

Vasco Brattka[1] and Guido Gherardi[2]

[1] Laboratory of Foundational Aspects of Computer Science, Department of Mathematics & Applied Mathematics, University of Cape Town, South Africa
Vasco.Brattka@uct.ac.za
[2] Dipartimento di Filosofia, Università di Bologna, Italy
Guido.Gherardi@unibo.it

Abstract. In this paper we study a new approach to classify mathematical theorems according to their computational content. Basically, we are asking the question which theorems can be continuously or computably transferred into each other? For this purpose theorems are considered via their realizers which are operations with certain input and output data. The technical tool to express continuous or computable relations between such operations is Weihrauch reducibility and the partially ordered degree structure induced by it. We have identified certain choice principles on closed sets which are cornerstones among Weihrauch degrees and it turns out that certain core theorems in analysis can be classified naturally in this structure. In particular, we study theorems such as the Intermediate Value Theorem, the Baire Category Theorem, the Banach Inverse Mapping Theorem, the Closed Graph Theorem and the Uniform Boundedness Theorem. Well-known omniscience principles from constructive mathematics such as LPO and LLPO can also naturally be considered as Weihrauch degrees and they play an important role in our classification. Our classification scheme does not require any particular logical framework or axiomatic setting, but it can be carried out in the framework of classical mathematics using tools of topology, computability theory and computable analysis. Finally, we present a number of metatheorems that allow to derive upper bounds for the classification of the Weihrauch degree of many theorems and we discuss the Brouwer Fixed Point Theorem as an example.

Keywords: Computable analysis, constructive analysis, reverse mathematics, effective descriptive set theory

* This project has been supported by the Italian Ministero degli Affari Esteri and the National Research Foundation of South Africa (NRF)

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 95-106
http://drops.dagstuhl.de/opus/volltexte/2009/2262

1 Introduction

The purpose of this paper is to propose a new approach to classify mathematical theorems according to their computational content and according to their logical complexity[3].

1.1 Realizability of theorems and Weihrauch reducibility

The basic idea is to interpret theorems, which are typically Π_2–theorems of the form

$$(\forall x \in X)(\exists y \in Y)(x, y) \in A,$$

as operations $F :\subseteq X \rightrightarrows Y, x \mapsto \{y \in Y : (x, y) \in A\}$ that map certain input data X into certain output data Y. In other words, we are representing theorems by their realizers or multi-valued Skolem functions, which is a very natural approach for many typical theorems. For instance, the Intermediate Value Theorem states that

$$(\forall f \in \mathcal{C}[0,1], f(0) \cdot f(1) < 0)(\exists x \in [0,1]) f(x) = 0$$

and hence it is natural to consider the partial multi-valued operation

$$\mathsf{IVT} :\subseteq \mathcal{C}[0,1] \rightrightarrows [0,1], f \mapsto \{x \in [0,1] : f(x) = 0\}$$

with $\mathrm{dom}(\mathsf{IVT}) := \{f \in \mathcal{C}[0,1] : f(0) \cdot f(1) < 0\}$ as a representative of this theorem. It follows from the Intermediate Value Theorem itself that this operation is well-defined. The goal of our study is to understand the computational content of theorems like the Intermediate Value Theorem and to analyze how they compare to other theorems. In order to understand the relation of two theorems T and T' to each other we will ask the question whether a realizer G of T' can be computably or continuously transformed into a realizer F of T. In other words, we consider theorems as points in a space (represented by their realizers) and we study whether these points can be computably or continuously transferred into each other. This study is carried out entirely in the domain of classical logic and using tools from topology, computability theory and computable analysis [17].

In fact the technical tool to express the relation of realizers to each other is a reducibility that Weihrauch introduced in the 1990s in two unpublished papers [15, 16] and which since then has been studied by several others (see for instance [11, 2, 3, 12, 10, 6, 13]). Basically, the idea is to say that a single-valued function F is Weihrauch reducible to G, in symbols $F \leq_{\mathrm{W}} G$, if there are computable function H and K such that

$$F = H\langle \mathrm{id}, GK\rangle.$$

Here K can be considered as an input adaption and H as an output adaption. The output adaption has direct access to the input, since in many cases the input

[3] This paper is only an extended abstract, but a full version with all definitions and proofs is available for the interested reader [7].

cannot be looped through G. Here and in the following $\langle\ \rangle$ denotes suitable finite or infinite tupling functions. This reducibility can be extended to sets of functions and to multi-valued functions on represented spaces. The resulting structure has been studied in [6] and among other things it has been proved that parallelization is a closure operator for Weihrauch reducibility. To parallelize a multi-valued function F just means to consider

$$\widehat{F}\langle p_0, p_1, p_2, ...\rangle := \langle F(p_0) \times F(p_1) \times F(p_2) \times ...\rangle,$$

i.e. to take countably many instances of F in parallel.

1.2 Effective choice and boundedness principles

A characterization of the Weihrauch degree of theorems is typically achieved by showing that the degree is identical to the degree of some other known principle. We have identified certain choice principles that turned out to be crucial cornerstones in our classification. These principles are *co-finite choice*, *discrete choice*, *interval choice*, *compact choice*, *closed choice*, and are exposed in Sect. 3.

Often it is more convenient to consider these choice principles as boundedness principles and in particular the principles of interval choice have equivalent boundedness versions. In Sect. 3 we will present some boundedness principles that correspond to the above mentioned choice principles.

In Sect. 3 we will show the equivalence of certain choice and boundedness principles and we will compare them to omniscience principles. Omniscience principles have been introduced by Brouwer and Bishop [1, 9] as non-acceptable principles in the intuitionistic framework of constructive analysis.

- (LPO) For any sequence $p \in \mathbb{N}^{\mathbb{N}}$ there exists an $n \in \mathbb{N}$ such that $p(n) = 0$ or $p(n) \neq 0$ for all $n \in \mathbb{N}$.
- (LLPO) For any sequence $p \in \mathbb{N}^{\mathbb{N}}$ such that $p(k) \neq 0$ for at most one $k \in \mathbb{N}$, it follows $p(2n) = 0$ for all $n \in \mathbb{N}$ or $p(2n+1) = 0$ for all $n \in \mathbb{N}$.

The abbreviations stand for *limited principle of omniscience* and *lesser limited principle of omniscience*. The realizers of these statements correspond to discontinuous operations of different degree of discontinuity [16].

The parallelizations $\widehat{\mathsf{LPO}}$ and $\widehat{\mathsf{LLPO}}$ turned out to be particularly important cornerstones in our classification scheme, since $\widehat{\mathsf{LPO}}$ is a $\mathbf{\Sigma}^0_2$–complete operation in the effective Borel hierarchy [3], i.e. it is complete among all limit computable operations with respect to Weihrauch reducibility and similarly $\widehat{\mathsf{LLPO}}$ is complete among all weakly computable operations [10, 6]. Limit computable operations are exactly the effectively $\mathbf{\Sigma}^0_2$–measurable operations and these are exactly those operations that can be computed on a Turing machine that is allowed to revise its output. We have defined weakly computable operations exactly by the above mentioned completeness property in [6]. In Sect. 3 we will show how the choice and boundedness principles are related to the omniscience principles and their parallelizations.

Figure 1 illustrates the relation between the choice principles and other results discussed in this paper.

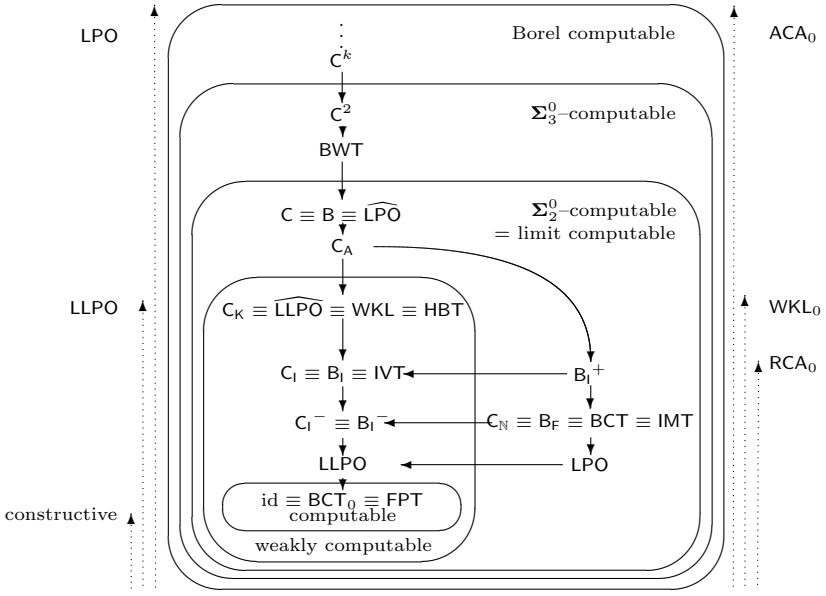


Fig. 1. Constructive, computable and reverse mathematics

1.3 Theorems in functional analysis

As a case study we analyze a number of theorems from analysis and functional analysis and we classify their Weihrauch degree. In particular, we will consider in Sections 4, 5, 6 and 7 the following theorems:

- (BCT_0) Given a sequence $(A_i)_{i\in\mathbb{N}}$ of closed nowhere dense subsets of a complete separable metric space X, there exists a point $x \in X \setminus \bigcup_{i\in\mathbb{N}} A_i$ (Baire Category Theorem).
- (BCT) Given a sequence $(A_i)_{i\in\mathbb{N}}$ of closed subsets of a complete separable metric space X with $X = \bigcup_{i=0}^{\infty} A_i$, there is some $n \in \mathbb{N}$ such that A_n is somewhere dense (Baire Category Theorem).
- (IMT) Any bijective linear bounded operator $T : X \to Y$ on separable Banach spaces X and Y has a bounded inverse $T^{-1} : Y \to X$ (Banach Inverse Mapping Theorem).
- (OMT) Any surjective linear bounded operator $T : X \to Y$ on separable Banach spaces X and Y is open, i.e. $T(U)$ is open for any open $U \subseteq X$ (Open Mapping Theorem).
- (CGT) Any linear operator $T : X \to Y$ with a closed graph$(T) \subseteq X \times Y$ is bounded (Closed Graph Theorem).
- (UBT) Any sequence $(T_i)_{i\in\mathbb{N}}$ of linear bounded operators that is pointwise bounded, i.e. such that $\sup\{||T_i x|| : i \in \mathbb{N}\}$ exists for all $x \in X$, is uniformly bounded, i.e. $\sup\{||T_i|| : i \in \mathbb{N}\}$ exists (Uniform Boundedness Theorem).

- (HBT) Any bounded linear functional $f : Y \to \mathbb{R}$, defined on some closed subspace Y of a Banach space X has a bounded linear extension $g : X \to \mathbb{R}$ with the same norm $||g|| = ||f||$ (Hahn-Banach Theorem).
- (IVT) For any continuous function $f : [0,1] \to \mathbb{R}$ with $f(0) \cdot f(1) < 0$ there exists a $x \in [0,1]$ with $f(x) = 0$ (Intermediate Value Theorem).
- (BFT) Any continuous function $f : [0,1]^n \to [0,1]^n$ has a fixed point $x \in [0,1]^n$, i.e. $f(x) = x$ (Brouwer Fixed Point Theorem).
- (BWT) Any sequence $(x_i)_{i\in\mathbb{N}}$ of numbers in $[0,1]^n$ has a convergent subsequence (Bolzano-Weierstraß Theorem).
- (WAT) For any continuous function $f : [0,1] \to \mathbb{R}$ and any $n \in \mathbb{N}$ there exists a rational polynomial $p \in \mathbb{Q}[x]$ such that $||f - p|| = \sup_{x\in[0,1]} |f(x) - p(x)| < 2^{-n}$ (Weierstraß Approximation Theorem).
- (WKL) Any infinite binary tree has an infinite path (Weak Kőnig's Lemma).

The Baire Category Theorem is an example of a theorem for which it matters which version is realized. In the formulation BCT_0 it leads to a continuous and even computable realizer, whereas the version BCT is discontinuous. The realizers of the given theorems are operations of different degree of discontinuity and our aim is classify the computational Weihrauch degree of these results. The benefit of such a classification is that practically all purely computability theoretic questions of interest about a theorem in computable analysis can be answered by such a classification. Typical questions are:

1. Is the theorem uniformly computable, i.e. can we compute the output information $y \in Y$ uniformly from the input information $x \in X$?
2. Is the theorem non-uniformly computable, i.e. does there exist a computable output information $y \in Y$ for any computable input information $x \in X$?
3. If there is no uniform solution, is there a uniform computation of a certain effective Borel complexity?
4. If there is no non-uniform computable solution, is there always a non-uniform result of a certain arithmetical complexity or Turing degree?

Answers to questions of this type can be derived from the classification of the Weihrauch degree of a theorem. In the diagram of Fig. 1 we summarize some of our results. The arrows in the diagram are pointing into the direction of computations and implicit logical implications and hence in the inverse direction of the corresponding reductions. No arrow in the diagram can be inverted and no arrows can be added (except those that follow by transitivity).

In Sect. 7 we provide a number of metatheorems that allow to determine upper bounds of the Weihrauch degree of many theorems straightforwardly, just because of the mere topological form of the statement. For instance, any classical result of the form

$$(\forall x \in X)(\exists y \in Y)(x, y) \in A$$

with a co-c.e. closed $A \subseteq X \times Y$ and a co-c.e. compact Y has a realizer that is reducible to compact choice C_K. The table in Fig. 2 summarizes the topological types of metatheorems and the corresponding version of computability. We il-

metatheorem	computability	unique case
open	computable	computable
compact	weakly computable	computable
locally compact	limit computable	non-uniformly computable

Fig. 2. Types of metatheorems, choice and computability

lustrate that these metatheorems are useful and we show that one directly gets upper bounds for theorems such as the Brouwer Fixed Point Theorem and the Peano Existence Theorem for the initial value problem of ordinary differential equations.

2 Weihrauch reducibility, omniscience principles and weak computability

In this section we briefly recall some definitions from [6] on Weihrauch reducibility. We assume that the reader has some basic familiarity with concepts from computable analysis and otherwise we refer the reader for all undefined concepts to [17]. In a first step we define Weihrauch reducibility for sets of functions on Baire space, as it was already considered by Weihrauch [15, 16].

Definition 1 (Weihrauch reducibility). Let $\mathcal{F}$ and $\mathcal{G}$ be sets of functions of type $f :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$. We say that $\mathcal{F}$ is *Weihrauch reducible* to $\mathcal{G}$, in symbols $\mathcal{F} \leq_{\mathrm{W}} \mathcal{G}$, if there are computable functions $H, K :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ such that

$$(\forall G \in \mathcal{G})(\exists F \in \mathcal{F})\ F = H\langle \mathrm{id}, GK\rangle.$$

Analogously, we define $\mathcal{F} \leq_{\mathrm{sW}} \mathcal{G}$ using the equation $F = HGK$ and in this case we say that $\mathcal{F}$ is *strongly Weihrauch reducible* to $\mathcal{G}$.

We denote the induced equivalence relations by $\equiv_{\mathrm{W}}$ and $\equiv_{\mathrm{sW}}$, respectively.

In the next step we define the concept of a realizer of a multi-valued function as it is used in computable analysis [17]. We recall that a *representation* $\delta_X :\subseteq \mathbb{N}^{\mathbb{N}} \to X$ of a set X is a surjective (and potentially partial) map. In general, the inclusion symbol "$\subseteq$" indicates partiality in this paper. In this situation we say that (X, δ_X) is a *represented space*.

Definition 2 (Realizer). Let (X, δ_X) and (Y, δ_Y) be represented spaces and let $f :\subseteq X \rightrightarrows Y$ be a multi-valued function. Then $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is called *realizer* of f with respect to (δ_X, δ_Y), in symbols $F \vdash f$, if

$$\delta_Y F(p) \in f\delta_X(p)$$

for all $p \in \mathrm{dom}(f\delta_X)$.

Usually, we do not mention the representations explicitly since they will be clear from the context. A multi-valued function $f :\subseteq X \rightrightarrows Y$ on represented

spaces is called *continuous* or *computable*, if it has a continuous or computable realizer, respectively. Using reducibility for sets and the concept of a realizer we can now define Weihrauch reducibility for multi-valued functions.

Definition 3 (Realizer reducibility). Let f and g be multi-valued functions on represented spaces. Then f is said to be *Weihrauch reducible* to g, in symbols $f \leq_{\mathrm{W}} g$, if and only if $\{F : F \vdash f\} \leq_{\mathrm{W}} \{G : G \vdash g\}$. Analogously, we define $f \leq_{\mathrm{sW}} g$ with the help of $\leq_{\mathrm{sW}}$ on sets.

That is, $f \leq_{\mathrm{W}} g$ holds if any realizer of g computes some realizer of f with some fixed uniform translations H and K. It is clear that Weihrauch reducibility and its strong version form preorders, i.e. both relations are reflexive and transitive.

One can show that the product of multi-valued functions $f \times g$ and the direct sum $f \oplus g$ are both monotone operations with respect to strong and ordinary Weihrauch reducibility and hence both operations can be extended to Weihrauch degrees. This turns the structure of partially ordered Weihrauch degrees into a lower-semi lattice with the direct sum operation as greatest lower bound operation. It turns out that a very important operation on this lower semi-lattice is parallelization, which can be understood as countably infinite product operation.

Definition 4 (Parallelization). Let $f :\subseteq X \rightrightarrows Y$ be a multi-valued function. Then we define the *parallelization* $\widehat{f} :\subseteq X^{\mathbb{N}} \rightrightarrows Y^{\mathbb{N}}$ of f by

$$\widehat{f}(x_i)_{i\in\mathbb{N}} := \bigtimes_{i=0}^{\infty} f(x_i)$$

for all $(x_i)_{i\in\mathbb{N}} \in X^{\mathbb{N}}$.

We mention that parallelization acts as a closure operator with respect to Weihrauch reducibility.

3 Choice and boundedness principles

In this section we study choice principles and boundedness principles. Both types of principles are closely related to each other and they are also related to the omniscience principles mentioned earlier. In some sense most of the boundedness principles are just variants of the choice principles that are more convenient for some applications.

By $\mathcal{A}(X)$ or $\mathcal{A}_-(X)$ we denote the set of closed subsets of a metric space X. The index "$-$" indicates that we assume that the hyperspace $\mathcal{A}_-(X)$ is equipped with the lower Fell topology and a corresponding negative information representation ψ_- (see [8] for details). All choice principles are restrictions of the multi-valued choice map

$$\mathrm{Choice} :\subseteq \mathcal{A}_-(X) \rightrightarrows X, A \mapsto A,$$

which is defined for non-empty closed sets $A \subseteq X$ and maps any such set in a multi-valued way to the set of its members. That is, the input is a non-empty closed set $A \in \mathcal{A}_-(X)$ and the output is one of the (possibly many) points $x \in A$. We can define restrictions of the choice map by specifying the respective domains and ranges.

Definition 5 (Choice principles). We define multi-valued operations as restrictions of the respective choice maps as follows:

1. $\mathsf{C}_\mathsf{F} :\subseteq \mathcal{A}_-(\mathbb{N}) \rightrightarrows \mathbb{N}$, $\mathrm{dom}(\mathsf{C}_\mathsf{F}) := \{A \subseteq \mathbb{N} : A \text{ co-finite}\}$.
2. $\mathsf{C}_\mathbb{N} :\subseteq \mathcal{A}_-(\mathbb{N}) \rightrightarrows \mathbb{N}$, $\mathrm{dom}(\mathsf{C}_\mathbb{N}) := \{A \subseteq \mathbb{N} : A \neq \emptyset\}$.
3. $\mathsf{C}_\mathsf{I} :\subseteq \mathcal{A}_-[0,1] \rightrightarrows [0,1]$, $\mathrm{dom}(\mathsf{C}_\mathsf{I}) := \{[a,b] : 0 \leq a \leq b \leq 1\}$.
4. $\mathsf{C}_\mathsf{I}^- :\subseteq \mathcal{A}_-[0,1] \rightrightarrows [0,1]$, $\mathrm{dom}(\mathsf{C}_\mathsf{I}^-) := \{[a,b] : 0 \leq a < b \leq 1\}$.
5. $\mathsf{C}_\mathsf{K} :\subseteq \mathcal{A}_-([0,1]) \rightrightarrows [0,1]$, $\mathrm{dom}(\mathsf{C}_\mathsf{K}) := \{K \subseteq [0,1] : K \neq \emptyset \text{ compact}\}$.
6. $\mathsf{C}_\mathsf{A} :\subseteq \mathcal{A}_-(\mathbb{R}) \rightrightarrows \mathbb{R}$, $\mathrm{dom}(\mathsf{C}_\mathsf{A}) := \{A \subseteq \mathbb{R} : A \neq \emptyset \text{ closed}\}$.

We refer to these operations as *co-finite choice*, *discrete choice*, *interval choice*, *proper interval choice*, *compact choice* and *closed choice*, respectively.

For practical purposes it is often more convenient to handle these choice principles in form of the closely related boundedness principles that we define now.

Definition 6 (Boundedness principles). We define the following multi-valued operations:

1. $\mathsf{B}_\mathsf{F} : \mathbb{R}_< \rightrightarrows \mathbb{R}, x \mapsto [x, \infty)$.
2. $\mathsf{B}_\mathsf{I} :\subseteq \mathbb{R}_< \times \mathbb{R}_> \rightrightarrows \mathbb{R}, (x,y) \mapsto [x,y]$, $\mathrm{dom}(\mathsf{B}_\mathsf{I}) := \{(x,y) : x \leq y\}$.
3. $\mathsf{B}_\mathsf{I}^- :\subseteq \mathbb{R}_< \times \mathbb{R}_> \rightrightarrows \mathbb{R}, (x,y) \mapsto [x,y]$, $\mathrm{dom}(\mathsf{B}_\mathsf{I}^-) := \{(x,y) : x < y\}$.
4. $\mathsf{B}_\mathsf{I}^+ :\subseteq \mathbb{R}_< \times \overline{\mathbb{R}_>} \to \mathbb{R}, (x,y) \mapsto [x,y]$, $\mathrm{dom}(\mathsf{B}_\mathsf{I}^+) := \{(x,y) : x \leq y\}$.
5. $\mathsf{B} : \mathbb{R}_< \to \mathbb{R}, x \mapsto x$.

Proposition 1 (Discrete choice). $\mathsf{B}_\mathsf{F} \equiv_\mathrm{sW} \mathsf{C}_\mathsf{F} \equiv_\mathrm{W} \mathsf{C}_\mathbb{N}$.

Proposition 2 (Interval choice). $\mathsf{B}_\mathsf{I} \equiv_\mathrm{sW} \mathsf{C}_\mathsf{I}$, $\mathsf{B}_\mathsf{I}^- \equiv_\mathrm{sW} \mathsf{C}_\mathsf{I}^-$, $\mathsf{B}_\mathsf{I}^+ \leq_\mathrm{sW} \mathsf{C}_\mathsf{A}$.

We recall that it is known that B is equivalent to $\mathsf{C} : A \mapsto \mathrm{cf}_A$, $\mathrm{dom}(\mathsf{C}) = \mathcal{A}_-(\mathbb{N})$ (which can be considered as countable closed choice).

Proposition 3 (Countable closed choice). $\mathsf{B} \equiv_\mathrm{W} \mathsf{C} \equiv_\mathrm{W} \widehat{\mathsf{LPO}}$.

We have identified two chains of choice principles that are related in the given way.

Corollary 1 (Choice hierarchies). *We obtain*

1. $\mathsf{LLPO} <_\mathrm{W} \mathsf{C}_\mathsf{I}^- <_\mathrm{W} \mathsf{C}_\mathsf{I} <_\mathrm{W} \mathsf{C}_\mathsf{K} \equiv_\mathrm{W} \widehat{\mathsf{LLPO}} \leq_\mathrm{W} \mathsf{C}_\mathsf{A}$.
2. $\mathsf{LPO} <_\mathrm{W} \mathsf{C}_\mathbb{N} <_\mathrm{W} \mathsf{B}_\mathsf{I}^+ <_\mathrm{W} \mathsf{C}_\mathsf{A} <_\mathrm{W} \mathsf{C} \equiv_\mathrm{W} \widehat{\mathsf{LPO}}$.
3. $\mathsf{LLPO} <_\mathrm{W} \mathsf{LPO}$, $\mathsf{C}_\mathsf{I}^- <_\mathrm{W} \mathsf{C}_\mathbb{N}$, $\mathsf{C}_\mathsf{I} <_\mathrm{W} \mathsf{B}_\mathsf{I}^+$.

Corollary 2 (Countable choice principles). *We obtain the following two equivalence classes:* $\widehat{\mathsf{LLPO}} \equiv_\mathrm{W} \widehat{\mathsf{C}_\mathsf{I}^-} \equiv_\mathrm{W} \widehat{\mathsf{C}_\mathsf{I}} \equiv_\mathrm{W} \widehat{\mathsf{C}_\mathsf{K}} <_\mathrm{W} \widehat{\mathsf{LPO}} \equiv_\mathrm{W} \widehat{\mathsf{C}_\mathbb{N}} \equiv_\mathrm{W} \widehat{\mathsf{C}_\mathsf{A}}$.

4 Discrete Choice and the Baire Category Theorem

In this section we want to classify the Weihrauch degree of the Baire Category Theorem and some core theorems from functional analysis such as the Banach Inverse Mapping Theorem, the Open Mapping Theorem, the Closed Graph Theorem and the Uniform Boundedness Theorem.

Theorem 1 (Baire Category Theorem). *Let X be a non-empty complete computable metric space. Then* $\mathsf{BCT}_X \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

Theorem 2 (Banach Inverse Mapping Theorem). *Let X, Y be computable Banach spaces. Then* $\mathsf{IMT}_{X,Y} \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{IMT}_{\ell_2,\ell_2}$.

Theorem 3 (Open Mapping Theorem). *Let X, Y be computable Banach spaces. Then* $\mathsf{OMT}_{X,Y} \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{OMT}_{\ell_2,\ell_2}$.

Theorem 4 (Closed Graph Theorem). *Let X, Y be computable Banach spaces. Then* $\mathsf{CGT}_{X,Y} \leq_{\mathrm{W}} \mathsf{C}_{\mathbb{N}} \equiv_{\mathrm{W}} \mathsf{CGT}_{\ell_2,\ell_2}$.

Theorem 5 (Uniform Boundedness Theorem). *Let X, Y be computable Banach spaces different from $\{0\}$. Then* $\mathsf{UBT}_{X,Y} \equiv_{\mathrm{W}} \mathsf{C}_{\mathbb{N}}$.

A common feature of all the theorems discussed in this section that are equivalent to $\mathsf{C}_{\mathbb{N}}$ are:

1. They are discontinuous and hence non-computable (since $\mathsf{C}_{\mathbb{N}}$ is so).
2. They admit non-uniform computable solutions (since $\mathsf{C}_{\mathbb{N}}$ has a realizer that maps computable inputs to computable outputs).
3. They have Δ^0_2–complete sequential counterexamples (since $\widehat{\mathsf{C}_{\mathbb{N}}} \equiv_{\mathrm{W}} \mathsf{C}$, any realizer maps some computable sequence to some Δ^0_2–complete sequence in the arithmetical hierarchy).

All the properties mentioned here are degree theoretic properties and any theorem equivalent to $\mathsf{C}_{\mathbb{N}}$ will be of the same category.

5 Interval Choice and the Intermediate Value Theorem

Theorem 6 (Intermediate Value Theorem). $\mathsf{IVT} \equiv_{\mathrm{sW}} \mathsf{C}_{\mathsf{I}}$.

We list some common features of all theorems that are equivalent to C_{I}.

1. They are discontinuous and hence non-computable (since C_{I} is so).
2. They admit non-uniform computable solutions (since C_{I} has a realizer that maps computable inputs to computable outputs).
3. They are uniformly computable under all classical conditions where the solution is uniquely determined (since C_{I} is weakly computable).
4. They have limit computable sequential counterexamples of any basis type (since $\widehat{\mathsf{C}_{\mathsf{I}}} \equiv_{\mathrm{W}} \mathsf{WKL}$).

By a basis type we mean any set $B \subseteq \mathbb{N}^{\mathbb{N}}$ that forms a basis for Π^0_1 subsets of Cantor space $\{0,1\}^{\mathbb{N}}$, such as the set of low points.

6 Compact Choice and the Hahn-Banach Theorem

Theorem 7 (Hahn-Banach Theorem). $\mathsf{HBT} \equiv_{\mathrm{W}} \mathsf{C}_{\mathsf{K}}$.

Common features of all theorems equivalent to C_{K} are:

1. They are discontinuous and hence non-computable (since C_{K} is so).
2. They are uniformly computable under all classical conditions where the solution is uniquely determined (since C_{K} is weakly computable).
3. They have limit computable counterexamples of any basis type (since we have that $\mathsf{C}_{\mathsf{K}} \equiv_{\mathrm{W}} \mathsf{WKL}$).

7 Metatheorems and Applications

In this section we want to discuss a number of metatheorems that allow some conclusions on the status of theorems merely regarding the logical form of these theorems. Essentially, we are trying to identify the computational status of Π_2–theorems, i.e. theorems of the form

$$(\forall x \in X)(\exists y \in Y)\ (x, y) \in A,$$

where depending on the properties of Y and A automatically certain computable versions of realizers of these theorems exist. In many cases this allows to get some upper bound on the Weihrauch degree of the corresponding theorem straightforwardly.

Theorem 8 (Open Metatheorem). *Let X, Y be computable metric spaces and let $U \subseteq X \times Y$ be c.e. open. If*

$$(\forall x \in X)(\exists y \in Y)(x, y) \in U,$$

then $R : X \rightrightarrows Y, x \mapsto \{y \in Y : (x, y) \in U\}$ is computable.

Corollary 3 (Weierstraß Approximation Theorem). $\mathsf{WAT} \equiv_{\mathrm{W}} \mathrm{id}$.

The next metatheorem is a similar observation for co-c.e. closed predicates and co-c.e. compact Y.

Theorem 9 (Compact Metatheorem). *Let X, Y be computable metric spaces and let Y be co-c.e. compact and $A \subseteq X \times Y$ co-c.e. closed. If*

$$(\forall x \in X)(\exists y \in Y)(x, y) \in A,$$

then $R : X \rightrightarrows Y, x \mapsto \{y \in Y : (x, y) \in A\}$ is weakly computable, i.e. $R \leq_{\mathrm{W}} \mathsf{C}_{\mathsf{K}}$.

Corollary 4 (Brouwer Fixed Point Theorem). $\mathsf{BFT} \leq_{\mathrm{W}} \mathsf{WKL}$.

Many other theorems of analysis that have to do with the solution of equations in compact spaces fall into the same category. This applies for instance to the Schauder Fixed Point Theorem and also to the Intermediate Value Theorem. Sometimes it is not immediately clear that a theorem is of this form. In case of the Peano Existence Theorem for solutions of initial value problems of ordinary differential equations it is easy to see that it can be reduced to the Schauder Fixed Point Theorem (see [14]). Another example of this type is the Hahn-Banach Theorem. As it is usually formulated, is not of the form of an equation with a solution in a compact space. However, using the Banach-Alaoglu Theorem, it can be brought into this form (see [4, 10]). Whenever a theorem that falls under the Compact Metatheorem has a unique solution, then that solution is automatically computable by Corollary 8.8 in [6].

Thus, under all (perhaps purely classical) conditions under which the Brouwer Fixed Point Theorem, the Intermediate Valued Theorem, the Hahn-Banach Theorem or the Peano Existence Theorem have unique solutions, they are already automatically fully computable.

Theorem 10 (Locally Compact Metatheorem). *Let X, Y be computable metric spaces, let Y be effectively locally compact and let $A \subseteq X \times Y$ be co-c.e. closed. If*

$$(\forall x \in X)(\exists y \in Y)(x, y) \in A,$$

then $R : X \rightrightarrows Y, x \mapsto \{y \in Y : (x, y) \in A\}$ satisfies $R \leq_{\mathrm{W}} \mathsf{C}_{\mathcal{A}(Y)}$, where $\mathsf{C}_{\mathcal{A}(Y)}$ is defined like C_A with Y instead of $\mathbb{R}$. In particular, R is limit computable.

Corollary 5. *Let X be a computable metric space and let Y be an effectively locally compact metric space. If $f : X \to Y$ is a function with a co-c.e. closed graph $\mathrm{graph}(f) = \{(x, y) \in X \times Y : f(x) = y\}$, then f is limit computable. In particular, the inverse $g^{-1} : X \to Y$ of any computable bijective function $g : Y \to X$ is limit computable.*

References

1. Bishop, E., Bridges, D. S.: Constructive Analysis. Volume 279 of Grundlehren der Mathematischen Wissenschaften. Springer, Berlin (1985)
2. Brattka, V.: Computable invariance. Theoretical Computer Science **210** (1999) 3–20
3. Brattka, V.: Effective Borel measurability and reducibility of functions. Mathematical Logic Quarterly **51**(1) (2005) 19–44
4. Brattka, V.: Borel complexity and computability of the Hahn-Banach Theorem. Archive for Mathematical Logic **46**(7–8) (2008) 547–564
5. Brattka, V., Gherardi, G.: Borel complexity of topological operations on computable metric spaces. Journal of Logic and Computation **19**(1) (2009) 45–76
6. Brattka, V., Gherardi, G.: Weihrauch degrees, omniscience principles and weak computability. http://arxiv.org/abs/0905.4679 (preliminary version)
7. Brattka, V., Gherardi, G.: Effective choice and boundedness principles in computable analysis. http://arxiv.org/abs/0905.4685 (preliminary version)
8. Brattka, V., Presser, G.: Computability on subsets of metric spaces. Theoretical Computer Science **305** (2003) 43–76

9. Bridges, D. S., Richman, F.: Varieties of Constructive Mathematics. Volume 97 of London Mathematical Society Lecture Note Series. Cambridge University Press, Cambridge (1987)
10. Gherardi, G., Marcone, A.: How incomputable is the separable Hahn-Banach theorem? Notre Dame Journal of Formal Logic (to appear). Extended abstract version in CCA 2008 Proceedings ENTCS **221** (2008) 85–102
11. Hertling, P.: Unstetigkeitsgrade von Funktionen in der effektiven Analysis. Dissertation. Volume 208 of Informatik Berichte 208. FernUniversität Hagen, Hagen (1996)
12. Mylatz, U.: Vergleich unstetiger Funktionen: "Principle of Omniscience" und Vollständigkeit in der C–hierarchie. PhD Thesis. Faculty for Mathematics and Computer Science. University Hagen. Germany (2006)
13. Pauly, A.: On the (semi)lattices induced by continuous reducibilities. http://arxiv.org/abs/0903.2177 (preliminary version)
14. Simpson, S. G.: Subsystems of Second Order Arithmetic. Perspectives in Mathematical Logic. Springer, Berlin (1999)
15. Weihrauch, K.: The degrees of discontinuity of some translators between representations of the real numbers. Technical Report TR-92-050. International Computer Science Institute. Berkeley. July (1992)
16. Weihrauch, K.: The TTE-interpretation of three hierarchies of omniscience principles. Volume 130 of Informatik Berichte FernUniversität Hagen, Hagen (1992)
17. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)

Computability of Homology for Compact Absolute Neighbourhood Retracts

Pieter Collins *

Centrum Wiskunde en Informatica,
Postbus 94079, 1090 GB Amsterdam,
The Netherlands
pieter.collins@cwi.nl

Abstract. In this note we discuss the information needed to compute the homology groups of a topological space. We argue that the natural class of spaces to consider are the compact absolute neighbourhood retracts, since for these spaces the homology groups are finite. We show that we need to specify both a function which defines a retraction from a neighbourhood of the space in the Hilbert cube to the space itself, and a sufficiently fine over-approximation of the set. However, neither the retraction itself, nor a description of an approximation of the set in the Hausdorff metric, is sufficient to compute the homology groups. We express the conditions in the language of computable analysis, which is a powerful framework for studying computability in topology and geometry, and use cubical homology to perform the computations.

Keywords: computability, homology, compact absolute neighbourhood retract

1 Introduction

Homology theory is one of the cornerstones of algebraic topology. The first homology theory, simplicial homology, was developed to provide invariants of a topological space (expressed as a simplicial complex) which could be more easily computed than the homotopy invariants. Other homology theories, most notably singular homology, were developed which extended the simplicial homology to arbitrary topological spaces, topological pairs and continuous functions. For an introduction to homology theory, see [ES52], [Mun84], [Mas91] or [Spa81]. However, while the simplicial homology can be easily computed by purely algebraic means, it is not clear precisely what information is needed about a space in order to compute its homology groups using a digital computer. The purpose of this article is to discuss the computability of homology for general metric spaces.

As is standard in computability theory, we use Turing machines as the underlying computational model. We consider different representations of the input

* This research was partially supported by Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO) Vidi grant 639.032.408.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 107-118
http://drops.dagstuhl.de/opus/volltexte/2009/2263

sets and/or functions in terms of symbols over some alphabet. Since the class of compact subsets of a (infinite) separable metric space has continuum cardinality, we need to represent these sets by streams of data, yielding successively better approximations to the set.

Since homology groups are well defined (by the Eilenburg-Steenrod axioms [ES45]) and finite for the class of compact absolute neighbourhood retracts, we restrict attention to these spaces. A natural way of describing a compact absolute neighbourhood retract is to specify a neighbourhood retraction onto the set. However, we shall see that this information itself is not quite sufficient to compute the homology; we also need to give a bounding set for the set which is a subset of the domain of the retraction.

The original approach to homology theory via simplicial complexes is well-suited to the computation of the homology of a topological space when an explicit construction of the space is known. However, it is less-well suited for the computation of the homology of an arbitrary continuous function, unless a homotopic simplicial map can easily be constructed. Further, the relative simplicity of interval methods for rigorous evaluation of continuous functions suggests the development of a homology theory based on cubical complexes. The first algorithms for the computation of cubical homology were developed in [KMŚ98,KMW99]; see [KMM04] for a self-contained exposition. More advanced algorithms have since been developed [MMP05,MPŻ08,MB09]. The computational homology package CHomP [KMP] contains implementations of the computation of the homology of simplicial and cubical complexes by Kalies, Mrozek and Pilarczyk.

The main results of this paper are that the homology of a general compact separable metric space X cannot be computable from a name of X as a compact set, and neither can the homology of a compact absolute neighbourhood retract X be computed from a name of a neighbourhood retraction $r : U \longrightarrow X$. However, the homology can be computed given both pieces of data; this is equivalent to a name of r and a single bound on X.

2 Preliminaries

In this section we review the main concepts and results from the theory of retracts, homology theory, computational cubical homology and computable analysis that we require.

2.1 Theory of retracts

Let E be a metrisable space. Recall that if $X \subset E$, then a function $r : E \longrightarrow X$ is a *retraction* if $r|_X = \mathrm{id}_X$. If U is an open neighbourhood of X in E and $r : U \longrightarrow X$ is such that $r|_X = \mathrm{id}_X$, then r is a *neighbourhood retraction*. We say X is a *neighbourhood retract* if there exists a neighbourhood retraction $r : U \longrightarrow X$. If $X \subset E$, we denote the embedding of X in E as $i : X \longrightarrow E$; note that $r|_X = r \circ i$. We say that $r : U \longrightarrow X$ is a *weak (neighbourhood) retraction* if $r|_X \sim \mathrm{id}|_X$, i.e. r is homotopic to the identity of X.

Recall that the *Hilbert cube* is the countably infinite product space $[-1,+1]^\infty$. We can give a metric by

$$d(x,y) = \left(\sum_{k=1}^{\infty} \left(\frac{x_k - y_k}{k} \right)^2 \right)^{1/2}.$$

The relative interior of the Hilbert cube is the subset $(-1,+1)^\infty$, which is not locally-compact.

A countable base for the Hilbert cube is given by open sets of the form

$$I_1 \times I_2 \times \cdots I_k \times [-1,+1] \times \cdots$$

with each I_j of the form (a_j, b_j), $(a_j, +1]$, $[-1, b_j)$ or $[-1, +1]$ for $a_j, b_j \in \mathbb{Q}$. The closures of these sets have the form

$$[a_1, b_1] \times \cdots \times [a_k, b_k] \times [-1,+1] \times \cdots$$

with $a_i, b_i \in \mathbb{Q}$ and $-1 \leq a_i < b_i \leq +1$ for $i = 1, \ldots, k$. A countable base for the relative interior of the Hilbert cube is given by the sets

$$(a_1, b_1) \times \cdots \times (a_k, b_k) \times (-1,+1) \times \cdots$$

with $a_i, b_i \in \mathbb{Q}$ and $-1 \leq a_i < b_i \leq +1$ for $i = 1, \ldots, k$.

A space X is an *absolute neighbourhood retract* if, whenever it embeds as a closed subset of a normal space Y, there is an open neighbourhood U of X in Y and a retraction $r : U \longrightarrow X$. It can be shown that a separable metric space is an absolute neighbourhood retract if, and only if, it embeds as a neighbourhood retract in the Hilbert cube. We can therefore consider absolute neighbourhood retracts as subsets of the Hilbert cube. A space is a *Euclidean neighbourhood retract* if it embeds as a neighbourhood retract in Euclidean space $\mathbb{R}^d$ for some d.

2.2 Homology Theory

Recall that a *topological pair* is a pair (X, A) where X is a topological space and A is a subset of X. By a slight abuse of notation, we will sometimes write X for the pair $(X, \emptyset)$. A *map of pairs* $f : (X, A) \longrightarrow (Y, B)$ is a continuous function $f : X \longrightarrow Y$ such that $f(A) \subset B$.

Recall that a *graded abelian group* G is a sequence $(G_q)_{q=0}^\infty$ of abelian groups. A homomorphism ϕ between graded abelian groups G and H is a sequence of group homomorphisms $\phi_q : G_q \longrightarrow H_q$. A graded abelian group G is *finite* if each G_q is finite, and $G_q = \{e\}$ for all but finitely many q.

Recall that a (finite or infinite) sequence of (graded) group homomorphisms

$$\cdots \longrightarrow G_k \xrightarrow{\phi_k} G_{k+1} \xrightarrow{\phi_{k+1}} G_{k+2} \longrightarrow \cdots$$

is *exact* if $\mathrm{im}(\phi_k) = \ker(\phi_{k+1})$ for all k.

There are a large number of homology theories, each with different properties. However, they all satisfy the following axioms.

Axioms 1 (Eilenberg-Steenrod) *A homology theory consists of a covariant functor H_* from (a full subcategory of) the category of topological pairs to the category of graded abelian groups, and a natural transformation ∂_* of degree -1 from $H_*(X,A)$ to $H_*(A)$ satisfying Axioms (i) to (iv) below.*

In other words, $H_q(X,A)$ is an abelian group for $q = 0,1,\ldots$, if $f : (X,A) \longrightarrow (Y,B)$ then $H_q(f) : H_q(X,A) \longrightarrow H_q(Y,B)$, $H_*(g \circ f) = H_*(g) \circ H_*(f)$ and $\delta_q : H_q(X,A) \longrightarrow H_{q-1}(A)$.

1. Homotopy: *If $f_0, f_1 : (X,A) \longrightarrow (Y,B)$ are homotopic, then*
$$H_*(f_0) = H_*(f_1) : H_*(X,A) \longrightarrow H_*(Y,B).$$
2. Exactness: *Each pair (X, A) induces a long exact sequence in homology, via the inclusions $i : A \longrightarrow X$ and $j : X \longrightarrow (X,A)$ by*
$$\cdots \longrightarrow H_q(A) \xrightarrow{i_*} H_q(X) \xrightarrow{j_*} H_q(X,A) \xrightarrow{\partial_*} H_{q-1}(A) \longrightarrow \cdots .$$
3. Excision: *If (X,A) is a pair and U is a subset of X such that $\bar{U} \subset A^\circ$, then the inclusion map $i : (X \setminus U, A \setminus U) \longrightarrow (X,A)$ induces an isomorphism in homology*
$$i_* : H_*(X \setminus U, A \setminus U) \approx H_*(X,A).$$
4. Dimension: *If P is a one-point space, then*
$$H_q(P) \equiv \begin{cases} 0 \text{ if } q \neq 0 \\ \mathbb{Z} \text{ if } q = 0. \end{cases}$$

It is well-known that the homology of a compact absolute neighbourhood retract is uniquely determined by the axioms. For the homology of simplicial set is determined by the axioms, and can be effectively computed from the axioms (though the computation is usually performed in practice using the simplicial homology theory). Additionally, any compact absolute neighbourhood retract is *dominated* by a finite simplicial complex, allowing computation of the homology. That any compact absolute neighbourhood retract has the homotopy type of a finite simplicial complex was a long-standing open conjecture, finally proved by West [Wes77]. We shall use a similar technique to relate the homology of a compact absolute neighbourhood retract to that of a finite cubical complex.

Recall that a single-valued function $f : X \longrightarrow Y$ is a *selection* of a multi-valued function $F : X \rightrightarrows Y$ if $f(x) \in F(x)$ for all $x \in X$. It is not difficult to show that if $F : X \rightrightarrows Y$ is convex-valued and f_0, f_1 are two continuous selections of F, then f_0 and f_1 are homotopic. We say that $F : (X,A) \rightrightarrows (Y,B)$ is a multivalued map of pairs if $F : X \rightrightarrows Y$ and $F(a) \subset B$ for all $a \in A$. Hence if $F : (X,A) \rightrightarrows (Y,B)$ is a multivalued map of pairs with convex values, then any continuous selections $f_0, f_1 : (X,A) \longrightarrow (Y,B)$ are homotopic, and so have the same homology. We can therefore speak of the homology of a multivalued map.

2.3 Computational Homology

The computational homology approach begins with the computation of the homology of *cubical sets* which are essentially finite unions of cubes in Euclidean space.

The following definition is modified from [KMM04, Definitions 2.1,3,9].

Definition 2 (Cubical Set). *An* elementary interval *is a closed interval $I \subset \mathbb{R}$ of the form $I = [k, k]$ or $I = [k, k+1]$ for some $k \in \mathbb{Z}$. An* elementary cube *$Q \subset \mathbb{R}^d$ is a finite product of elementary intervals $Q = I_1 \times I_2 \times \cdots \times I_d$. An* elementary cubical chain *is a formal sum of oriented elementary cubes. The* boundary *∂Q of an elementary cube Q is the formal sum of the elementary cubes of dimension $\dim(Q) - 1$ with the natural orientation.*

An elementary cubical complex *$\mathcal{Q}$ is a set of elementary cubes Q such that if $Q \in \mathcal{Q}$, then any elementary cube which is a subset of Q is also an element of $\mathcal{Q}$.*

A cubical complex *is a set $\mathcal{X}$ of the form $\mathcal{X} = \{s_l(Q) \mid Q \in \mathcal{Q}\}$ where $\mathcal{Q}$ is an elementary cubical complex and $s_l(x) = x/2^l$ is a scaling transformation. A cubical complex $\mathcal{X}'$ is a* refinement *of $\mathcal{X}$ if $Q = \bigcup\{Q' \in \mathcal{X}' | Q' \subset Q\}$ for all $Q \in \mathcal{X}$.*

The support *$|\mathcal{X}|$ of a cubical complex $\mathcal{X}$ is the union of all elementary cubes of X. A set X is* cubical *if there is a cubical complex $\mathcal{X}$ such that $X = |\mathcal{X}|$.*

Definition 3 (Cubical Map). *Let $\mathcal{X}$ and $\mathcal{Y}$ be cubical complexes. A cubical function is a multivalued function $\mathcal{F} : \mathcal{X} \rightrightarrows \mathcal{Y}$ such that $\mathcal{F}(Q_1 \cap Q_2) = \mathcal{F}(Q_1) \cap \mathcal{F}(Q_2)$. A cubical function is* convex *if $|\mathcal{F}(Q)|$ is convex for all $Q \in \mathcal{X}$.*

The support *$|\mathcal{F}|$ of a cubical function $\mathcal{F}$ is the lower-semicontinuous multivalued map $|\mathcal{F}| : |\mathcal{X}| \rightrightarrows |\mathcal{Y}|$ defined by $|\mathcal{F}|(x) = |\mathcal{F}(Q)|$ for $x \in \operatorname{rel\,int}(Q)$. We say that a multivalued map $F : X \longrightarrow Y$ is* cubical *if there are cubical complexes $\mathcal{X}$ and $\mathcal{Y}$ with $X = |\mathcal{X}|$, $Y = |\mathcal{Y}|$, and a cubical function $\mathcal{F} : \mathcal{X} \rightrightarrows \mathcal{Y}$ such that $F = |\mathcal{F}|$.*

The following theorem asserts that cubical homology is effectively computable.

Theorem 4.

1. *Let $(\mathcal{X}, \mathcal{A})$ be cubical complexes. Then the cubical homology $H_*(|\mathcal{X}|, |\mathcal{A}|)$ is effectively computable given $\mathcal{X}$ and $\mathcal{A}$.*
2. *Let $\mathcal{F} : (\mathcal{X}, \mathcal{A}) \longrightarrow (\mathcal{Y}, \mathcal{B})$ be a convex cubical function. Then the cubical homology $H_*(|\mathcal{F}|)$ is effectively computable given $\mathcal{F}$.*

Note that the cubical homology theory is essentially a combinatorial theory (for cubical complexes and convex cubical functions) which induces a topological theory on the supports. It is possible to show that the homology of a cubical set (or map) does not depend on the cubical complex (or function) used for the representation. However, the cubical theory is only defined on the special classes of cubical sets and cubical maps. To extend the theory to arbitrary sets and maps, we need to reduce to the cubical theory. The main results of this paper involve showing that these reductions can be performed effectively.

2.4 Computability Theory

In this section we give an overview of computability in analysis, following the type-two effectivity theory of [Wei00].

Let Σ be a finite alphabet, such as the binary digits $\{0,1\}$ or the ASCII character set. By Σ^* we mean the set of finite words on Σ, and by Σ^ω the set of infinite sequences. We say a function $\eta :\subset \Sigma^{*/\omega} \times \cdots \times \Sigma^{*/\omega} \longrightarrow \Sigma^{*/\omega}$ is *computable* if it can be evaluated by a Turing machine. The set of computable functions is closed under composition.

We will sometimes need a computable *tupling* operation $\tau\ (\Sigma^*)^\omega \longrightarrow \Sigma^\omega$, denoted $(w_1, w_2, \ldots) \mapsto \langle w_1, w_2, \ldots \rangle$.

Let M be a set. A *notation* of M is a partial surjective function $\nu :\subset \Sigma^* \longrightarrow M$. A *representation* of M is a partial surjective function $\delta :\subset \Sigma^\omega \longrightarrow M$. A δ-*name* of $x \in M$ is an element $p \in \Sigma^\omega$ such that $\delta(p) = x$.

If $\delta_0, \ldots, \delta_k$ are representations of $M_0, \ldots, M_k$ respectively, then a function $f : M_1 \times \cdots \times M_k \longrightarrow M_0$ is *computable* if there is a computable function $\eta :\subset \Sigma^\omega \times \cdots \times \Sigma^\omega \longrightarrow \Sigma^\omega$ such that $f(\delta_1(p_1), \ldots, \delta_k(p_k)) = \delta_0(\eta(p_1, \ldots, p_k))$ whenever the left-hand side is defined. If the representations of $M_0, \ldots, M_k$ being used are clear from the context, we simply say that f is (effectively) computable.

If M is a topological space, we are interested in representations which are compatible with the topological structure. A *computable topological space* is a tuple (M, τ, σ, ν) where σ is a sub-base for a T_0 topology τ on M, and ν is a notation of τ. The *standard representation* of (M, τ, σ, ν) is the representation δ of M defined by

$$\delta\langle w_1, w_2, w_3, \ldots \rangle = x \iff \{\nu(w_i) \mid i \in \mathbb{N}\} = \{I \in \sigma \mid x \in I\}.$$

In other words, p encodes a list of all sub-basic sets I containing x. By the T_0 hypothesis on (M, τ), this p encodes a unique element of M.

If M is the Hilbert cube and ν is an encoding of the standard basis set β, then the standard arithmetical operations $+$, $-$, $\times$ and $\div$ are computable with respect to the standard representation.

Given a locally-compact Hausdorff space X and a base β for X with notation ν we can construct representations for open and compact subsets of X as follows:

1. A $\theta_<$-name of an open subset U of X encodes a list of all $I \in \beta$ such that $\bar{I} \subset U$.
2. A $\kappa_>$-name of a compact subset C of X encodes a list of all tuples $(J_1, \ldots, J_k) \in \beta^*$ such that $C \subset \bigcup_{i=1}^k J_i$.
3. A κ-name of a compact subset C of X encodes a list of all tuples $(J_1, \ldots, J_k) \in \beta^*$ such that $C \subset \bigcup_{i=1}^k J_i$ and $J_i \cap C \neq \emptyset$ for all $i = 1, \ldots, k$.

These are standard representations with respect to the Scott topology on open sets, and the (upper) Vietoris topology on compact sets. We can also construct representations for continuous functions:.

4. Let U be an open subset of X, and $f : U \longrightarrow Y$ a continuous function. A γ-name of f encodes a list of all pairs $(I, J) \in \beta_X \times \beta_Y$ such that $\bar{I} \subset U$ and $f(\bar{I}) \subset J$.

Note that a γ-name of f implicitly contains a $\theta_<$-name of $\mathrm{dom}(f)$. This is a standard representation with respect to the compact-open topology on continuous functions.

3 Computability of Homology Groups

In this section we present the main results on computability and uncomputability of homology groups. We first show that homology is uncomputable with respect to certain representations of the space, and then find conditions under which homology is computable.

3.1 Uncomputability of homology

We now show that the homology cannot be computed from a κ-name of X, nor from a γ-name of a neighbourhood retract $r : U \longrightarrow X$ alone. These results are strong, in the sense that there is no space for which the homology can be computed from the given data.

Theorem 5. *Let X be a compact absolute neighbourhood retract. The homology function H_* is discontinuous at X in the Vietoris topology, and hence is uncomputable.*

Proof. It suffices to construct a sequence of compact absolute neighbourhood retracts X_n such that $X_n \to X$ in the Vietoris topology, but $H_*(X_n)$ does not converge to $H_*(X)$. Let x_i be a sequence of points such that each $x_i \notin X$ but $\lim_{n\to\infty} x_i = x_\infty \in X$. Let $X_n = X \cup \bigcup_{i=n+1}^{2n} x_i$. Then each X_n is an absolute neighbourhood retract and $X_n \to X$ in the Vietoris topology on compact sets, but $H_0(X_n) \approx H_0(X) \oplus \mathbb{Z}^n$, so the homology does not converge.

Theorem 6. *Let $X \subset \mathbb{R}^\infty$ be a compact absolute neighbourhood retract. The homology of X cannot be computed from a γ-name of a neighbourhood retraction $r : U \to \mathbb{R}^\infty$ with $r(U) = X$.*

Proof. Let $p \in \Sigma^\omega$ be a γ-name of r encoding a sequence (I_k, J_k) of basic open sets such that $r(\bar{I}_k) \subset J_k$. Let U' be an open ball $\mathrm{cl}(U') \cap \mathrm{cl}(U) = \emptyset$, let $x' \in U'$ and $r' : U' \longrightarrow \{x'\}$. Let p' be a γ-name of r' encoding a sequence (I'_k, J'_k) such that $r'(\bar{I}'_k) \subset J'_k$.

Take $\hat{U} = U \cup U'$, $\hat{X} = X \cup X'$ and define $\hat{r} : \hat{U} \longrightarrow \hat{X}$ by $\hat{r}(a) = r(a)$ if $a \in U$ and $\hat{r}(a) = r'(a)$ if $a \in U'$. Then $\hat{r}$ is a retraction from $\hat{U}$ to $\hat{X}$. We can construct names of $\hat{r}$ by taking an arbitrarily long prefix of a name of $\tilde{r}$, and then splicing in a name of r'. For $n \in \mathbb{N}$, define $(\hat{I}_{n,k}, \hat{J}_{n,k}) = (I_k, J_k)$ for $k \leq n$, and $(\hat{I}_{n+2j-1}, \hat{J}_{n+2j-1}) = (I_{n+j}, J_{n+j})$, $(\hat{I}_{n+2j}, \hat{J}_{n+2j}) = (I_{n+j}, J_{n+j})$ for $j \in \mathbb{N}$. Let $\hat{p}_n$ be the encoding of the sequence $(\hat{I}_{n,i}, \hat{J}_{n,i})$. Then for all $n \in \mathbb{N}$, $\hat{p}_n$ is an encoding of $\hat{r}$, but $\lim_{n\to\infty} \hat{p}_n = p$, which is a name of r. This means that given the name p of r, at no point can we deduce p is a name of r and not $\hat{r}$, and so at no point can we deduce $H_0(X)$.

We note that while the first result is due to an argument that the homology is discontinuous, for the second we needed to consider the details of the representation. This suggests that the homology can "almost" be computed from a name of a neighbourhood retraction. In the next section we shall see that this is indeed the case.

3.2 Homology of Euclidean neighbourhood retracts

To give an idea of the general method, we first prove effective computability of $H_*(X)$ for a Euclidean neighbourhood retract X.

Theorem 7. *Let X be a compact Euclidean neighbourhood retract. Then $H_*(X)$ can be effectively computed from a γ-name of a retraction $r : U \longrightarrow X$ with U an open subset of $\mathbb{R}^d$, and from a $\kappa_>$-name of X as a compact subset of $\mathbb{R}^d$.*

Proof (Proof (Sketch)). From the $\kappa_>$-name of X and a $\theta_<$-name of U we can effectively compute a cubical set C such that $X \subset C^\circ$ and $C \subset U$. Since $r(C) = X \subset C^\circ$, every point x of C has a basic open neighbourhood I such that $f(\bar{I}) \subset J$ with $J \subset C$. From a γ-name of r, we can therefore compute a convex-valued cubical map $R : C \rightrightarrows C$ such that $r(x) \in R(x)$ for all x. Since C is a cubical set and R is a cubical map, $H_*(C)$ and $H_*(R)$ can be computed using Theorem 4.

Let $i : X \longrightarrow C$ be the embedding of X in C, and $p : C \longrightarrow X$ be the restriction of r to C. Then $p \circ i = \mathrm{id}_X$, so $H_*(p \circ i) = \mathrm{id}_{H_*(X)}$. Hence $H_*(p)$ is surjective and $H_*(i)$ is injective. Since $i \circ p = r|_C$, the cubical map R is an over-approximation to $i \circ p$, so $H_*(i \circ p) = H_*(R)$. Then $H_*(X) = H_*(p)(H_*(C)) \approx H_*(i)(H_*(p)(H_*(C))) = H_*(i \circ p)(H_*(C)) = H_*(R)(H_*(C))$, so can be effectively computed.

The presentation of $H_*(X)$ is as a subgroup of $H_*(C)$ for which we have an explicit presentation. The subgroup is the image of $H_*(C)$ under the homomorphism $H_*(R)$. Notice that $H_*(R)$ is a projection on $H_*(C)$, since $H_*(R) = H_*(i \circ p) = H_*(i \circ (p \circ i) \circ p) = H_*(i \circ p \circ i \circ p) = H_*(i \circ p)^2 = H_*(R)^2$.

3.3 Computation of homology for compact absolute neighbourhood retracts

Lemma 8. *Let (X, A) be a pair of compact absolute neighbourhood retracts embedded in the Hilbert cube. Then given $\kappa_>$-names of X and A, and γ-names of $r_X : U_X \longrightarrow X$ and $r_A : U_A \longrightarrow A$, it is possible to effectively compute a pair $(\hat{X}, \hat{A})$ of cubical sets, and maps of pairs $i : (X, A) \longrightarrow (\hat{X}, \hat{A})$ and $p : (\hat{X}, \hat{A}) \longrightarrow (X, A)$ such that $p \circ i \sim \mathrm{id}_{X,A}$.*

Proof. By the effective Urysohn lemma [Wei01], we can construct a function $\phi : U_X \longrightarrow [0, 1]$ such that $\phi(x) = 1$ on a small neighbourhood of A, and $\phi(x) = 0$ outside U_A. We define $q : U_X \longrightarrow X$ by $q(x) = r_X(\phi(x) r_A(x) + (1 - \phi(x))x)$. It is straightforward to verify that q maps a small neighbourhood V_A of A in U_X

to A, that $q|_X$ is homotopic to the identity, and that we can compute a γ-name of q.

Since $X \subset U_X$, and using the topology of the Hilbert cube, we can effectively compute a cubical subset $\hat{X}$ of $\mathbb{R}^d$ such that $X \subset \hat{X}^\circ \times (-1,+1)^\infty$ and $\hat{X} \times [-1,+1]^\infty \subset U_X$. Further, we can ensure that $\hat{X}$ has a cubical subset $\hat{A}$ such that $A \subset \hat{A}^\circ \times (-1,+1)^\infty$ and $\hat{A} \times [-1,+1]^\infty \subset V_A$.

We take $i : (X,A) \longrightarrow (\hat{X},\hat{A})$ as $i(x) = \pi(x)$, which is clearly computable, and $p : (\hat{X},\hat{A}) \longrightarrow (X,A)$ by $p(x) = q(x,0,\ldots)$. Since q is homotopic to the identity on (X,A), we find $p \circ i \sim \mathrm{id}_{X,A}$ by the homotopy extension theorem.

Lemma 9. *Let $(\hat{X},\hat{A})$ and $(\hat{Y},\hat{B})$ be cubical sets, and $f : (\hat{X},\hat{A}) \longrightarrow (\hat{Y}^\circ,\hat{B}^\circ)$. Then given a γ-name of f, it is possible to effectively compute a convex cubical map $F : (\hat{X},\hat{A}) \longrightarrow (\hat{Y},\hat{B})$ such that f is a selector of F.*

Proof. Given a γ-name of f, we list all pairs (I,J) such that $f(\bar{I}) \subset J$, that $\bar{J} \subset \hat{Y}^\circ$ and $\bar{J} \subset \hat{B}^\circ$ if $I \cap A \neq \emptyset$. We eventually obtain an open cover of $\hat{X}$ by such sets I. By refining $\hat{X}$ if necessary, we can assume that each cell Q of $\hat{X}$ lies in some I with corresponding J. We define $\mathcal{F}(Q) = \{Q' \in \mathcal{K}(\hat{Y}) \mid J \cap Q' \neq \emptyset\}$. It is easy to verify that $|\mathcal{F}|$ is the required convex cubical map.

We can now compute the homology of an arbitrary topological pair.

Theorem 10. *Let (X,A) be a pair of compact absolute neighbourhood retracts embedded in the Hilbert cube. Then the homology $H_*(X,A)$ can be effectively computed from $\kappa_>$-names of X and A, and γ-names of r_X and r_A.*

Proof. Let $i : (X,A) \longrightarrow (\hat{X},\hat{A})$ and $p : (\hat{X},\hat{A}) \longrightarrow (X,A)$ be as given by Lemma 8, so that $p \circ i \sim \mathrm{id}_{X,A}$. Then $H_*(p \circ i) = \mathrm{id}_{H_*(X,A)}$ so $H_*(p)$ is surjective, and $H_*(i)$ is injective, and hence $H_*(X,A) \approx H_*(i \circ p)(H_*(\hat{X},\hat{A}))$. By Lemma 9 we can effectively compute a cubical map $P : (\hat{X},\hat{A}) \rightrightarrows (\hat{X},\hat{A})$ such that $i \circ p$ is a selection of P. Since $i \circ p$ is a selection of P, $H_*(i \circ p) = H_*(P)$. The result follows since we can compute the homology of $H_*(\hat{X},\hat{A})$ and $H_*(P)$ by Theorem 4.

We now consider the computation of the homology of a map of pairs.

Theorem 11. *Let (X,A) and (Y,B) be compact absolute neighbourhood retracts, equipped with the information needed to compute the homology. Let $f : (X,A) \longrightarrow (Y,B)$ be a map of pairs. Then the homology $H_*(f)$ can be computed from a γ-name of f.*

Proof. From Lemma 8, the approximate projection $p : (\hat{X},\hat{A}) \longrightarrow (X,A)$ can be effectively computed, as can the approximate embedding $i : (Y,B) \longrightarrow (\hat{Y},\hat{B})$. Then $i_{Y,B} \circ f \circ p_{X,A} : (\hat{X},\hat{A}) \longrightarrow (\hat{Y},\hat{B})$ can be effectively computed. We can therefore compute a cubical map $\hat{f} : (\hat{X},\hat{A}) \longrightarrow (\hat{Y},\hat{B})$ which is an over-approximation to $i_{Y,B} \circ f \circ p_{X,A}$. The homology of f is then given by $H_*(f) \approx H_*(p_{Y,B}) \circ H_*(\hat{f})$, since $H_*(p_{Y,B})$ is a projection of $H_*(\hat{Y},\hat{B})$ onto $H_*(Y,B)$ considered as a subgroup of $H_*(\hat{Y},\hat{B})$.

4 Conclusions

In this paper, we have considered the information required to compute the homology groups of compact absolute neighbourhood retracts. We have shown that the homology can be computed given a bound for the set, and the name of a neighbourhood retract from a subset of the Hilbert cube to the space. The derivations use standard homotopy arguments to reduce the problem to a problem of computing cubical homology.

An interesting question for further research is whether the requirements that X be a compact absolute neighbourhood retract can be weakened. If X is not compact, then the homology groups are not finite, but X still has the homotopy type of a (now infinite) simplicial complex. If X is not an absolute neighbourhood retract, then it cannot be embedded in Euclidean space or the Hilbert cube as a neighbourhood retract, and so a different representation of X is required.

An alternative approach would be to reduce the problem to the problem of computing simplicial homology. However, since existing numerical approaches work better with interval arithmetic and cubical sets, the cubical approach is closer to existing implementations.

References

[ES45] Samuel Eilenberg and Norman E. Steenrod. Axiomatic approach to homology theory. *Proc. Nat. Acad. Sci. U. S. A.*, 31:117–120, 1945.

[ES52] Samuel Eilenberg and Norman Steenrod. *Foundations of algebraic topology*. Princeton University Press, Princeton, New Jersey, 1952.

[KMM04] Tomasz Kaczynski, Konstantin Mischaikow, and Marian Mrozek. *Computational homology*, volume 157 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 2004.

[KMP] William Kalies, Marian Mrozek, and Paweł Pilarczyk. Homology software [in:] computational homology program. available at: http://chomp.rutgers.edu/.

[KMŚ98] T. Kaczyński, M. Mrozek, and M. Ślusarek. Homology computation by reduction of chain complexes. *Comput. Math. Appl.*, 35(4):59–70, 1998.

[KMW99] William D. Kalies, Konstantin Mischaikow, and Greg Watson. Cubical approximation and computation of homology. In *Conley index theory (Warsaw, 1997)*, volume 47 of *Banach Center Publ.*, pages 115–131. Polish Acad. Sci., Warsaw, 1999.

[Mas91] William S. Massey. *A basic course in algebraic topology*, volume 127 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1991.

[MB09] Marian Mrozek and Bogdan Batko. Coreduction homology algorithm. *Discrete Comput. Geom.*, 41(1):96–118, 2009.

[MMP05] Konstantin Mischaikow, Marian Mrozek, and Pawel Pilarczyk. Graph approach to the computation of the homology of continuous maps. *Found. Comput. Math.*, 5(2):199–229, 2005.

[MPŻ08] Marian Mrozek, Paweł Pilarczyk, and Natalia Żelazna. Homology algorithm based on acyclic subspace. *Comput. Math. Appl.*, 55(11):2395–2412, 2008.

[Mun84] James R. Munkres. *Elements of algebraic topology*. Addison-Wesley Publishing Company, Menlo Park, CA, 1984.

[Spa81] Edwin H. Spanier. *Algebraic topology*. Springer-Verlag, New York, 1981. Corrected reprint.

[Wei00] Klaus Weihrauch. *Computable analysis*. Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2000. An introduction.

[Wei01] Klaus Weihrauch. On computable metric spaces Tietze-Urysohn extension is computable. In *Computability and complexity in analysis (Swansea, 2000)*, volume 2064 of *Lecture Notes in Comput. Sci.*, pages 357–368. Springer, Berlin, 2001.

[Wes77] James E. West. Mapping Hilbert cube manifolds to ANR's: a solution of a conjecture of Borsuk. *Ann. Math. (2)*, 106(1):1–18, 1977.

$\mathbf{\Sigma}^0_\alpha$-Admissible Representations (Extended Abstract)

Matthew de Brecht and Akihiro Yamamoto

Graduate School of Informatics, Kyoto University
Yoshida Honmachi, Sakyo-ku, Kyoto, Japan 606-8501
matthew@iip.ist.i.kyoto-u.ac.jp
akihiro@i.kyoto-u.ac.jp

Abstract. We investigate a hierarchy of representations of topological spaces by measurable functions that extends the traditional notion of admissible representations common to computable analysis. Specific instances of these representations already occur in the literature (for example, the naive Cauchy representation of the reals and the "jump" of a representation), and have been used in investigating the computational properties of discontinuous functions. Our main contribution is the integration of a recently developing descriptive set theory for non-metrizable spaces that allows many previous results to generalize to arbitrary countably based T_0 topological spaces. In addition, for a class of topological spaces that include the reals (with the Euclidean topology) and the power set of ω (with the Scott-topology), we give a complete characterization of the functions that are (topologically) realizable with respect to the level of the representations of the domain and codomain spaces.

1 Introduction

In this paper, we introduce and investigate the topological properties of a hierarchy of representations of topological spaces, which we call $\mathbf{\Sigma}^0_\alpha$-admissible representations. A partial function $\rho :\subseteq \omega^\omega \to X$ is called a $\mathbf{\Sigma}^0_\alpha$-admissible representation ($1 \leq \alpha < \omega_1$) of the topological space X if and only if ρ is $\mathbf{\Sigma}^0_\alpha$-measurable and every $\mathbf{\Sigma}^0_\alpha$-measurable partial function to X is continuously reducible to ρ (see Definition 4). As $\mathbf{\Sigma}^0_1$-measurable functions are exactly the continuous functions, a $\mathbf{\Sigma}^0_1$-admissible representation is the same as the traditional notion of an "admissible" representation common to computable analysis (see [12] and [9]). A well known example of a $\mathbf{\Sigma}^0_2$-admissible representation is the naive Cauchy representation of the reals [4], and examples of representations in the finite levels of the hierarchy can be obtained iteratively by taking the "jump" of a representation [14]. These representations have been used in investigating the computational properties of discontinuous functions (see [3], [14], and [5]).

Whereas previous results have focused on metrizable spaces and finite levels of the hierarchy, in this paper we will investigate these representations for arbitrary countably based T_0 spaces and all countable levels of the hierarchy. Perhaps one reason that previous research has been restricted to metrizable spaces

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 119-130
http://drops.dagstuhl.de/opus/volltexte/2009/2264

is that the classical definition of the Borel hierarchy behaves rather poorly on non-metrizable spaces. Since the domain of a $\mathbf{\Sigma}^0_\alpha$-admissible representation is a metrizable space, we can use the classical definition of the Borel hierarchy in defining these representations, even for arbitrary topological spaces. However, to better understand their properties, a slight modification of the definition of the Borel hierarchy is needed for non-metrizable spaces. It turns out that the correct definition is the one that has only recently been used by Tang [11] in studying $\mathcal{P}(\omega)$ and more extensively studied by Selivanov (see [10] for a survey). Using this modification, it can be shown that the Borel complexity of a subset of a countably based T_0 space is exactly determined by the complexity of the preimage of the set under a $\mathbf{\Sigma}^0_1$-admissible representation (see Corollary 3 below). Similar properties hold for higher levels of the hierarchy, and this regularity allows us to easily characterize the types of functions that are topologically realizable with respect to these representations. In particular, we give a complete characterization for a class of topological spaces that include the reals and $\mathcal{P}(\omega)$ (see Theorem 9), and have also extended some important realizability results by Brattka [3] and Ziegler [14] to all countably based T_0-spaces (see Theorem 8).

A final result worth mentioning is that, given a representation $\rho :\subseteq \omega^\omega \to X$ of a set X, if there is a sequential topology τ on X that makes ρ a $\mathbf{\Sigma}^0_\alpha$-admissible representation, then both τ and α are uniquely determined (see Corollary 4). Thus, $\mathbf{\Sigma}^0_\alpha$-admissible representations provide a useful means of characterizing representations that cannot be interpretted as being admissible in the usual (continuous) sense.

We will define the Borel hierarchy for arbitrary topological spaces and review its basic properties in the next section. In Section 3 we will investigate some basic properties of $\mathbf{\Sigma}^0_\alpha$-measurable functions between topological spaces. We prove that $\mathbf{\Sigma}^0_\alpha$-admissible representations exist for all countable ordinals α and all countably based T_0 spaces in Section 4, and further investigate their properties in Section 5. Section 6 investigates which functions between topological spaces are realizable with respect to $\mathbf{\Sigma}^0_\alpha$-admissible representations, and we conclude in Section 7. Several proofs have been omitted due to a lack of space. They can be obtained by contacting the first author.

2 The Borel Hierarchy

In this section we define the Borel hierarchy on arbitrary topological spaces and introduce some basic properties. We will use a definition of the Borel hierarchy that differs from the classical definition (e.g., the definition in [7]) on non-metrizable spaces, but is more suitable for general topological spaces.

We let ω_1 denote the least uncountable ordinal, ω the set of natural numbers (or the first infinite ordinal), and for sets A and B we let $A \setminus B$ denote the subset of A of elements not in B.

Definition 1. *Let X be a topological space. For each ordinal α ($1 \leq \alpha < \omega_1$) we define $\mathbf{\Sigma}^0_\alpha(X)$ inductively as follows.*

1. $\mathbf{\Sigma}^0_1(X)$ *is the set of all open subsets of* X.
2. *For* $\alpha > 1$, $\mathbf{\Sigma}^0_\alpha(X)$ *is the set of all subsets* A *of* X *which can be expressed in the form*

$$A = \bigcup_{i \in \omega} B_i \setminus B'_i,$$

where for each i, B_i *and* B'_i *are in* $\mathbf{\Sigma}^0_{\beta_i}(X)$ *for some* $\beta_i < \alpha$.

We define $\mathbf{\Pi}^0_\alpha(X) = \{X \setminus A \mid A \in \mathbf{\Sigma}^0_\alpha(X)\}$, $\mathbf{\Delta}^0_\alpha(X) = \mathbf{\Sigma}^0_\alpha(X) \cap \mathbf{\Pi}^0_\alpha(X)$, *and* $\mathbf{B}(X) = \bigcup_{1 \leq \alpha < \omega_1} \mathbf{\Sigma}^0_\alpha(X)$. □

The above definition of the Borel hierarchy is equivalent to the definition that was used by Tang [11] in studying descriptive set theory on $\mathcal{P}(\omega)$ (the power set of the natural numbers with the Scott-topology), and more systematically investigated by Selivanov (see [10] for a survey of results and an extensive list of references).

The classical definition of the Borel hierarchy (which requires $B_i = X$ for all i in the second clause of Definition 1) is not suitable for non-metrizable spaces. For example, consider the Sierpinski space $\mathcal{S} = \{\bot, \top\}$ (where $\{\top\}$ is open, but $\{\bot\}$ is not). If we used the classical definition then $\mathbf{\Sigma}^0_{2n+1}(\mathcal{S})$ is the set of open subsets of $\mathcal{S}$ and $\mathbf{\Sigma}^0_{2n+2}(\mathcal{S})$ is the closed subsets, so $\mathbf{\Sigma}^0_{2n+1}(\mathcal{S}) \not\subseteq \mathbf{\Sigma}^0_{2n+2}(\mathcal{S})$ (for $0 \leq n < \omega$). The Borel hierarchy defined in Definition 1 is equivalent to the classical definition for all metrizable spaces, and behaves as we expect it should even for non-metrizable spaces.

In the following, X and Y will denote arbitrary topological spaces, unless stated otherwise. The following results are easily proven, and can also be found in [10].

Proposition 1. *For each* α *(*$1 \leq \alpha < \omega_1$*),*

1. $\mathbf{\Sigma}^0_\alpha(X)$ *is closed under countable unions and finite intersections,*
2. $\mathbf{\Pi}^0_\alpha(X)$ *is closed under countable intersections and finite unions,*
3. $\mathbf{\Delta}^0_\alpha(X)$ *is closed under finite unions, finite intersections, and complementation.*

□

Proposition 2. *If* $\beta < \alpha$ *then* $\mathbf{\Sigma}^0_\beta(X) \cup \mathbf{\Pi}^0_\beta(X) \subseteq \mathbf{\Delta}^0_\alpha(X)$. □

Proposition 3. *For* $\alpha > 2$, *each* $A \in \mathbf{\Sigma}^0_\alpha(X)$ *can be expressed in the form* $A = \bigcup_{i \in \omega} B_i$, *where for each* i, B_i *is in* $\mathbf{\Pi}^0_{\beta_i}(X)$ *for some* $\beta_i < \alpha$. □

Proposition 4. *If* X *is a metrizable space, then every* $A \in \mathbf{\Sigma}^0_2(X)$ *is equal to a countable union of closed sets.* □

Proposition 5. *If* X *is a subspace of* Y, *then* $\mathbf{\Sigma}^0_\alpha(X) = \{A \cap X \mid A \in \mathbf{\Sigma}^0_\alpha(Y)\}$ *and* $\mathbf{\Pi}^0_\alpha(X) = \{A \cap X \mid A \in \mathbf{\Pi}^0_\alpha(Y)\}$. □

A topological space X is called a *T_D-space* if every singleton set $\{x\} \subseteq X$ is locally closed, i.e. $\{x\}$ is equal to the intersection of an open set and a closed set. T_D is a separation axiom proposed by Aull and Thron [2] that is strictly between the T_0 and T_1 axioms.

Proposition 6. *For any first-countable topological space X,*

1. *Every singleton set $\{x\} \subseteq X$ is in $\mathbf{\Pi}^0_2(X) \iff X$ is a T_0-space,*
2. *Every singleton set $\{x\} \subseteq X$ is in $\mathbf{\Delta}^0_2(X) \iff X$ is a T_D-space,*
3. *Every singleton set $\{x\} \subseteq X$ is in $\mathbf{\Pi}^0_1(X) \iff X$ is a T_1-space,*
4. *Every singleton set $\{x\} \subseteq X$ is in $\mathbf{\Delta}^0_1(X) \iff X$ is a discrete space.*

□

3 $\mathbf{\Sigma}^0_\alpha$-measurable functions

In this section we will investigate some basic properties of $\mathbf{\Sigma}^0_\alpha$-measurable functions. Below, we will write $f :\subseteq X \to Y$ to indicate that f is a partial function from X to Y. The domain of definition of f will be denoted $dom(f)$. We say that $f :\subseteq X \to Y$ is *continuous* if and only if for every open $U \subseteq Y$, there is open $V \subseteq X$ such that $f^{-1}(U) = V \cap dom(f)$. In other words, $f :\subseteq X \to Y$ is continuous if and only if the total function $f : dom(f) \to Y$ is continuous with respect to the subspace topology on $dom(f)$.

Definition 2. *A function $f : X \to Y$ is* $\mathbf{\Sigma}^0_\alpha$-measurable *if and only if for every open $U \subseteq Y$, $f^{-1}(U) \in \mathbf{\Sigma}^0_\alpha(X)$. A partial function $f :\subseteq X \to Y$ is said to be* $\mathbf{\Sigma}^0_\alpha$-measurable *if and only if for every open $U \subseteq Y$, there is $A \in \mathbf{\Sigma}^0_\alpha(X)$ such that $f^{-1}(U) = A \cap dom(f)$.* □

Equivalently, a partial function $f :\subseteq X \to Y$ is $\mathbf{\Sigma}^0_\alpha$-measurable if and only if for every open $U \subseteq Y$, $f^{-1}(U) \in \mathbf{\Sigma}^0_\alpha(dom(f))$, where $dom(f)$ is given the relative topology.

For any fixed $\alpha > 1$, the $\mathbf{\Sigma}^0_\alpha$-measurable functions are not closed under composition. To characterize how composition behaves, we will need ordinal addition. Addition on ordinals is defined recursively as follows:

1. $\alpha + 0 = \alpha$
2. $\alpha + (\beta + 1) = (\alpha + \beta) + 1 =$ the successor of $\alpha + \beta$.
3. $\alpha + \lambda = \lim_{\beta < \lambda}(\alpha + \beta)$ for limit ordinal λ.

Note that ordinal addition is non-commutative. For example, $1 + \omega = \omega \neq \omega + 1$. Also note that if $\alpha < \beta$, then there is a unique ordinal γ such that $\alpha + \gamma = \beta$.

Composing with continuous functions does not change the level of a function. For that reason it would have been more convenient for our purposes to define the Borel Hierarchy so that open sets and continuous functions were of level 0 (the additive identity for ordinals). To simplify the statement of some of the following theorems and proofs, we will often make use of the following "hat" notation, so that we can treat the Borel Hierarchy as if we defined the open sets to be at level 0.

Definition 3. *For $0 \leq \alpha < \omega_1$, define $\widehat{\alpha} = \alpha + 1$ if $\alpha < \omega$ and $\widehat{\alpha} = \alpha$ if $\alpha \geq \omega$.* □

Note that $\alpha < \beta \iff \widehat{\alpha} < \widehat{\beta}$ and $\widehat{\alpha+\beta} = \widehat{\alpha} + \beta$ hold for any countable ordinals α and β.

Lemma 1. *Let X and Y be countably based T_0 spaces. If $f :\subseteq X \to Y$ is $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable ($0 \leq \alpha < \omega_1$) and $A \in \mathbf{\Sigma}^0_{\widehat{\beta}}(Y)$ ($0 \leq \beta < \omega_1$), then $f^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(dom(f))$.* □

Theorem 1. *Let X, Y, and Z be countably based T_0 spaces, $f :\subseteq X \to Y$ a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable function ($0 \leq \alpha < \omega_1$), and $g :\subseteq Y \to Z$ a $\mathbf{\Sigma}^0_{\widehat{\beta}}$-measurable function ($0 \leq \beta < \omega_1$). Then $g \circ f :\subseteq X \to Z$ is $\mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}$-measurable.* □

In particular, if f is $\mathbf{\Sigma}^0_2$-measurable and g is $\mathbf{\Sigma}^0_\omega$-measurable, then due to the non-commutativity of ordinal addition, $g \circ f$ is $\mathbf{\Sigma}^0_\omega$-measurable but $f \circ g$ is $\mathbf{\Sigma}^0_{\omega+1}$-measurable (assuming the compositions make sense).

The following is due to Wadge (this is Theorem 22.10 in [7]). We let ω^ω denote the Baire space.

Proposition 7 (Wadge). *If $B \subseteq \omega^\omega$ is in $\mathbf{B}(\omega^\omega) \setminus \mathbf{\Pi}^0_{\widehat{\alpha}}(\omega^\omega)$ ($0 \leq \alpha < \omega_1$), then for any $A \in \mathbf{\Sigma}^0_{\widehat{\alpha}}(\omega^\omega)$ there is continuous total $f : \omega^\omega \to \omega^\omega$ such that $A = f^{-1}(B)$.* □

We will need the following generalization of Wadge's results that characterize reductions using measurable functions.

Theorem 2. *For $0 \leq \alpha < \omega_1$ and $0 \leq \beta < \omega_1$, if $B \in \mathbf{B}(\omega^\omega) \setminus \mathbf{\Pi}^0_{\widehat{\beta}}(\omega^\omega)$, then for any $A \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(\omega^\omega)$ there exists a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable total function $f : \omega^\omega \to \omega^\omega$ such that $A = f^{-1}(B)$.* □

4 Existence of $\mathbf{\Sigma}^0_\alpha$-admissible representations

The goal of this section is to show that every countably based T_0 space has a $\mathbf{\Sigma}^0_\alpha$-admissible representation for $1 \leq \alpha < \omega_1$ (Theorem 3 below). We also show the complexity of converting between representations of different levels (Theorem 4), and consider representations of representations of a space (Corollary 2), which is a generalization of Ziegler's "jump" of a representation [14].

Definition 4. *A* $\mathbf{\Sigma}^0_\alpha$-admissible representation *of a topological space X is a $\mathbf{\Sigma}^0_\alpha$-measurable partial function $\rho :\subseteq \omega^\omega \to X$ such that for every $\mathbf{\Sigma}^0_\alpha$-measurable partial function $f :\subseteq \omega^\omega \to X$, there exists continuous $g :\subseteq \omega^\omega \to \omega^\omega$ such that $f = \rho \circ g$.* □

Note that the above definition implies that $\mathbf{\Sigma}^0_\alpha$-admissible representations are always surjective. Clearly, a $\mathbf{\Sigma}^0_1$-admissible representation is equivalent to what is usually called an "admissible representation" in the computable analysis literature (see, e.g., [12] and [9]). The above definition applies to arbitrary topological spaces, but most of our results will focus on countably based spaces.

We let $\mathcal{S}$ denote the Sierpinski space, which has only two points $\top$ and $\bot$, and where $\{\top\}$ is open but $\{\bot\}$ is not open.

Proposition 8. *Let $A \in \mathbf{\Sigma}^0_\alpha(\omega^\omega) \setminus \mathbf{\Pi}^0_\alpha(\omega^\omega)$ and define $\rho :\subseteq \omega^\omega \to \mathcal{S}$ so that $\rho(y) = \top$ if $y \in A$ and $\rho(y) = \bot$ if $y \notin A$. Then ρ is a $\mathbf{\Sigma}^0_\alpha$-admissible representation for $\mathcal{S}$.*

Proof. It is clear that ρ is $\mathbf{\Sigma}^0_\alpha$-measurable. Let $f :\subseteq \omega^\omega \to \mathcal{S}$ be a $\mathbf{\Sigma}^0_\alpha$-measurable partial function. Then $f^{-1}(\{\top\}) \in \mathbf{\Sigma}^0_\alpha(dom(f))$, so there is $B \in \mathbf{\Sigma}^0_\alpha(\omega^\omega)$ such that $f^{-1}(\{\top\}) = B \cap dom(f)$. From Proposition 7 there is continuous $g :\subseteq \omega^\omega \to \omega^\omega$ such that $g^{-1}(A) = B$. Then for all $y \in dom(f)$, $f(y) = \top \iff g(y) \in A \iff \rho(g(y)) = \top$. Hence, by restricting the domain of g if necessary, $f = \rho \circ g$. □

Corollary 1. *For $0 \leq \alpha < \omega_1$ and $0 \leq \beta < \omega_1$, if $\rho_{\alpha+\beta} :\subseteq \omega^\omega \to \mathcal{S}$ is a $\mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}$-admissible representation of $\mathcal{S}$ and $\rho_\beta :\subseteq \omega^\omega \to \mathcal{S}$ is a $\mathbf{\Sigma}^0_{\widehat{\beta}}$-admissible representation of $\mathcal{S}$, then there exists a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable function $f :\subseteq \omega^\omega \to \omega^\omega$ such that $\rho_{\alpha+\beta} = \rho_\beta \circ f$.*

Proof. Immediate from Theorem 2 and Proposition 8. □

Proposition 9. *If X is a subspace of Y and $\rho :\subseteq \omega^\omega \to Y$ is a $\mathbf{\Sigma}^0_\alpha$-admissible representation of Y, then $\rho_X :\subseteq \omega^\omega \to X$ defined as the restriction of ρ to $dom(\rho_X) = \rho^{-1}(X)$, is a $\mathbf{\Sigma}^0_\alpha$-admissible representation of X.* □

Proposition 10. *If $\{X_i\}_{i\in\omega}$ and $\{Y_i\}_{i\in\omega}$ are all countably based T_0-spaces, and for each i $f_i :\subseteq X_i \to Y_i$ is $\mathbf{\Sigma}^0_\alpha$-measurable ($1 \leq \alpha < \omega_1$), then $f^\omega :\subseteq \prod X_i \to \prod Y_i$ is $\mathbf{\Sigma}^0_\alpha$-measurable, where $\prod X_i$ and $\prod Y_i$ are given the product topologies and f^ω is defined so that $f^\omega(\xi)(i) = f_i(\xi(i))$.* □

For the following proposition, let $\phi :\omega^\omega \to (\omega^\omega)^\omega$ be a homeomorphism.

Proposition 11. *Let X_i be a countably based T_0 space and $\rho_i :\subseteq \omega^\omega \to X_i$ a $\mathbf{\Sigma}^0_\alpha$-admissible representation for X_i ($i \in \omega$). Then $\rho^\omega \circ \phi$ is a $\mathbf{\Sigma}^0_\alpha$-admissible representation for $\prod X_i$.*

Proof. The proof that $\rho^\omega \circ \phi$ is $\mathbf{\Sigma}^0_\alpha$-measurable follows from Proposition 10.

Let $f :\subseteq \omega^\omega \to \prod X_i$ be a $\mathbf{\Sigma}^0_\alpha$-measurable partial function. By the $\mathbf{\Sigma}^0_\alpha$-admissibility of $\rho_i :\subseteq \omega^\omega \to X_i$, for $i \in \omega$ there is continuous $g_i :\subseteq \omega^\omega \to \omega^\omega$ such that $\pi_i \circ f = \rho_i \circ g_i$, where $\pi_i : \prod X_i \to X$ is the i-th projection. Since π_i is a total function, we must have that $dom(f) = dom(\pi_i \circ f) \subseteq dom(g_i)$ for all $i \in \omega$. Define $g :\subseteq \omega^\omega \to (\omega^\omega)^\omega$ so that $g(\xi)(i) = g_i(\xi)$. Then $dom(f) \subseteq dom(g)$ and

$$\rho^\omega(g(\xi))(i) = \rho_i(g(\xi)(i)) = \rho_i(g_i(\xi)) = \pi_i(f(\xi)) = f(\xi)(i),$$

so $f = \rho^\omega \circ g$. Define $h :\subseteq \omega^\omega \to \omega^\omega$ as $h = \phi^{-1} \circ g$. Clearly, h is continuous and $f = \rho^\omega \circ g = \rho^\omega \circ \phi \circ h$. □

Since every countably based T_0 space is homeomorphic to a subspace of $\mathcal{S}^\omega$, we obtain the following.

Theorem 3. *For every countably based T_0 space X and every α ($1 \leq \alpha < \omega_1$), there exists a $\mathbf{\Sigma}^0_\alpha$-admissible representation of X.* □

The following can be proved for $X = \mathcal{S}^\omega$ by using representations obtained from Proposition 11 and applying Corollary 1 in parallel. Subspaces of $\mathcal{S}^\omega$ are handled by restricting the functions as necessary.

Theorem 4 (Reductions between representations). *Let X be a countably based T_0-space. For $0 \leq \alpha < \omega_1$ and $0 \leq \beta < \omega_1$, if $\rho_{\alpha+\beta} :\subseteq \omega^\omega \to X$ is a $\mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}$-admissible representation of X and $\rho_\beta :\subseteq \omega^\omega \to X$ is a $\mathbf{\Sigma}^0_{\widehat{\beta}}$-admissible representation of X, then there exists a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable function $f :\subseteq \omega^\omega \to \omega^\omega$ such that $\rho_{\alpha+\beta} = \rho_\beta \circ f$.* □

Corollary 2 (Representations of representations). *Let X be a countably based T_0 space, $\rho_\beta :\subseteq \omega^\omega \to X$ a $\mathbf{\Sigma}^0_{\widehat{\beta}}$-admissible representation of X, and $\rho_\alpha :\subseteq \omega^\omega \to dom(\rho_\beta)$ a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-admissible representation of $dom(\rho_\beta)$, ($0 \leq \alpha < \omega_1$, $0 \leq \beta < \omega_1$). Then $\rho_\beta \circ \rho_\alpha :\subseteq \omega^\omega \to X$ is a $\mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}$-admissible representation of X.*

Proof. First note that $\rho_\beta \circ \rho_\alpha$ is $\mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}$-measurable by Theorem 1. Let $\rho :\subseteq \omega^\omega \to X$ be a $\mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}$-admissible representation of X. By Theorem 4, there is a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable $f :\subseteq \omega^\omega \to \omega^\omega$ such that $\rho = \rho_\beta \circ f$. We can assume without loss of generality that $range(f) \subseteq dom(\rho_\beta)$, and so by the $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-admissibility of ρ_α there is a continuous $g :\subseteq \omega^\omega \to \omega^\omega$ such that $f = \rho_\alpha \circ g$. It follows that g is a continuous reduction of ρ to $\rho_\beta \circ \rho_\alpha$, thus $\rho_\beta \circ \rho_\alpha$ is $\mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}$-admissible. □

Let $\iota' :\subseteq \omega^\omega \to \omega^\omega$ be a $\mathbf{\Sigma}^0_2$-admissible representation of ω^ω. By the above theorem, if $\rho :\subseteq \omega^\omega \to X$ is a $\mathbf{\Sigma}^0_\beta$-admissible representation ($1 \leq \beta < \omega$) of a countably based T_0 space X, then $\rho \circ \iota'$ is a $\mathbf{\Sigma}^0_{\beta+1}$-admissible representation of X. This corresponds to Ziegler's "jump" of a representation [14]. However, it should be noted that if ρ is $\mathbf{\Sigma}^0_\beta$-admissible for $\beta \geq \omega$, then $\rho \circ \iota'$ is still $\mathbf{\Sigma}^0_\beta$-measurable and thus *not* $\mathbf{\Sigma}^0_{\beta+1}$-admissible.

5 Properties of $\mathbf{\Sigma}^0_\alpha$-admissible representations

The main purpose of this section is to relate the Borel complexity of a subset of a space with the complexity of the preimage of the subset under a $\mathbf{\Sigma}^0_\alpha$-admissible representation. These results will be useful in the following section where we characterize the functions that are realizable with respect to these representations.

Many of the following results are heavily dependent on the following proposition by J. Saint Raymond (Lemma 17 in [8]). Although the original statement of the result was in terms of metrizable spaces, it is easy to verify that the arguments in the proof hold for more general spaces when we define the Borel hierarchy according to Definition 1.

Proposition 12 (Saint-Raymond [8]). *Let $\phi : X \to Y$ be an open continuous surjective total function with Polish fibers (i.e. $\phi^{-1}(y)$ is Polish for each $y \in Y$), where X is a separable metric space and Y is a countably based T_0 topological*

space. Then for every $A \subseteq Y$ *and* $1 \leq \alpha < \omega_1$, $A \in \mathbf{\Sigma}^0_\alpha(Y)$ *if and only if* $\phi^{-1}(A) \in \mathbf{\Sigma}^0_\alpha(X)$. □

Since every countably based T_0 space has a $\mathbf{\Sigma}^0_1$-admissible representation that is open and has Polish fibers (see Corollary 15 and Proposition 16 in [4]), we find that the Borel hierarchy is preserved under $\mathbf{\Sigma}^0_1$-admissible representations of countably based T_0 spaces.

Corollary 3. *Let* X *be a countably based* T_0 *space and* $\rho :\subseteq \omega^\omega \to X$ *a* $\mathbf{\Sigma}^0_1$*-admissible representation of* X. *Then for* $1 \leq \alpha < \omega_1$, $A \in \mathbf{\Sigma}^0_\alpha(X)$ *if and only if* $\rho^{-1}(A) \in \mathbf{\Sigma}^0_\alpha(dom(\rho))$. □

Our next goal is to generalize Corollary 3 to some $\mathbf{\Sigma}^0_\alpha$-admissible representations. Let ω^* have as a base set $\omega \cup \{\infty\}$ and the topology so that U is open if and only if either $\infty \notin U$ or else U is cofinite (i.e., for some $m < \omega$, $n \in U$ for all $n \geq m$). Note that ω^* is the one-point compactification of ω with the discrete topology, hence the notation (which should not be confused with the set of finite strings of natural numbers).

Lemma 2. *Let* $\rho :\subseteq \omega^\omega \to \omega^*$ *be* $\mathbf{\Sigma}^0_\alpha$*-admissible* ($1 \leq \alpha < \omega_1$). *Then* $S \subseteq \omega^*$ *is open if and only if* $\rho^{-1}(S) \in \mathbf{\Sigma}^0_\alpha(dom(\rho))$. □

Definition 5. *Let* X *be an arbitrary topological space. A subset* $A \subseteq X$ *is* sequentially open *if and only if for every sequence* $\{x_i\}_{i\in\omega}$ *that converges to* $x \in A$, *there is some* m *such that* $x_n \in A$ *for all* $n \geq m$. X *is a* sequential space *if and only if all sequentially open subsets of* X *are open.* □

Note that all countably based spaces are sequential spaces (see Theorem 1.6.14 in [6]).

Theorem 5. *Let* X *be a sequential* T_0 *space and* $\rho :\subseteq \omega^\omega \to X$ *be* $\mathbf{\Sigma}^0_\alpha$*-admissible* ($1 \leq \alpha < \omega_1$). *Then* $U \subseteq X$ *is open if and only if* $\rho^{-1}(U) \in \mathbf{\Sigma}^0_\alpha(dom(\rho))$.

Proof. If U is open then $\rho^{-1}(U) \in \mathbf{\Sigma}^0_\alpha(dom(\rho))$ holds because ρ is $\mathbf{\Sigma}^0_\alpha$-measurable.

Assume that $\rho^{-1}(U) \in \mathbf{\Sigma}^0_\alpha(dom(\rho))$ and let $\{x_i\}_{i\in\omega}$ be a sequence converging to $x \in U$. Define $f :\omega^* \to X$ so that $f(n) = x_n$ and $f(\infty) = x$. Then f is clearly continuous. If δ is a $\mathbf{\Sigma}^0_\alpha$-admissible representation of ω^*, then $f \circ \delta$ is $\mathbf{\Sigma}^0_\alpha$-measurable, so by the $\mathbf{\Sigma}^0_\alpha$-admissibility of ρ there is continuous $g :\subseteq \omega^\omega \to \omega^\omega$ such that $f \circ \delta = \rho \circ g$. Since g is continuous, $\delta^{-1}(f^{-1}(U)) = g^{-1}(\rho^{-1}(U)) \in \mathbf{\Sigma}^0_\alpha(dom(\delta))$. It follows that $f^{-1}(U)$ is open by Lemma 2. Since $\infty \in f^{-1}(U)$, there is $m < \omega$ such that $n \in f^{-1}(U)$ for all $n \geq m$. Therefore, $x_n \in U$ for all $n \geq m$. Since $\{x_i\}_{i\in\omega}$ and its limit $x \in U$ were arbitrary, U is sequentially open, hence open because X is a sequential space. □

The rest of this section extends Theorem 5 to the entire hierarchy for a special class of topological spaces.

Lemma 3. *Let* $\rho :\subseteq \omega^\omega \to \omega^\omega$ *be a* $\mathbf{\Sigma}^0_{\widehat{\alpha}}$*-admissible representation of* ω^ω ($0 \leq \alpha < \omega_1$). *For* $0 \leq \beta < \omega_1$ *and* $A \subseteq \omega^\omega$, $A \in \mathbf{\Sigma}^0_{\widehat{\beta}}(\omega^\omega)$ *if and only if* $\rho^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(dom(\rho))$. □

Lemma 4. *Let X be a zero-dimensional Polish space and $\rho :\subseteq \omega^\omega \to X$ a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-admissible representation of X $(0 \leq \alpha < \omega_1)$. For $0 \leq \beta < \omega_1$, $A \in \mathbf{\Sigma}^0_{\widehat{\beta}}(X)$ if and only if $\rho^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(dom(\rho))$.*

Proof. For the non-trivial part of the lemma, we can assume that X is a closed subset of ω^ω (see Theorem 7.8 in [7]) and $\rho :\subseteq \omega^\omega \to X$ is the restriction of a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-admissible representation $\rho' :\subseteq \omega^\omega \to \omega^\omega$ of ω^ω as in Proposition 9 (i.e., $dom(\rho) = (\rho')^{-1}(X)$, and $\rho = \rho'|_{dom(\rho)}$). It follows from these assumptions that $dom(\rho) \in \mathbf{\Pi}^0_{\widehat{\alpha}}(\omega^\omega)$ because X is a closed subset of ω^ω and ρ' is $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable.

The case $\beta = 0$ is the statement of Theorem 5, so assume $\beta \geq 1$ and $A \subseteq X$ is such that $\rho^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(dom(\rho))$. By Proposition 5 there is $B \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(\omega^\omega)$ such that $\rho^{-1}(A) = B \cap dom(\rho)$. Since $\alpha < \alpha + \beta$ and $dom(\rho) \in \mathbf{\Pi}^0_{\widehat{\alpha}}(\omega^\omega)$, $\rho^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(\omega^\omega)$. Since $(\rho')^{-1}(A) = \rho^{-1}(A)$, it follows from Lemma 3 that $A \in \mathbf{\Sigma}^0_{\widehat{\beta}}(\omega^\omega)$ and hence $A \in \mathbf{\Sigma}^0_{\widehat{\beta}}(X)$. □

Definition 6. *We will say that a space X has a* Polish representation *if and only if there is a $\mathbf{\Sigma}^0_1$-admissible representation $\rho :\subseteq \omega^\omega \to X$ of X such that $dom(\rho)$ with the subspace topology is a (zero-dimensional) Polish space.*

In particular, the real numbers with the Euclidean topology and $\mathcal{P}(\omega)$ with the Scott-topology have Polish representations (an admissible representation of the reals with closed domain of definition is given in [13], and the representation $\delta :\omega^\omega \to \mathcal{P}(\omega)$ defined as $\delta(\xi) = \{n - 1 \mid \exists j(\xi(j) = n \neq 0)\}$ is total and can be shown to be admissible).

Theorem 6. *Let X be a countably based T_0 space with a Polish representation and $\rho :\subseteq \omega^\omega \to X$ a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-admissible representation of X $(0 \leq \alpha < \omega_1)$. For $0 \leq \beta < \omega_1$, $A \in \mathbf{\Sigma}^0_{\widehat{\beta}}(X)$ if and only if $\rho^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(dom(\rho))$.*

Proof. For the non-trivial part of the proof, let $\delta :\subseteq \omega^\omega \to X$ be $\mathbf{\Sigma}^0_1$-admissible such that $dom(\delta)$ is Polish. Let $\delta' :\subseteq \omega^\omega \to dom(\delta)$ be a $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-admissible representation of $dom(\delta)$. Since $\delta \circ \delta'$ is $\mathbf{\Sigma}^0_{\widehat{\alpha}}$-measurable, there is continuous $f :\subseteq \omega^\omega \to \omega^\omega$ such that $\delta \circ \delta' = \rho \circ f$.

Assume $A \subseteq X$ is such that $\rho^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(dom(\rho))$. Then $(\delta')^{-1}(\delta^{-1}(A)) = f^{-1}(\rho^{-1}(A)) \in \mathbf{\Sigma}^0_{\widehat{\alpha+\beta}}(dom(\delta'))$ because f is continuous (here we are using the fact that $dom(\delta') \subseteq dom(f)$). It follows from Lemma 4 that $\delta^{-1}(A) \in \mathbf{\Sigma}^0_{\widehat{\beta}}(dom(\delta))$, hence $A \in \mathbf{\Sigma}^0_{\widehat{\beta}}(X)$ from Corollary 3. □

6 Realizability Theorems

In this section we will investigate which functions are realizable with respect to $\mathbf{\Sigma}^0_\alpha$-admissible representations. We only consider topological realizability, and do not consider computational issues.

Definition 7. *Let X and Y be arbitrary topological spaces, and $f\colon X \to Y$ a function. We say that f is $\langle \mathbf{\Sigma}^0_\alpha, \mathbf{\Sigma}^0_\beta \rangle$-realizable by a $\mathbf{\Sigma}^0_\gamma$-measurable function if there is a $\mathbf{\Sigma}^0_\alpha$-admissible representation ρ_X of X and a $\mathbf{\Sigma}^0_\beta$-admissible representation ρ_Y of Y and a $\mathbf{\Sigma}^0_\gamma$-measurable partial function $g\colon\subseteq \omega^\omega \to \omega^\omega$ such that $f \circ \rho_X = \rho_Y \circ g$. If a continuous such g exists, then we say that f is $\langle \mathbf{\Sigma}^0_\alpha, \mathbf{\Sigma}^0_\beta \rangle$-continuously realizable.* □

Lemma 5. *Let X be an arbitrary topological space, and $\rho\colon\subseteq \omega^\omega \to X$ be a $\mathbf{\Sigma}^0_\alpha$-admissible representation of X $(1 \leq \alpha < \omega_1)$. Then X is a T_0-space.*

Proof. Exactly like Schröder's proof for $\mathbf{\Sigma}^0_1$-admissible representations (Theorem 13 in [9]). □

Lemma 6. *For $1 \leq \beta < \alpha < \omega_1$, a function from the discrete two point space $\mathbf{2}$ to the Sierpinski space $\mathcal{S}$ is $\langle \mathbf{\Sigma}^0_\alpha, \mathbf{\Sigma}^0_\beta \rangle$-continuously realizable if and only if it is a constant function.* □

Note that the following theorem does not assume that X and Y are countably based.

Theorem 7. *Let X and Y be any topological spaces such that X has a $\mathbf{\Sigma}^0_\alpha$-admissible representation and Y has a $\mathbf{\Sigma}^0_\beta$-admissible representation, where $1 \leq \beta < \alpha < \omega_1$. Then a function from X to Y is $\langle \mathbf{\Sigma}^0_\alpha, \mathbf{\Sigma}^0_\beta \rangle$-continuously realizable if and only if it is a constant function.* □

Statement (3) in the following is a topological generalization of Brattka's extention [3] of the Kreitz-Weihrauch Representation Theorem [12] to all countably based T_0-spaces and all countable ordinals. Statements (1) and (2) are generalizations of some results by Ziegler [14].

Theorem 8. *Let X and Y be countably based T_0 spaces, $f\colon X \to Y$ a total function, and $1 \leq \alpha < \omega_1$.*

1. *f is $\langle \mathbf{\Sigma}^0_1, \mathbf{\Sigma}^0_\alpha \rangle$-continuously realizable if and only if f is $\mathbf{\Sigma}^0_\alpha$-measurable,*
2. *f is $\langle \mathbf{\Sigma}^0_\alpha, \mathbf{\Sigma}^0_\alpha \rangle$-continuously realizable if and only if f is continuous,*
3. *f is $\langle \mathbf{\Sigma}^0_1, \mathbf{\Sigma}^0_1 \rangle$-realizable by a $\mathbf{\Sigma}^0_\alpha$-measurable function if and only if f is $\mathbf{\Sigma}^0_\alpha$-measurable.*

Proof. The "if" part of (1) and (2) immediately follow from the definition of admissibility. For (3), assume f is $\mathbf{\Sigma}^0_\alpha$-measurable. From statement (1) it follows that f is $\langle \mathbf{\Sigma}^0_1, \mathbf{\Sigma}^0_\alpha \rangle$-continuously realizable, and by Theorem 4 there is a $\mathbf{\Sigma}^0_\alpha$-measurable reduction of any $\mathbf{\Sigma}^0_\alpha$ representation of Y to a $\mathbf{\Sigma}^0_1$-admissible representation of Y. Composing the two produces a $\mathbf{\Sigma}^0_\alpha$-measurable function that $\langle \mathbf{\Sigma}^0_1, \mathbf{\Sigma}^0_1 \rangle$-realizes f.

The proof of the "only if" parts are similar for all three statements, so we only prove (1). Let ρ_X be a $\mathbf{\Sigma}^0_1$-admissible representation of X, ρ_Y a $\mathbf{\Sigma}^0_\alpha$-admissible representation of Y, and assume $g\colon\subseteq \omega^\omega \to \omega^\omega$ is continuous such that $f \circ \rho_X = \rho_Y \circ g$. Let $U \subseteq Y$ be open. Then $\rho_X^{-1}(f^{-1}(U)) = g^{-1}(\rho_Y^{-1}(U)) \in \mathbf{\Sigma}^0_\alpha(dom(\rho_X))$ because ρ_Y is $\mathbf{\Sigma}^0_\alpha$-measurable, g is continuous, and $dom(\rho_X) \subseteq dom(g)$. By Corollary 3, it follows that $f^{-1}(U) \in \mathbf{\Sigma}^0_\alpha(X)$, hence f is $\mathbf{\Sigma}^0_\alpha$-measurable (for statement (2), use Theorem 5 instead of Corollary 3). □

The following shows that, assuming that a representation of a set is admissible at some level with respect to some topology on the set, then the level of the representation and any corresponding sequential topology on the set is uniquely determined. Note, however, that it is easy to construct representations of a set that are not admissible at any level with respect to any topology on the set.

Corollary 4. *Let X be a set with at least two elements, and let $\rho :\subseteq \omega^\omega \to X$ be an arbitrary function. If τ and τ' are two topologies on X such that ρ is $\mathbf{\Sigma}^0_\alpha$-admissible ($1 \leq \alpha < \omega_1$) with respect to τ, and ρ is $\mathbf{\Sigma}^0_\beta$-admissible ($1 \leq \beta < \omega_1$) with respect to τ', then $\alpha = \beta$. If in addition τ and τ' are sequential topologies then $\tau = \tau'$.* □

Finally, we give a complete characterization for the case that X has a Polish representation (recall that ordinal addition is non-commutative). Note that a generalization of Theorem 6 to all countably based T_0-spaces would allow us to drop the "Polish representation" restriction on X.

Theorem 9. *Let X and Y be countably based T_0 spaces, and further assume X has a Polish representation. For any total function $f : X \to Y$ and any countable ordinals α, β and γ, there exists a $\mathbf{\Sigma}^0_{\widehat{\gamma}}$-measurable $g :\subseteq \omega^\omega \to \omega^\omega$ that $\langle \mathbf{\Sigma}^0_{\widehat{\alpha}}, \mathbf{\Sigma}^0_{\widehat{\beta}} \rangle$-realizes f if and only if:*

1. *$\alpha > \gamma + \beta$ and f is a constant function, or*
2. *$\alpha \leq \gamma + \beta$ and f is a $\mathbf{\Sigma}^0_{\widehat{\eta}}$-measurable function, where η is (the unique ordinal) such that $\alpha + \eta = \gamma + \beta$.*

□

7 Conclusion

We have introduced and investigated the basic properties of a hierarchy of representations of topological spaces. $\mathbf{\Sigma}^0_\alpha$-admissible representations provide a well-behaved topological interpretation of representations that can not be interpreted as admissible in the traditional (continuous) sense (see Corollary 4). These representations are also significant for better understanding the computational properties of discontinuous functions, which has been investigated for metric spaces in [3], [14], and [5].

The first open problem is to generalize Theorem 6 to all countably based T_0-spaces. One difficulty in generalizing Saint-Raymond's result (Proposition 12) is that the fibers of $\mathbf{\Sigma}^0_\alpha$-admissible representations are not Polish in general.

A second open problem is to classify precisely which topological spaces have $\mathbf{\Sigma}^0_\alpha$-admissible representations. An attractive conjecture is that they are exactly the spaces with $\mathbf{\Sigma}^0_1$-admissible representations, which were completely classified by Schröder [9].

Finally, a further refinement of the hierarchy would be useful, particularly between the continuous and $\mathbf{\Sigma}^0_2$ representations. One interesting class of functions are the $\mathbf{\Delta}^0_2$-functions (i.e., preimages of open sets are $\mathbf{\Delta}^0_2$, or, equivalently, preimages of $\mathbf{\Sigma}^0_2$ sets are $\mathbf{\Sigma}^0_2$), which are closed under composition. Wadge reducibility

and game semantics for these functions have been investigated by Andretta [1]. Note that a $\mathbf{\Sigma}^0_2$-admissible representation of a discrete space can be interpretted as a "$\mathbf{\Delta}^0_2$-admissible" representation, and, because they are closed under composition, a $\mathbf{\Delta}^0_2$-admissible representation can at best only determine the topology of the represented set up to $\mathbf{\Delta}^0_2$-isomorphism (i.e., a bijection that along with its inverse is a $\mathbf{\Delta}^0_2$-function).

Acknowledgements

This work was partly supported by Grant-in-Aid for Scientific Research (B) 19300046 from JSPS, and by Kyoto University Global COE "Information Education and Research Center for Knowledge-Circulating Society". We thank the anonymous reviewers for their helpful comments.

References

1. A. Andretta, *More on Wadge determinacy*, Annals of Pure and Applied Logic **144** (2006), 2–32.
2. C. E. Aull and W. J. Thron, *Separation axioms between T_0 and T_1*, Indag. Math. **24** (1963), 26–37.
3. V. Brattka, *Effective Borel measurability and reducibility of functions*, Mathematical Logic Quarterly **51** (2005), 19–44.
4. V. Brattka and P. Hertling, *Topological properties of real number representations*, Theoretical Computer Science **284** (2002), 241–257.
5. V. Brattka and M. Makananise, *Limit computable functions and subsets*, (to appear).
6. R. Engelking, *General topology*, Heldermann, 1989.
7. A. Kechris, *Classical descriptive set theory*, Springer-Verlag, 1995.
8. J. Saint Raymond, *Preservation of the Borel class under countable-compact-covering mappings*, Topology and its Applications **154** (2007), 1714–1725.
9. M. Schröder, *Extended admissibility*, Theoretical Computer Science **284** (2002), 519–538.
10. V. Selivanov, *Towards a descriptive set theory for domain-like structures*, Theoretical Computer Science **365** (2006), 258–282.
11. A. Tang, *Chain properties in $P(\omega)$*, Theoretical Computer Science **9** (1979), 153–172.
12. K. Weihrauch, *Computable analysis*, Springer-Verlag, 2000.
13. K. Weihrauch and C. Kreitz, *Representations of the real numbers and of the open subsets of the set of real numbers*, Annals of Pure and Applied Logic **35** (1987), 247–260.
14. M. Ziegler, *Revising type-2 computation and degrees of discontinuity*, Electronic Notes in Theoretical Computer Science **167** (2007), 255–274.

Uniqueness, Continuity, and Existence of Implicit Functions in Constructive Analysis

Hannes Diener[1] and Peter Schuster[2]

[1] Universität Siegen, Fachbereich 6: Mathematik
Walter-Flex-Straße 3, 57072 Siegen, Germany
diener@math.uni-siegen.de
[2] Dipartimento di Filosofia, Università degli Studi di Firenze
Via Bolognese 52, 50139 Firenze, Italy
peter.schuster@unifi.it

Abstract. We extract a quantitative variant of uniqueness from the usual hypotheses of the implicit functions theorem. This leads not only to an a priori proof of continuity, but also to an alternative, fully constructive existence proof.

1 Introduction

To show the differentiability of an implicit function one often relies on its continuity. The latter is mostly seen as a by-product of the not uncommon construction of the implicit function as the limit of a uniformly convergent sequence of continuous functions. We now show that the continuity of the implicit function is prior to its existence, and thus independent of any particular construction. More specifically, we deduce the continuity from a quantitative strengthening of the uniqueness, which in turn follows from the hypotheses one needs to impose on the equation the implicit function is expected to satisfy. The same quantitative strengthening of uniqueness enables us to ultimately give an alternative existence proof for implicit functions that is fully constructive in Bishop's sense.

We use ideas from [6], which loc.cit. have only been spelled out in the case of implicit functions with values in $\mathbb{R}$. The existence proof given in [6] therefore can rely on reasoning by monotonicity, whereas in the general case—treated in this paper—of implicit functions with values in $\mathbb{R}^m$ we need to employ an extreme value argument. Similar considerations in related contexts can be found in Sections 3.3 and 3.4 of [10] during the course of the proof of the theorem on implicit functions via the inverse mapping theorem and Banach's fixed point theorem, respectively. We refer to [11, 20] for the implicit function theorem and the open mapping theorem in computable analysis à la Weihrauch [19].

The predecessor [18] of the present paper essentially contains the same material as far as uniqueness and continuity of the implicit function are concerned. When it comes to proving existence, however, it follows the intrinsically classical argument that a continuous function on a compact set attains its minimum. This

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 131-140
http://drops.dagstuhl.de/opus/volltexte/2009/2265

argument fails being practicable constructively, unless one adds the hypothesis that there quantitatively is at most one point at which the minimum can be attained. In fact, there is a heuristic principle valid [17] even in Bishop–style constructive mathematics without countable choice: if a continuous function on a complete metric space has approximate roots and in a quantitative manner at most one root, then it actually has a root. We may refer to [17] for more on this, including the principle's history with references.

As a matter of fact, however, in the case of implicit functions the required additional hypothesis is contained in the quantitative variant of uniqueness which we find at our disposal anyway. Therefore, we only need to prove that for every parameter the given equation admits approximate solutions. Altogether we achieve the existence of an exact solution at every parameter and then, by the principle of unique choice, the existence of an implicit function: as the one and only function which assigns to every parameter the solution uniquely determined by this parameter.

The present paper as a whole is conceived in the realm of Bishop's constructive mathematics [4, 5, 7, 8]. Compared with the—so-called classical—customary way of doing mathematics, the principal characteristic of the framework created by Bishop is the exclusive use of intuitionistic logic, which allows one to view Bishop's setting as a generalisation of classical mathematics [13]. Moreover, we follow [14] in doing constructive mathematics à la Bishop without countable choice, also inasmuch as we understand real numbers as located Dedekind cuts. In particular, the so-called cotransitivity property "if $x < y$, then $x < z$ or $z < y$" amounts to say that the Dedekind cut z is located whenever x, y are rational numbers, and follows by approximation in the general case.

Avoiding countable choice is further indispensable, because we want our work to be expressible in constructive Zermelo–Fraenkel set theory (CZF) as begun in [1]: countable choice does not belong to CZF. Details on this and on CZF in general can be found in [2, 12]. We will, however, use the principle of unique choice, sometimes called the principle of non–choice. By the functions–as–graphs paradigm common to set theory, unique choice is trivially in CZF.

2 Preliminaries

We first recall that in Bishop's setting every differentiable function comes with a continuous derivative [5, Chapter 2, Section 5]. In other words, for Bishop every differentiable function is by definition continuously differentiable. We nonetheless keep speaking of continously differentiable functions, also to facilitate any reading by a classically trained person. Note in this context that in Bishop's framework continuity means uniform continuity on every compact (that is, totally bounded and complete) subset of the domain; see [15] for a discussion of this.

Secondly, although in the work of Bishop and of his followers there barely is any talk of (partial or total) differentiability for functions of several real variables, we do not develop this concept in the present paper either. According to

our opinion it is in order to take this for granted: the task of checking the classical route as far as necessary can indeed be performed in a relatively straightforward way, and is sometimes simplified by Bishop's assumption of the automatic continuity of the derivative.

For the lack of appropriate references in the constructive literature we next transfer two facts from real analysis. With Theorem 5.4 and Theorem 6.8 of [5, Chapter 2] at hand the standard proofs indeed go through constructively. (For instance, the proofs of Satz 5 and of its Corollar given in [9, I, §6] require only one addendum to the proof of the Hilfssatz: for all $K, L \in \mathbb{R}$ with $L \geqslant 0$ the implication "if $K^2 \leqslant KL$, then $K \leqslant L$" is also constructively valid. To verify this, assume that $K^2 \leqslant KL$; it suffices to prove that $K < L + \varepsilon$ for every $\varepsilon > 0$. For each $\varepsilon > 0$ either $0 < K$ or $K < \varepsilon$. In the former case, multiplying $K^2 \leqslant KL$ by $1/K > 0$ yields $K \leqslant L$; in the latter case we have $K < L + \varepsilon$ because $L \geqslant 0$.)

Lemma 1. *Let $g : W \to \mathbb{R}^n$ be a continuously differentiable mapping on an open set $W \subseteq \mathbb{R}^m$, and $c, d \in W$. If the line segment between c and d lies entirely in W, then*

$$g(d) - g(c) = \left(\int_0^1 Dg(c + t(d-c))dt \right) \cdot (d - c) \,.$$

Corollary 2. *Under the hypotheses of Lemma 1 we have*

$$\|g(d) - g(c)\| \leqslant \sup_{t \in [0,1]} \|Dg(c + t(d-c))dt\| \cdot \|d - c\| \,.$$

While Lemma 5.5 of [5, Chapter 2] is an approximative alternative of Rolle's theorem, our next lemma is a strong variant of the contrapositive.

Lemma 3. *Let $h : [c, d] \to \mathbb{R}$ be continuously differentiable, and assume that there is $r > 0$ such that $h'(x) > r$ for all $x \in [c, d]$. Furthermore assume that $c < d$. Then $h(c) < h(d)$.*

Proof. Assume that $h(d) - h(c) < \frac{r(d-c)}{4}$. By the mean value theorem [5, Theorem 5.6] there exists $\xi \in [c, d]$ such that

$$|h'(\xi)(d-c) - (h(d) - h(c))| < \frac{r(d-c)}{2} \,.$$

Then

$$\begin{aligned} h'(\xi)(d-c) - (h(d) - h(c)) &> r(d-c) - (h(d) - h(c)) \\ &> r(d-c) - \frac{r(d-c)}{4} \\ &> \frac{r(d-c)}{2} \,. \end{aligned}$$

Hence we get a contradiction, and thus $h(d) - h(c) > 0$. □

The last lemma in this section is an approximative substitute for the classical result that if a differentiable function attains its minimum at a point in the interior of a compact set, then the gradient of that function vanishes at this point.

Lemma 4. *Let $W \subseteq \mathbb{R}^n$ be an open neighbourhood of $[0,1]^n$ and $h : W \to \mathbb{R}$ a continuously differentiable function. If there is a point $\xi \in [0,1]^n$ and $s > 0$ such that*

$$h(x) > h(\xi) + s \tag{1}$$

for all $x \in \partial[0,1]^n$, then for every $\varepsilon > 0$ there exists $y \in [0,1]^n$ such that $\|\nabla h(y)\| < \varepsilon$.

Proof. For convenience we will use the supremum norm on $\mathbb{R}^n$ throughout this proof. Choose $N \in \mathbb{N}$ such that for $x, y \in [0,1]^n$, if $\|y - x\| < 2^{-N}$ then both

$$\|\nabla h(x) - \nabla h(y)\| < \frac{\varepsilon}{4} \tag{2}$$

and

$$|h(x) - h(y)| < \frac{s}{2}\,. \tag{3}$$

Let

$$G = \left\{ \left(\frac{i_1}{2^N}, \ldots, \frac{i_n}{2^N} \right) : (i_1, \ldots, i_n) \in \mathbb{N}^n \right\} \cap [0,1]^n.$$

For any $x \in G$ and $i \leqslant n$, let $x_i^{\pm}$ denote the point $x \pm 2^{-N} e_i$—i.e. the neighbouring point of x in G in the positive/negative direction of the i^{th} coordinate. For any $x \in G$ and $i \leqslant n$ fix $\lambda_{x,i}^+ \in \{-1, 0, 1\}$ and $\lambda_{x,i}^- \in \{-1, 0, 1\}$, such that

$$\begin{aligned}
\lambda_{x,i}^+ = 0 &\Rightarrow \left| \frac{\partial h}{\partial x_i}(x + 2^{-(N+1)} e_i) \right| < \frac{3\varepsilon}{4}, \\
\lambda_{x,i}^+ = -1 &\Rightarrow \frac{\partial h}{\partial x_i}(x + 2^{-(N+1)} e_i) < -\frac{\varepsilon}{2}, \\
\lambda_{x,i}^+ = 1 &\Rightarrow \frac{\partial h}{\partial x_i}(x + 2^{-(N+1)} e_i) > \frac{\varepsilon}{2}, \\
\lambda_{x,i}^- = 0 &\Rightarrow \left| \frac{\partial h}{\partial x_i}(x - 2^{-(N+1)} e_i) \right| < \frac{3\varepsilon}{4}, \\
\lambda_{x,i}^- = -1 &\Rightarrow \frac{\partial h}{\partial x_i}(x - 2^{-(N+1)} e_i) > \frac{\varepsilon}{2}, \\
\lambda_{x,i}^- = 1 &\Rightarrow \frac{\partial h}{\partial x_i}(x - 2^{-(N+1)} e_i) < -\frac{\varepsilon}{2}.
\end{aligned}$$

Notice that if $\lambda_{x,i}^+ = -1$ then for all $y \in [x, x_i^+]$

$$\frac{\partial h}{\partial x_i}(y) < -\frac{\varepsilon}{4},$$

and therefore, by Lemma 3,

$$h(x) > h(x_i^+). \tag{4}$$

Similarly, when $\lambda_{x,i}^- = -1$, we obtain

$$\frac{\partial h}{\partial x_i}(y) > \frac{\varepsilon}{4}$$

for all $y \in [x_i^-, x]$, and then

$$h(x) > h(x_i^-). \tag{5}$$

Furthermore notice that, by continuity and (2),

$$\text{if } \lambda_{x,i}^+ \in \{0,1\} \text{ and } \lambda_{x,i}^- \in \{0,1\}, \text{ then } \left|\frac{\partial h}{\partial x_i}(x)\right| < \varepsilon. \tag{6}$$

Next, because of (3), we can find $x_0 \in G$ such that $|h(x_0) - h(\xi)| < s$. If there exists i such that $\lambda_{x_0,i}^+ = -1$ (or $\lambda_{x_0,i}^- = -1$), set $x_1 = (x_0)_i^+$ (or $x_1 = (x_0)_i^-$), for which $h(x_0) > h(x_1)$. Continuing this construction we will, because of (4), never visit the same point twice and never reach a point in $\partial[0,1]^n \cap G$. Therefore, we eventually reach a point $x_m \in (0,1)^n \cap G$ for which both $\lambda_{x_m,i}^+ \neq -1$ and $\lambda_{x_m,i}^- \neq -1$ for all $1 \leqslant i \leqslant n$. By (6) this implies that $\|\nabla h(x_m)\| < \varepsilon$. □

3 Uniqueness and Continuity

Situation. *Let $U \subseteq \mathbb{R}^n$ and $V \subseteq \mathbb{R}^m$ be open neighbourhoods of $a \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$, respectively, with $m, n \geqslant 1$. We denote the coordinates on $\mathbb{R}^n$ and $\mathbb{R}^m$ by $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_m)$, respectively, and endow $\mathbb{R}^n \times \mathbb{R}^m$ with the norm $\|(x,y)\| = \|x\| + \|y\|$. The Jacobian of a partially differentiable function $F : U \times V \to \mathbb{R}^m$ at $(x,y) \in U \times V$ is written as*

$$DF(x,y) = \left(\frac{\partial F}{\partial x}(x,y), \frac{\partial F}{\partial y}(x,y)\right), \quad \frac{\partial F}{\partial x}(x,y) \in \mathbb{R}^{m\times n}, \quad \frac{\partial F}{\partial y}(x,y) \in \mathbb{R}^{m\times m}.$$

Finally, let $F : U \times V \to \mathbb{R}^m$ be a continuously differentiable function such that $\frac{\partial F}{\partial y}(a,b)$ is invertible; in particular $\nu > 0$ where

$$\nu = \left\| \frac{\partial F}{\partial y}(a,b)^{-1} \right\|.$$

Lemma 5. *For every $\lambda \in]1, +\infty[$ there are compact neighbourhoods $U_\lambda \subseteq U$ and $V_\lambda \subseteq V$ of a and b, respectively, such that for all $x \in U_\lambda$ und $y, y' \in V_\lambda$:*

$$\|y - y'\| \leqslant \lambda \cdot \nu \cdot \|F(x,y) - F(x,y')\|. \tag{7}$$

Proof. By replacing F with $\frac{\partial F}{\partial y}(a,b)^{-1} \cdot F$, we may assume that $\frac{\partial F}{\partial y}(a,b)$ is the unit matrix and therefore $\nu = 1$. Now consider

$$G : U \times V \to \mathbb{R}^m, \quad (x,y) \mapsto y - F(x,y).$$

Since G is continuously differentiable with $\frac{\partial G}{\partial y}(a,b) = 0$, there are compact neighbourhoods $U_\lambda \subseteq U$ and $V_\lambda \subseteq V$ of a and b, respectively, such that V_λ is convex and

$$\left\| \frac{\partial G}{\partial y}(x,y) \right\| \leqslant 1 - 1/\lambda \tag{8}$$

for all $(x,y) \in U_\lambda \times V_\lambda$. Then, for all $x \in U_\lambda$ and $y, y' \in V_\lambda$, we have

$$\begin{aligned} \|y - y'\| &\leqslant \|(y - G(x,y)) - (y' - G(x,y'))\| + \|G(x,y) - G(x,y')\| \\ &\leqslant \|F(x,y) - F(x,y')\| + (1 - 1/\lambda) \cdot \|y - y'\| \end{aligned}$$

by (8) and Corollary 2; whence (7) holds with $\nu = 1$. □

Throughout the following $\lambda \in]1, +\infty[$ is arbitrary and U_λ, V_λ are as in Lemma 5.

Equation (7) implies, for fixed $x \in U_\lambda$, that $y \in V_\lambda$ and $y' \in V_\lambda$ lie close together, whenever F is small at (x,y) and (x,y'). Therefore (7) can be seen as a quantitative way to express that any y with $F(x,y) = 0$ is uniquely determined by x.

This can be made more precise. We say that a function $H : S \to \mathbb{R}$ on a metric space S with $H \geqslant 0$ has *uniformly at most one* root [16] if

$$\forall \delta > 0 \, \exists \varepsilon > 0 \, \forall y, y' \in S \; (d(y,y') \geqslant \delta \Rightarrow H(y) \geqslant \varepsilon \vee H(y') \geqslant \varepsilon) \; .$$

If H has uniformly at most one root, then H has *at most one* root [3]: i.e.,

$$\forall y, y' \in S \; (y \neq y' \Rightarrow H(y) > 0 \vee H(y') > 0) \; .$$

If H has at most one root, then its root—if it exists at all—is uniquely determined:

$$\forall y, y' \in S \; (H(y) = 0 \wedge H(y') = 0 \Rightarrow y = y') \; .$$

Corollary 6. *For each $x \in U_\lambda$ the function*

$$H : V_\lambda \to \mathbb{R}, \; y \mapsto \|F(x,y)\|$$

has uniformly at most one root; in particular, for all $y, y' \in V_\lambda$,

$$F(x,y) = 0 \wedge F(x,y') = 0 \Rightarrow y = y' \, .$$

Theorem 7. *Every function $f : U_\lambda \to V_\lambda$ with $F(x, f(x)) = 0$ for all $x \in U_\lambda$ is continuous.*

Proof. Consider $\varepsilon > 0$ arbitrary. Since F is uniformly continuous on the compact set $U_\lambda \times V_\lambda$, there exists $\delta > 0$ such that

$$\|F(x,y) - F(x',y')\| \leqslant (\lambda \cdot \nu)^{-1} \cdot \varepsilon \, .$$

whenever $(x,y), (x',y') \in U_\lambda \times V_\lambda$ are such that $\|x - x'\| + \|y - y'\| < \delta$. In particular,

$$\|F(x, f(x'))\| \leqslant (\lambda \cdot \nu)^{-1} \cdot \varepsilon$$

for all $x, x' \in U_\lambda$ with $\|x - x'\| < \delta$ (recall that $F(x', f(x')) = 0$). Using this and (7) we get

$$\begin{aligned}\|f(x) - f(x')\| &\leqslant \lambda \cdot \nu \cdot \|F(x, f(x)) - F(x, f(x'))\| \\ &= \lambda \cdot \nu \cdot \|F(x, f(x'))\| \\ &\leqslant \varepsilon\end{aligned}$$

for all $x, x' \in U_\lambda$ with $\|x - x'\| < \delta$. Hence f is uniformly continuous. □

This proof's heuristic can be explained as follows. If x and x' are close, then $F(x, f(x'))$ is close to $F(x', f(x')) = 0$, and therefore close to $F(x, f(x)) = 0$; Equation (7) now implies that $f(x)$ and $f(x')$ are close.

Following the standard argument, one can now easily show that every f as in Theorem 7 is differentiable in the interior of U_λ^0 with uniformly continuous derivative

$$Df(x) = -\frac{\partial F}{\partial y}(x, f(x))^{-1} \cdot \frac{\partial F}{\partial x}(x, f(x)) .$$

Note that the quantitative version (7) of uniqueness was sufficient to prove continuity, which therefore only depends on differentiability inasmuch as this is needed to prove (7).

4 Existence

Last, we present an alternative approach to the existence of the implicit function, which is—just as the proof of continuity—based on the quantitative version (7) of uniqueness, but again requires involving the partial derivative of the given equation. An additional ingredient is the following result, for whose validity in Bishop-style constructive mathematics without choice we refer to [17, Theorem 5]:

Theorem 8. *Let S be a complete metric space and $H : S \to \mathbb{R}$ uniformly continuous. If $\inf H = 0$ and H has uniformly at most one root, then there is $y_H \in S$ with $H(y_H) = 0$.*

Note that $\inf H = 0$ means that $H \geqslant 0$ and that H has approximate roots.

From now on we also assume that $F(a, b) = 0$. (An assumption that has not been used so far.)

Theorem 9. *There are compact neighbourhoods $U_\lambda^0 \subseteq U_\lambda$ and $V_\lambda^0 \subseteq V_\lambda$ of a and b, respectively, such that there is a function $f : U_\lambda^0 \to V_\lambda^0$ with $F(x, f(x)) = 0$ for all $x \in U_\lambda^0$.*

As a by-product of Corollary 6, there is exactly one f, which by Theorem 7 is continuous.

Proof. Using Corollary 6 and the principle of unique choice, we only need to find compact neighbourhoods U_λ^0 and V_λ^0 of a and b, respectively, with $U_\lambda^0 \times V_\lambda^0 \subseteq U_\lambda \times V_\lambda$, such that for every $x \in U_\lambda^0$ there exists $y \in V_\lambda^0$ with $F(x,y) = 0$. We may also assume that $(a,b) = (0,0)$. Setting $(x,y') = (0,0)$ in (7), we get

$$\|y\| \leqslant \lambda \cdot \nu \cdot \|F(0,y)\| \tag{9}$$

for all $y \in U_\lambda$, since $F(0,0) = 0$. We can now find $r, s > 0$, such that

$$U_\lambda^0 = [-r,+r]^n \ , \quad V_\lambda^0 = [-s,+s]^m$$

completely lie in U_λ and V_λ respectively. By choosing r, s small enough, we may assume that

$$\left\| \frac{\partial F}{\partial y}(x,y)^{-1} \right\| \leqslant \nu + 1 \tag{10}$$

for all $(x,y) \in U_\lambda^0 \times V_\lambda^0$. Since F is uniformly continuous on the compact set $U_\lambda^0 \times V_\lambda^0$, by making r sufficiently small, we may further assume that

$$\lambda \cdot \nu \cdot \|F(x,y) - F(x',y)\| \leqslant s/3$$

for all $(x,y), (x',y) \in U_\lambda^0 \times V_\lambda^0$. If we now substitute $x' = 0$, we get

$$\lambda \cdot \nu \cdot \|F(x,y) - F(0,y)\| \leqslant s/3 \tag{11}$$

for all $(x,y) \in U_\lambda^0 \times V_\lambda^0$; if we also substitute $y = 0$, we get

$$\lambda \cdot \nu \cdot \|F(x,0)\| \leqslant s/3 \tag{12}$$

for all $x \in U_\lambda^0$. (If we were only interested in getting (12), it would suffice to point out that $F(\_,0)$ is continuous at 0 and that $F(0,0) = 0$.) Equations (9) and (11) imply that

$$2s/3 \leqslant \lambda \cdot \nu \cdot \|F(x,y)\| \tag{13}$$

for all $x \in U_\lambda^0$ and $y \in \partial V_\lambda^0$, i.e. $\|y\| = s$. Now consider $x \in U_\lambda^0$ arbitrary, but fixed. The function

$$h : V \to \mathbb{R}\,,\ y \mapsto \|F(x,y)\|^2$$

is differentiable with continuous derivative

$$\nabla h(y) = 2 \cdot F(x,y) \cdot \frac{\partial F}{\partial y}(x,y) \ . \tag{14}$$

By (12) and (13) we have

$$\lambda^2 \cdot \nu^2 \cdot h(0) + s^2/3 \leqslant \lambda^2 \cdot \nu^2 \cdot h(y)$$

for all $y \in \partial V_\lambda^0$; whence by virtue of Lemma 4

$$\inf_{y \in V_\lambda^0} \|\nabla h(y)\| = 0\,.$$

In view of (10) and (14) this implies

$$\inf_{y \in V_\lambda^0} \|F(x,y)\| = 0\,.$$

By Corollary 6 and Theorem 8, we achieve $y \in V_\lambda^0$ with $F(x,y) = 0$. □

Acknowledgments. A lecture on implicit functions the second author gave at the University of Bern made him thinking first of the subject matters present in this paper. Next, D.S. Bridges pointed the authors to [6]; O. Forster, H. Kalf, and the two referees of this paper gave useful hints. The first author would like to thank the departments of mathematics of the University of Canterbury and the Universität Siegen for supporting him as an outgoing postdoctoral fellow and an incoming staff member respectively. During the preparation of this paper the second author was holding a Feodor Lynen Research Fellowship for Experienced Researchers granted by the Alexander von Humboldt Foundation from sources of the German Federal Ministry of Education and Research; he is grateful to A. Cantini and G. Sambin for their hospitality in Florence and Padua, respectively.

References

1. Aczel, P., The type theoretic interpretation of constructive set theory. In: A. Macintyre, L. Pacholski, J. Paris, eds., *Logic Colloquium '77.* North–Holland, Amsterdam, 1978, pp. 55–66
2. Aczel, P., and M. Rathjen, *Notes on Constructive Set Theory.* Institut Mittag–Leffler Preprint No. 40, 2000/01
3. Berger, J., D. Bridges, and P. Schuster, The fan theorem and unique existence of maxima. *J. Symbolic Logic* 71 (2006), 713–720
4. Bishop, E., *Foundations of Constructive Analysis.* McGraw–Hill, New York, 1967
5. Bishop, E., and D. Bridges, *Constructive Analysis.* Springer, Berlin etc., 1985
6. Bridges, D., Calude, C., Pavlov, B., and Ştefănescu, D., The constructive implicit function theorem and applications in mechanics. *Chaos, Solitons & Fractals* **10** (1999), 927–934
7. Bridges, D., and F. Richman, *Varieties of Constructive Mathematics.* Cambridge University Press, Cambridge, 1987
8. Bridges, D., and L. Vîţă, *Techniques of Constructive Analysis.* Springer, New York, 2006
9. Forster, O.,*Analysis 2. Differentialrechnung im* $\mathbb{R}^n$*, gewöhnliche Differentialgleichungen.* 6nd ed., Vieweg, Wiesbaden, 2005
10. Krantz, S.G., and Parks, H.R., *The Implicit Function Theorem. History, Theory, and Applications.* Birkhäuser, Boston, 2002
11. McNicholl, T.H., A uniformly computable implicit function theorem. *MLQ Math. Log. Q.* 54 (2008), 272–279
12. Rathjen, M., Choice principles in constructive and classical set theories. In: Z. Chatzidakis, P. Koepke, W. Pohlers, eds., *Logic Colloquium '02.* Proceedings, Münster, 2002. Lect. Notes Logic 27, Assoc. Symbol. Logic, La Jolla (2006), 299–326
13. Richman, F., Intuitionism as generalization. *Philos. Math. (3)* 5 (1990), 124–128

14. Richman, F., The fundamental theorem of algebra: a constructive development without choice. *Pacific J. Math.*196 (2000), 213–230
15. Schuster, P., What is continuity, constructively? *J. UCS* 11 (2005), 2076–2085.
16. Schuster, P., Unique solutions. *Math. Log. Quart.* 52 (2006), 534–539. Corrigendum: *Math. Log. Quart.* 53 (2007), 214
17. Schuster, P., Problems, solutions, and completions. *J. Logic Algebr. Program.* (2009), doi: 10.1016/j.jlap.2009.02.007 [10 February 2009]
18. Schuster, P., Zur Eindeutigkeit impliziter Funktionen. Preprint, 2009
19. Weihrauch, K., Computable Analysis. An Introduction. Springer, Berlin, 2000.
20. Ziegler, M., Effectively open real functions. *J. Complexity* 22 (2006), 827–849

Relativizations of the P =? DNP Question for the BSS Model

Christine Gaßner

Institut für Mathematik und Informatik, Ernst-Moritz-Arndt-Universität,
Robert-Blum-Str. 2, 17487 Greifswald, Germany
gassnerc@uni-greifswald.de

Abstract. We consider the uniform BSS model of computation where the machines can perform additions, multiplications, and tests of the form $x \geq 0$. The oracle machines can also check whether a tuple of real numbers belongs to a given oracle set $\mathcal{O}$ or not. We construct oracles such that the classes P and DNP relative to these oracles are equal or not equal.

1 Introduction

The uniform BSS model of computation was introduced in [Blum et al. 1989]. The BSS machines can perform labelled instructions of the form $Z_i := Z_j + Z_k$, $Z_i := Z_j - Z_k$, $Z_i := Z_j \cdot Z_k$, $Z_j := c$, *if* $Z_j \geq 0$ *then goto* l_1 *else goto* l_2, $Z_{I_j} := Z_{I_k}$, $I_j := 1$, $I_j := I_j + 1$, and *if* $I_j = I_k$ *then goto* l_1 *else goto* l_2. Each assignment of an input $(x_1, \ldots, x_n) \in \bigcup_{i \geq 1} \mathbb{R}^i$ to the registers of a machine $\mathcal{M}$ is realized by $Z_1 := x_1; \ldots; Z_n := x_n; I_1 := n; \ldots; I_{k_\mathcal{M}} := n$. Moreover, *oracle machines* can execute *if* $(Z_1, \ldots, Z_{I_1}) \in \mathcal{O}$ *then goto* l_1 *else goto* l_2 for some oracle $\mathcal{O} \subseteq \mathbb{R}^\infty$. The *non-deterministic* machines are able to guess an arbitrary number of arbitrary elements $y_1, \ldots, y_m \in \mathbb{R}$ in one step after the input and to assign the guesses to $Z_{I_1+1}, \ldots, Z_{I_1+m}$. A (digital) non-deterministic BSS machine $\mathcal{M}$ *accepts* an input $(x_1, \ldots, x_n) \in \mathbb{R}^\infty$ if there is some guessed sequence $(y_1, \ldots, y_m) \in \mathbb{R}^\infty$ and $(y_1, \ldots, y_m) \in \{0,1\}^\infty$, respectively, such that $\mathcal{M}$ outputs 1 on input $(x_1, \ldots, x_n)$ for the guesses $y_1, \ldots, y_m$. Let $\mathrm{P}_\mathbb{R}$, $\mathrm{DNP}_\mathbb{R}$, and $\mathrm{NP}_\mathbb{R}$ be the classes of problems recognized by deterministic, digital non-deterministic, and non-deterministic machines, respectively, in polynomial time. Let $\mathrm{P}^\mathcal{O}_\mathbb{R}$, $\mathrm{DNP}^\mathcal{O}_\mathbb{R}$, and $\mathrm{NP}^\mathcal{O}_\mathbb{R}$ are the corresponding classes for one given oracle $\mathcal{O}$. We have $\mathrm{P}_\mathbb{R} \subseteq \mathrm{DNP}_\mathbb{R} \subseteq \mathrm{NP}_\mathbb{R}$ and $\mathrm{P}^\mathcal{O}_\mathbb{R} \subseteq \mathrm{DNP}^\mathcal{O}_\mathbb{R} \subseteq \mathrm{NP}^\mathcal{O}_\mathbb{R}$.

In [Baker et al. 1975] and [Emerson 1994] for Turing machines and the BSS model, respectively, oracles were defined in order to get the equality of relativized versions of $\mathrm{P}_{(\mathbb{R})}$ and $\mathrm{NP}_{(\mathbb{R})}$. Such a *universal oracle* $\mathcal{O}$ can be defined by $\mathcal{O} = \bigcup_{i \geq 1} W_i$ where $W_0 = \emptyset$ and

$$W_i = \{(\underbrace{1, \ldots, 1}_{t\times}, \boldsymbol{x}, \mathit{Code}(\mathcal{M})) \in \mathbb{R}^i \mid$$
$$\mathcal{M} \text{ is a non-deterministic machine using } \textstyle\bigcup_{j<i} W_j \text{ as oracle } \ \& \ \mathcal{M}(\boldsymbol{x}) \downarrow^t\}.$$

Thus, we get the following.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 141-148
http://drops.dagstuhl.de/opus/volltexte/2009/2266

Proposition 1. *There is an oracle $\mathcal{O}$ such that* $\mathrm{P}_{\mathbb{IR}}^{\mathcal{O}} = \mathrm{DNP}_{\mathbb{IR}}^{\mathcal{O}} = \mathrm{NP}_{\mathbb{IR}}^{\mathcal{O}}$.

In [Emerson 1994] Emerson presented also an oracle $\mathcal{Q}$ such that $\mathrm{P}_{\mathbb{IR}}^{\mathcal{Q}} \neq \mathrm{NP}_{\mathbb{IR}}^{\mathcal{Q}}$. Emerson's proof technique also allows to separate relativized versions of $\mathrm{DNP}_{\mathbb{IR}}$ and $\mathrm{NP}_{\mathbb{IR}}$ by a diagonalization procedure in the following way. Let $U \subseteq \mathbb{IR}^\infty$ be a set of codes $\boldsymbol{u}$ representing all pairs that contain a polynomial $p_{\boldsymbol{u}}$ and the program $P_{\boldsymbol{u}}$ of a digital non-deterministic oracle BSS machine. Let $\mathcal{N}_{\boldsymbol{u}}^{\mathcal{B}}$ be the machine using an oracle $\mathcal{B} \subseteq \mathbb{IR}^\infty$ and performing only $p_{\boldsymbol{u}}(n)$ instructions of P_u on inputs of size n. Let the oracle $\mathcal{Q}_1 = \bigcup_{i\geq 1} W_i$ be defined in stages. Let $V_0 = \emptyset$.

Stage $i \geq 1$:

Let

$K_i = \{\boldsymbol{u} \in U \mid (\forall \mathcal{B} \subseteq \mathbb{IR}^\infty)$
$(\mathcal{N}_{\boldsymbol{u}}^{\mathcal{B}}$ does not use any $r > i$ in a query on input $\boldsymbol{u})\}$,

$W_i = \bigcup_{k<i} V_k$,

$V_i = \{(i+1, \boldsymbol{u}) \mid \boldsymbol{u} \in K_i \;\&\; \mathcal{N}_{\boldsymbol{u}}^{W_i}$ does not accept $\boldsymbol{u}\}$.

The defined sequence of codes, $K_1, K_2, \ldots$, covers the set of all digital non-deterministic oracle machines recognizing problems in $\mathrm{DNP}_{\mathbb{IR}}^{\mathcal{B}}$ for some $\mathcal{B}$. Consequently we get $L_1 = \{\boldsymbol{y} \mid (\exists n \in \mathbb{N}^+)((n, \boldsymbol{y}) \in \mathcal{Q}_1)\} \notin \mathrm{DNP}_{\mathbb{IR}}^{\mathcal{Q}_1}$. On the other hand, we have $L_1 \in \mathrm{NP}_{\mathbb{IR}}^{\mathcal{Q}_1}$ since a non-deterministic BSS machine can guess each integer in one step.

Proposition 2. *There is an oracle $\mathcal{Q}$ such that* $\mathrm{DNP}_{\mathbb{IR}}^{\mathcal{Q}} \neq \mathrm{NP}_{\mathbb{IR}}^{\mathcal{Q}}$.

Moreover, by analogy with [Gaßner 2009] it is also possible to show $\mathrm{DNP}_{\mathbb{IR}}^{\mathbb{Z}} \neq \mathrm{NP}_{\mathbb{IR}}^{\mathbb{Z}}$ and $\mathrm{DNP}_{\mathbb{IR}}^{\mathbb{Q}} \neq \mathrm{NP}_{\mathbb{IR}}^{\mathbb{Q}}$.

It remains to show that there are also oracles such that the classes $\mathrm{P}_{\mathbb{IR}}$ and $\mathrm{DNP}_{\mathbb{IR}}$ relative to these oracles are not equal. For the computation over structures of enumerable signature, a method to separate relativized classes of problems recognized by deterministic and digital non-deterministic machines, respectively, goes back to T. Baker, J. Gill, and R. Solovay [Baker et al. 1975]. In order to obtain the inequality between relativized versions of P_{R} and $\mathrm{DNP}_{\mathsf{R}}$ for oracle machines over the ordered ring $\mathsf{R} = (\mathbb{IR}; 0, 1; +, -, \cdot; \geq)$, we can use the enumerability of all polynomials $p : \mathbb{N} \to \mathbb{N}$ and all programs of deterministic oracle machines and diagonalization techniques by analogy with [Baker et al. 1975]. Let $i \in \mathbb{N}^+$ be the code of the pair containing the i^{th} polynomial p_i and the i^{th} program P_i of a deterministic oracle machine using only the machine constants 0 and 1. If $\mathcal{N}_i^{\mathcal{B}}$ is the machine which uses an oracle $\mathcal{B} \subseteq \mathbb{IR}^\infty$ and performs only $p_i(n)$ instructions of P_i on inputs of size n, then the definition of the oracle $\mathcal{Q}_2 = \bigcup_{i\geq 1} W_i$ is possible in stages as in [Baker et al. 1975]. Let $V_0 = \emptyset$ and $m_0 = 0$.

Stage $i \geq 1$: Let n_i be any integer such that $n_i > m_{i-1}$ and $p_i(n_i) + n_i < 2^{n_i}$. Moreover, let

$W_i = \bigcup_{j<i} V_j$,

$V_i = \{\boldsymbol{x} \in \{0,1\}^{n_i} \mid \mathcal{N}_i^{W_i}$ rejects $(0, \ldots, 0) \in \mathbb{IR}^{n_i}$
$\&\; \boldsymbol{x}$ is not queried by $\mathcal{N}_i^{W_i}$ on input $(0, \ldots, 0) \in \mathbb{IR}^{n_i}\}$,

$m_i = 2^{n_i}$.
For $L_2 = \{\boldsymbol{y} \mid (\exists i \in \mathbb{N}^+)(\boldsymbol{y} \in \mathbb{R}^{n_i} \ \&\ V_i \neq \emptyset)\}$, we get $L_2 \in \mathrm{DNP}_{\mathsf{R}}^{\mathcal{Q}_2} \setminus \mathrm{P}_{\mathsf{R}}^{\mathcal{Q}_2}$.

Proposition 3. *For BSS machines using only the constants* 0 *and* 1*, there is an oracle* $\mathcal{Q}$ *such that* $\mathrm{P}_{\mathsf{R}}^{\mathcal{Q}} \neq \mathrm{DNP}_{\mathsf{R}}^{\mathcal{Q}}$.

This method as well as Emerson's method fail if we want to construct an oracle such that the relativized versions of $\mathrm{P}_{\mathbb{R}}$ and $\mathrm{DNP}_{\mathbb{R}}$ are not equal. We cannot enumerate the programs of all deterministic BSS machines, and the digital nondeterministic machines cannot guess any integer in one step. A discussion about the possibilities to transfer the ideas of [Baker et al. 1975] and [Emerson 1994] was done in [Gaßner 2008] for several types of groups. This discussion gives also insights which constructions can be used for which types of rings. In the next section we want to show that it is still possible to use diagonalization techniques for separating the classes $\mathrm{P}_{\mathbb{R}}$ and $\mathrm{DNP}_{\mathbb{R}}$ relative to an oracle. Our construction requires to consider a sequence of sequences of sets of machines and consequently a new recursive definition in every stage of a recursive definition. Techniques of this kind are often used if more natural decision problems having special properties are not known. For models of computation over algebraic structures, a summary of papers where these techniques have been applied is given, for instance, in [Bürgisser 1999]. In the last section we derive a suitable oracle from the Real Knapsack Problem such that the resulting relativized versions of $\mathrm{P}_{\mathbb{R}}$ and $\mathrm{DNP}_{\mathbb{R}}$ are also not equal. This construction is possible without using the powerful diagonalization techniques.

2 The Separation of Relativized Versions of $\mathrm{P}_{\mathbb{R}}$ and $\mathrm{DNP}_{\mathbb{R}}$ by Diagonalization Techniques

Now let us consider again the BSS machines over $(\mathbb{R}; \mathbb{R}; +, -, \cdot; \geq)$ where any real number can be a machine constant. Since we also want to define an oracle

$$\mathcal{Q}_3 \subseteq \bigcup_{i \geq 1} \mathbb{N}^{n_i}$$

recursively, we will at first define a suitable sequence $((\mathcal{K}_{i,j})_{j \geq 1})_{i \geq 1}$ of sequences containing all deterministic oracle BSS machines working in polynomial time. For any oracle $\mathcal{B} \subseteq \mathbb{R}^\infty$, any deterministic oracle BSS machine $\mathcal{N}^{\mathcal{B},c_1,\ldots,c_k}$ is determined by its machine constants $c_1, \ldots, c_k$ and a program P which is encoded by a tuple in $\{0,1\}^\infty$. Let every character of the program P, including the indices $j \in \{1, \ldots, k\}$ of the constants c_j, be unambiguously translated into a finite sequence in $\{0,1\}^\infty$ and let the oracle queries be encoded independently of the used oracle $\mathcal{B}$ by taking the same sequence of characters 0 and 1 as code for all oracle queries. Consequently, the set Prog of all programs of oracle machines and the set poly of all polynomial functions of $\mathbb{N}$ into $\mathbb{N}$ are enumerable. We will take the positive integers in order

– to enumerate all $(p_1, P_1), (p_2, P_2), (p_3, P_3), \ldots \in \mathsf{poly} \times \mathsf{Prog}$,

– to characterize the behavior of all oracle machines on special inputs of size $n_1, n_2, \ldots$ by additional numbers $N_{\mathrm{char}}(i, c_1, \ldots, c_{k_i})$ which are dependent only on the following properties:
  - The machines perform $p_i(n_i)$ instructions on inputs of size n_i.
  - The machines use only the reals $c_1, \ldots, c_{k_i}$ as machine constants.
  - The inputs belong to $\{0\}^{n_i-1} \times \{N \in \mathbb{N} \mid N \geq N_{\mathrm{char}}(i, c_1, \ldots, c_{k_i})\}$.

The Definition of the Machine $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$. *Any $i \in \mathbb{N}^+$ is the number of a pair $(p_i, P_i) \in \mathsf{poly} \times \mathsf{Prog}$ which determines a class of deterministic oracle machines $\{\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}} \mid \mathcal{B} \subseteq \mathbb{R}^\infty \ \&\ c_1, \ldots, c_{k_i} \in \mathbb{R}\}$ by the following.*

(a) The BSS machine $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ performs the instructions of the program P_i.
(b) If $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ queries an oracle, then $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ uses the oracle $\mathcal{B}$.
(c) The only constants of $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ are $c_1, \ldots, c_{k_i}$ encoded by $1, \ldots, k_i$ in the code of P_i.
(d) The number of the instructions of P_i carried out by $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ is simultaneously counted by $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ by means of an additional index register.
(e) For any input in $\mathbb{R}^n$, the machine $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ halts after at most $p_i(n)$ steps of the execution of P_i. (The bound $p_i(n)$ can be computed by using index registers.)
(f) If the output of P_i is reached in this time, then $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ outputs the value determined by P_i, $\mathcal{B}$, and $c_1, \ldots, c_{k_i}$. If the output instruction of P_i is not reached in this time, then $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ rejects the input.

Then, for any oracle $\mathcal{B} \subseteq \mathbb{R}^\infty$ and any problem $\mathcal{P} \in \mathrm{P}_{\mathbb{R}}^{\mathcal{B}}$ there are an $i \geq 1$ and constants $c_1, \ldots, c_{k_i}$ such that the machine $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ decides $\mathcal{P}$.

Let us now characterize the behavior of $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ on inputs of the form $(0, \ldots, 0, x) \in \mathbb{N}^{n_i}$. The value of any register computed by $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ on these inputs can be described by some term of the form (1). We are especially interested in oracles $\mathcal{B} \subseteq \mathbb{N}^\infty$.

The Definition of the Number $N_{\mathrm{char}}(i, c_1, \ldots, c_{k_i})$. *We consider the sequence $(f_1, f_2, \ldots, f_s)$ containing all polynomials $f_k \in \mathbb{R}[x]$ whose values $f_k(x)$ can be described by the terms of the following form*

$$\sum_{j=0}^{2^{p_i(n_i)}} \Big(\sum_{j_1,\ldots,j_{k_i}=0}^{2^{p_i(n_i)}} \alpha_{j_1,\ldots,j_{k_i},j} c_1^{j_1} \cdots c_{k_i}^{j_{k_i}} \Big) x^j \tag{1}$$

where any $\alpha_{j_1,\ldots,j_{k_i},j} \in \mathbb{Z} \cap [-2^{p_i(n_i)}, 2^{p_i(n_i)}]$. Let $N_{\mathrm{char}}(i, c_1, \ldots, c_{k_i})$ be the Cantor number of $(\mu_1, \ldots, \mu_s, \nu_1, \ldots, \nu_s, \mu, \mu', \nu)$ given by

$$\mu_k = \mathrm{code}(f_k) \in \mathbb{N}^+ \qquad \text{if } f_k \in \mathbb{Q}[x], \tag{2}$$

$$\mu_k = 0 \qquad \text{if } f_k \notin \mathbb{Q}[x], \tag{3}$$

$$\nu_k = \lim_{x\to\infty} \operatorname{sgn}(f_k(x)), \tag{4}$$

$$\mu = \min \bigcap_{\substack{k=1,\ldots,s \\ \text{degree}(f_k)\geq 1}} \{n \in \mathbb{N} \mid \forall x(f_k(x) = 0 \vee f_k(x) = 1 \Rightarrow n > x)\}, \tag{5}$$

$$\mu' = \min \bigcap_{\substack{k=1,\ldots,s \\ \mu_k = 0}} \{n \in \mathbb{N} \mid (\forall x \in \mathbb{N})(f_k(x) \in \mathbb{N} \Rightarrow n > x)\}, \tag{6}$$

$$\nu = \min \bigcap_{k=1,\ldots,s} \{n \in \mathbb{N} \mid f_k(n) < 2^n\}. \tag{7}$$

Remark 1. Here, $\operatorname{sgn}(x) = 1$ iff $x > 0$, $\operatorname{sgn}(x) = -1$ iff $x < 0$, and $\operatorname{sgn}(0) = 0$. Because of the following lemma (cp. [Gaßner 2009]), the minimum of the set in (6) exists.

Lemma 1. *For any polynomial $p \in \mathbb{R}[x] \setminus \mathbb{Q}[x]$, there is only a finite number of rational numbers $q \in \mathbb{Q}$ satisfying $p(q) \in \mathbb{Q}$.*

The Definition of $\mathcal{K}_{i,j}$ and the Constants $C_{i,1}, C_{i,2}, \ldots$. *For $i \geq 1$, let $N_{i,1}, N_{i,2}, \ldots$ be an enumeration of the set*

$$\{N_{\text{char}}(i, c_1, \ldots, c_{k_i}) \mid c_1, \ldots, c_{k_i} \in \mathbb{R}\}$$

such that $N_{i,j+1} > N_{i,j}$. For $i, j \geq 1$, let

$$\mathcal{K}_{i,j} = \{\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}} \mid \mathcal{B} \subseteq \mathbb{R}^\infty \;\&\; N_{i,j} = N_{\text{char}}(i, c_1, \ldots, c_{k_i})\}.$$

Moreover, let $C_{i,1} = N_{i,1}$ and, for $j \geq 2$, let $C_{i,j} = \max\{2^{C_{i,j-1}}, N_{i,j}\}$.
Since $C_{i,j} \geq N_{i,j} > \max\{\mu, \mu', \nu\}$, we have the following properties.

(i) By (5), $N_{i,j}$ is greater than any zero of the corresponding function f_k if $\text{degree}(f_k) \geq 1$. Therefore, by (4) we have

$$\nu_k = \operatorname{sgn}(f_k(N_{i,j})) = \operatorname{sgn}(f_k(C_{i,j})).$$

(ii) If an oracle machine $\mathcal{M} \in \mathcal{K}_{i,j}$ computes a positive integer N on input $\boldsymbol{x} \in \{0\}^{n_i-1} \times \{C_{i,j}\}$, then, by (6) there is a $k \leq s$ such that $\mu_k \neq 0$ and $N = f_k(C_{i,j})$. In this case, $f_k \in \mathbb{Q}[x]$ follows from (3). That means because of (2) that N is uniquely determined by μ_k and, consequently, by $N_{i,j}$.

(iii) A consequence of (7) is that $\mathcal{M} \in \mathcal{K}_{i,j}$ cannot compute the positive integers $C_{i,j+1}, C_{i,j+2}, \ldots$ on input $\boldsymbol{x} \in \{0\}^{n_i-1} \times \{C_{i,j}\}$ within $p_i(n_i)$ steps since these numbers are greater than $2^{C_{i,j}}$.

(iv) Property (5) implies also that $\mathcal{M} \in \mathcal{K}_{i,j}$ computes an integer $N \in \{0, 1\}$ on input $\boldsymbol{x} \in \{0\}^{n_i-1} \times \{C_{i,j}\}$ only if there is a $k \leq s$ such that $\text{degree}(f_k) = 0$ and consequently $f_k(x) = 0$ for all $x \in \mathbb{R}$ or $f_k(x) = 1$ for all $x \in \mathbb{R}$.

In the following construction, for any $\mathcal{B}$ and any $i, j \geq 1$, let $\mathcal{K}_{i,j}^{\mathcal{B}}$ be the subset of $\mathcal{K}_{i,j}$ given by

$$\mathcal{K}_{i,j}^{\mathcal{B}} = \{\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}} \mid N_{i,j} = N_{\mathrm{char}}(i, c_1, \ldots, c_{k_i})\}.$$

The Construction of $\mathcal{Q}_3$. *Let $m_0 = 0$. We construct the set $\mathcal{Q}_3$ in stages. Stage $i \geq 1$: Let n_i be an integer such that $n_i > m_{i-1}$, $p_i(n_i) < 2^{n_i-1}$, and $p_i(n_i) + n_i < 2^{n_i}$. Let $V_{i,0} = \emptyset$. Stage $j \geq 1$:*

$$W_{i,j} = \bigcup_{i'<i,} V_{i'} \cup \bigcup_{j'<j,} V_{i,j'},$$

$$V_{i,j} = \{\boldsymbol{x} \in \{0,1\}^{n_i-1} \times \{C_{i,j}\} \mid (\exists \mathcal{M} \in \mathcal{K}_{i,j}^{W_{i,j}})(\mathcal{M} \textit{ rejects } (0,\ldots,0,C_{i,j})$$

$$\& \ \boldsymbol{x} \textit{ is not queried by } \mathcal{M} \textit{ on input } (0,\ldots,0,C_{i,j}) \in \mathbb{N}^{n_i})\}.$$

Moreover, let $V_i = \bigcup_{j\geq 1} V_{i,j}$ and $m_i = 2^{n_i}$.
Finally, let $\mathcal{Q}_3 = \bigcup_{i\geq 1} V_i$ and

$$L_3 = \bigcup_{i\geq 1} \{(y_1,\ldots,y_{n_i-1},N) \in \{0,1\}^{n_i-1} \times \mathbb{N} \mid V_i \cap (\{0,1\}^{n_i-1} \times \{N\}) \neq \emptyset)\}.$$

The contents of the registers of any $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}}$ can be described by (1) if the input has the form $(0,\ldots,0,x) \in \mathbb{R}^{n_i}$. For any $\mathcal{B} \subseteq \mathbb{N}^\infty$, the value $N_{i,j}$ and the oracle $\mathcal{B}$ determine the computation path of any machine $\mathcal{N}_i^{\mathcal{B},c_1,\ldots,c_{k_i}} \in \mathcal{K}_{i,j}^{\mathcal{B}}$ traversed by the input $(0,\ldots,0,C_{i,j}) \in \mathbb{N}^{n_i}$ uniquely since (i) and (ii) hold. By (i), the result of a test of the form $f_k(C_{i,j}) \geq 0$ follows from $\mathrm{sgn}(v_k)$. The question $(f_{i_1}(C_{i,j}),\ldots,f_{i_t}(C_{i,j})) \in \mathcal{B}$? is answered no if one of the values $f_{i_1}(C_{i,j}),\ldots,f_{i_t}(C_{i,j})$ is not in $\mathbb{N}$. If the values are in $\mathbb{N}$, then, by (ii), the answer results from the values $\mu_{i_1},\ldots,\mu_{i_t}$ which are given by $N_{i,j}$.

Thus, the computation paths covered by $\mathcal{N}_i^{W_{i,j},c_1,\ldots,c_{k_i}} \in \mathcal{K}_{i,j}^{W_{i,j}}$ and by $\mathcal{N}_i^{\mathcal{Q}_3,c_1,\ldots,c_{k_i}} \in \mathcal{K}_{i,j}^{\mathcal{Q}_3}$
on $(0,\ldots,0,C_{i,j}) \in \mathbb{N}^{n_i}$ are even the same since we have also the following properties.

- By (iii), any $\boldsymbol{x} \in \{0,1\}^{n_i-1} \times \{C_{i,j+1}, C_{i,j+2}, \ldots\}$ is not queried.
- The length of the tuples in the oracle queries is less than 2^{n_i} and consequently less than n_{i+1} by definition of n_{i+1}.
- The machines $\mathcal{N}_i^{W_{i,j},c_1,\ldots,c_{k_i}}$ and $\mathcal{N}_i^{\mathcal{Q}_3,c_1,\ldots,c_{k_i}}$ do not query the tuples in $V_{i,j}$.

Moreover, for all $i, j \geq 1$, $p_i(n_i) < 2^{n_i-1}$ and (iv) imply that $V_{i,j}$ contains a tuple in $\{0,1\}^{n_i-1} \times \{C_{i,j}\}$ if a machine in $\mathcal{K}_{i,j}^{W_{i,j}}$ and, hence, any machine $\mathcal{N}_i^{W_{i,j},c_1,\ldots,c_{k_i}} \in \mathcal{K}_{i,j}^{W_{i,j}}$ and, consequently, any machine $\mathcal{N}_i^{\mathcal{Q}_3,c_1,\ldots,c_{k_i}} \in \mathcal{K}_{i,j}^{\mathcal{Q}_3}$ reject $(0,\ldots,0,C_{i,j}) \in \mathbb{R}^{n_i}$. That implies $L_3 \notin \mathrm{P}_{\mathbb{R}}^{\mathcal{Q}_3}$ and therefore the following.

Lemma 2. $L_3 \in \mathrm{DNP}_{\mathbb{R}}^{\mathcal{Q}_3} \setminus \mathrm{P}_{\mathbb{R}}^{\mathcal{Q}_3}$.

Proposition 4. *There is an oracle $\mathcal{Q}$ such that $\mathrm{P}_{\mathbb{R}}^{\mathcal{Q}} \neq \mathrm{DNP}_{\mathbb{R}}^{\mathcal{Q}}$.*

3 An Oracle Derived from the Knapsack Problem

The *Real Knapsack Problem*

$$\mathrm{KP}_{\mathbb{R}} = \bigcup_{n=1}^{\infty} \{(x_1, \ldots, x_n) \in \mathbb{R}^n \mid (\exists(\alpha_1, \ldots, \alpha_n) \in \{0,1\}^n)(\sum_{i=1}^{n} \alpha_i x_i = 1)\}$$

was introduced in [Blum et al. 1989] and studied, for instance, in [Koiran 1994] and [Meer 1992]. $\mathrm{KP}_{\mathbb{R}}$ belongs to $\mathrm{DNP}_{\mathbb{R}}$ since, for an input $(x_1, \ldots, x_n) \in \mathbb{R}^\infty$, a digital non-deterministic machine can guess any sequence $(\alpha_1, \ldots, \alpha_n) \in \{0,1\}^n$ and compute $\alpha_1 x_1 + \cdots + \alpha_n x_n$. It is not known whether $\mathrm{KP}_{\mathbb{R}} \in \mathrm{P}_{\mathbb{R}}$ holds.

Let $E_0 = \mathbb{Q}$, let $\tau_1, \tau_2, \ldots$ be a sequence of transcendental numbers such that τ_{i+1} is transcendental over $E_i =_{\mathrm{df}} E_{i-1}(\tau_i)$, and let the oracle $\mathcal{Q}_4$ and the decision problem L_4 be given.

$$A_n = \{(v_1, \ldots, v_{2n}) \in \{0, v\}^{2n} \mid v \in \mathbb{Z} \setminus \{0\} \ \&\ \textstyle\sum_{i=1}^{2n} v_i = nv\}.$$

$$\mathcal{Q}_4 = \textstyle\bigcup_{n=1}^{\infty} \{(\mathrm{sgn}(|v_1|), \ldots, \mathrm{sgn}(|v_{2n}|), \sum_{i=1}^{2n} v_i \tau_i) \in \mathbb{R}^{2n+1} \mid (v_1, \ldots, v_{2n}) \in A_n\}.$$

$$L_4 = \textstyle\bigcup_{n=1}^{\infty} \{(0, \ldots, 0, r) \in \mathbb{R}^{2n+1} \mid (\exists(v_1, \ldots, v_{2n}) \in A_n)(r = \sum_{i=1}^{2n} v_i \tau_i)\}.$$

Let us assume that the BSS machine $\mathcal{M}$ decides L_4 by using the oracle $\mathcal{Q}_4$ within a time bounded by a polynomial p and that $\mathcal{M}$ has only the constants $c_1, \ldots, c_k$. Let $F_0 = \bigcup_{i=0}^{\infty} E_i$. For $i = 1, \ldots, k$, let $F_i = F_{i-1}$ and $d_i = 1$ if $c_i \in F_{i-1}$, let $F_i = F_{i-1}(c_i)$ and $d_i = \infty$ if c_i is not algebraic over F_{i-1}, and let $F_i = F_{i-1}[c_i]$ if there is an irreducible polynomial $p_i \in F_{i-1}[x]$ of degree $d_i \geq 2$ with $p_i(c_i) = 0$. The value of any register computed by $\mathcal{M}$ on input $(0, \ldots, 0, x) \in \mathbb{R}^m$ can be described by some term of the form $\sum_{j_1, \ldots, j_k, j \leq 2^{p(m)}} \alpha_{j_1, \ldots, j_k, j} c_1^{j_1} \cdots c_k^{j_k} x^j$ where $\alpha_{j_1, \ldots, j_k, j} \in \mathbb{Z}$ and, consequently, by a polynomial of the form

$$q_j(x) = \tfrac{1}{r_0} \textstyle\sum_{j=0}^{2^{p(m)}} r_{j+1} x^j \text{ where}$$

$$r_j \quad = \sum_{\substack{m_1, \ldots, m_{i_0} \leq m_0 \\ j_s < \min\{d_s, j_0\}}} z_{m_1, \ldots, m_{i_0}, j_1, \ldots, j_k, j} \tau_1^{m_1} \cdots \tau_{i_0}^{m_{i_0}} c_1^{j_1} \cdots c_k^{j_k}$$

for some i_0, m_0, j_0, and $z_{m_1, \ldots, m_{i_0}, j_1, \ldots, j_k, j} \in \mathbb{Z}$ and $z_{m'_1, \ldots, m'_{i_0}, j'_1, \ldots, j'_k, 0} \neq 0$ for certain $m'_1, \ldots, m'_{i_0}, j'_1, \ldots, j'_k$. Thus, for the inputs of the form $(0, \ldots, 0, x) \in \mathbb{R}^m$, a non-trivial oracle query $(z_1, \ldots, z_s, q_j(x)) \in \mathcal{Q}_4$? (where $\mathrm{degree}(q_j) \geq 1$) can only be answered yes if $q_j(x) = \sum_{i=1}^{2n'} v'_i \tau_i$ is satisfied for some $(v'_1, \ldots, v'_{2n'}) \in A_{n'}$. Thus, we get the following.

Lemma 3. *Let $n > i_0$, $(0, \ldots, 0, v_{i_0+1}, \ldots, v_{2n}) \in A_n$, and $x = \sum_{i=i_0+1}^{2n} v_i \tau_i$. For $v_l \neq 0$, $v_{l+1} = \cdots = v_{2n} = 0$, a non-trivial oracle query $(z_1, \ldots, z_s, q_j(x)) \in \mathcal{Q}_4$? can be answered yes on inputs of the form $(0, \ldots, 0, x)$ only if $s \geq 2n$ and*

$$(z_{i_0+1}, \ldots, z_s) = (\mathrm{sgn}(|v_{i_0+1}|), \ldots, \mathrm{sgn}(|v_l|), 0, \ldots, 0).$$

Let n_0 be an even positive integer such that $n_0 > 2i_0$ and $p(2n_0+1) < 2^{\frac{n_0}{2}}$. Let P be the computation path of $\mathcal{M}$ described for inputs $(0,\dots,0,x)$ of size $2n_0+1$ uniquely by conditions of the form

$$(g_{j,1}(x),\dots,g_{j,s_j}(x)) \notin \mathcal{Q}_4 \quad (j \le t') \qquad \text{and} \qquad f_1(x) > 0,\dots,f_t(x) > 0$$

where $g_{j,1},\dots,g_{j,s_j}$ are polynomials, degree$(g_{j,s_j}) > 0$, and each f_j is defined by some equation of the form $f_j(x) = x^{n_j} + a_{n_j-1}x^{n_j-1} + \cdots + a_1x + a_0$.

Let $\tau > 0$ be transcendental over F_k and greater than all zeros of $f_1,\dots,f_t$. Then, $(0,\dots,0,\tau) \in \mathbb{R}^{2n_0+1} \setminus L_4$ traverses the path P. If $g_{j,1}(\tau),\dots,g_{j,s_j-1}(\tau) \in \{0,1\}$, then the polynomials $g_{j,1},\dots,g_{j,s_j-1}$ are constant. Since we have $|G| < 2^{\frac{n_0}{2}}$ for

$$G = \bigcup_{j<p(2n_0+1)} \{(g_{j,i_0+1}(x),\dots,g_{j,2n_0}(x)) \mid g_{j,i_0+1},\dots,g_{j,2n_0} \text{ are constant functions}\},$$

there is some $(0,\dots,0,x_0) \in \mathbb{R}^{2n_0+1}$ with $x_0 = \sum_{i=i_0+1}^{2n_0} w_i\tau_i$ satisfying

a) $(0,\dots,0,w_{i_0+1},\dots,w_{2n_0}) \in A_{n_0}$ and $w_{2n_0} \neq 0$,
b) $x_0 > \max(\{\tau\} \cup \bigcup_{\substack{j<p(2n_0+1)\\ s\le s_j-1}} \{x \mid g_{j,s}(x) \in \{0,1\} \ \&\ \text{degree}(g_{j,s}) \ge 1\})$,
c) $(\text{sgn}(|w_{i_0+1}|),\dots,\text{sgn}(|w_{2n_0}|)) \notin G$.

a) implies that $(0,\dots,0,x_0) \in L_4$. Moreover, we have $f_j(x_0) > 0$ by b). Therefore, by Lemma 3 and c), P is also traversed by $(0,\dots,0,x_0) \in \mathbb{R}^{2n_0+1}$. Hence, we get the following.

Lemma 4. $L_4 \in \text{DNP}_{\mathbb{R}}^{\mathcal{Q}_4} \setminus \text{P}_{\mathbb{R}}^{\mathcal{Q}_4}$.

Proposition 5. *There is an oracle $\mathcal{Q}$ which can be derived from* $\text{KP}_{\mathbb{R}}$ *such that* $\text{P}_{\mathbb{R}}^{\mathcal{Q}} \neq \text{DNP}_{\mathbb{R}}^{\mathcal{Q}}$.

References

[Baker et al. 1975] Baker, T., J. Gill, and R. Solovay: "Relativizations of the P =? NP question"; SIAM J. Comput. 4 (1975), 431–442.

[Blum et al. 1989] Blum, L., M. Shub, and S. Smale: "On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines"; Bulletin of the Amer. Math. Soc. 21 (1989), 1–46.

[Bürgisser 1999] Bürgisser, P.: "On the structure of Valiant's complexity classes"; Discrete Mathematics & Theoretical Computer Science 3(3) (1999), 73-94.

[Emerson 1994] Emerson, T.: "Relativizations of the P =? NP question over the reals (and other ordered rings)"; Theoretical Computer Science 133 (1994), 15–22.

[Gaßner 2008] Gaßner, C.: "On the power of relativized versions of P, DNP, and NP for groups"; (2008) Submitted.

[Gaßner 2009] Gaßner, C.: "Oracles and relativizations of the P =? NP question for several structures"; JUCS vol. 15 no. 6 (2009), 1186–1205.

[Koiran 1994] Koiran, P.: "Computing over the reals with addition and order"; Theoretical Computer Science 133 (1994), 35–47.

[Meer 1992] Meer, K.: "A note on a P ≠ NP result for a restricted class of real machines"; Journal of Complexity 8 (1992), 451–453.

[Poizat 1995] Poizat, B.: "Les Petits Cailloux"; Aléas (1995).

Curves That Must Be Retraced

Xiaoyang Gu[1] [§] [‡], Jack H. Lutz[1] [§] [¶], and Elvira Mayordomo[2] [¶†]

[1] Department of Computer Science, Iowa State University, Ames, IA 50011, USA.
Email: {xiaoyang,lutz}@cs.iastate.edu
[2] Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, 50018 Zaragoza, Spain. Email: elvira@unizar.es

Abstract. We exhibit a polynomial time computable plane curve $\mathbf{\Gamma}$ that has finite length, does not intersect itself, and is smooth except at one endpoint, but has the following property. For every computable parametrization f of $\mathbf{\Gamma}$ and every positive integer m, there is some positive-length subcurve of $\mathbf{\Gamma}$ that f retraces at least m times. In contrast, every computable curve of finite length that does not intersect itself has a constant-speed (hence non-retracing) parametrization that is computable relative to the halting problem.

1 Introduction

A curve is a mathematical model of the path of a particle undergoing continuous motion. Specifically, in a Euclidean space $\mathbb{R}^n$, a curve is the range Γ of a continuous function $f : [a, b] \to \mathbb{R}^n$ for some $a < b$. The function f, called a *parametrization* of Γ, clearly contains more information than the pointset Γ, namely, the precise manner in which the particle "traces" the points $f(t) \in \Gamma$ as t, which is often considered a time parameter, varies from a to b. When the particle's motion is algorithmically governed, the parametrization must be computable (as a function on the reals; see below).

This paper shows that the geometry of a curve Γ may force every *computable* parametrization f of Γ to retrace various parts of its path (i.e., "go back and forth along Γ") many times, even when Γ is an efficiently computable, smooth, finite-length curve that does not intersect itself. In fact, our main theorem exhibits a plane curve $\mathbf{\Gamma} \subseteq \mathbb{R}^2$ with the following properties.

1. $\mathbf{\Gamma}$ is *simple*, i.e., it does not intersect itself.

[‡] Research supported in part by National Science Foundation Grant 0830479.
[§] Research supported in part by National Science Foundation Grants 0344187, 0652569, and 0728806.
[¶] Research supported in part by the Spanish Ministry of Education and Science (MEC) and the European Regional Development Fund (ERDF) under project TIN2005-08832-C03-02.
[†] Part of this author's research was performed during a visit at Iowa State University, supported by Spanish Government (Secretaría de Estado de Universidades e Investigación del Ministerio de Educación y Ciencia) grant for research stays PR2007-0368.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 149-160
http://drops.dagstuhl.de/opus/volltexte/2009/2267

2. $\mathbf{\Gamma}$ is *rectifiable*, i.e., it has finite length.
3. $\mathbf{\Gamma}$ is *smooth except at one endpoint*, i.e., $\mathbf{\Gamma}$ has a tangent at every interior point and a 1-sided tangent at one endpoint, and these tangents vary continuously along $\mathbf{\Gamma}$.
4. $\mathbf{\Gamma}$ is *polynomial time computable* in the strong sense that there is a polynomial time computable position function $\vec{s} : [0,1] \to \mathbb{R}^2$ such that the velocity function $\vec{v} = \vec{s}'$ and the acceleration function $\vec{a} = \vec{v}'$ are polynomial time computable; the total distance traversed by $\vec{s}$ is finite; and $\vec{s}$ parametrizes $\mathbf{\Gamma}$, i.e., $\mathrm{range}(\vec{s}) = \mathbf{\Gamma}$.
5. $\mathbf{\Gamma}$ *must be retraced* in the sense that every parametrization $f : [a,b] \to \mathbb{R}^2$ of $\mathbf{\Gamma}$ that is computable in *any* amount of time has the following property. For every positive integer m, there exist disjoint, closed subintervals $I_0, \ldots, I_m$ of $[a,b]$ such that the curve $\Gamma_0 = f(I_0)$ has positive length and $f(I_i) = \Gamma_0$ for all $1 \leq i \leq m$. (Hence f retraces Γ_0 at least m times.)

The terms "computable" and "polynomial time computable" in properties 4 and 5 above refer to the "bit-computability" model of computation on reals formulated in the 1950s by Grzegorczyk [9] and Lacombe [17], extended to feasible computability in the 1980s by Ko and Friedman [13] and Kreitz and Weihrauch [16], and exposited in the recent paper by Braverman and Cook [4] and the monographs [20,14,22,5]. As will be shown here, condition 4 also implies that the pointset $\mathbf{\Gamma}$ is polynomial time computable in the sense of Brattka and Weihrauch [2]. (See also [22,3,4].)

A fundamental and useful theorem of classical analysis states that every simple, rectifiable curve Γ has a *normalized constant-speed parametrization*, which is a one-to-one parametrization $f : [0,1] \to \mathbb{R}^n$ of Γ with the property that $f([0,t])$ has arclength tL for all $0 \leq t \leq 1$, where L is the length of Γ. (A simple, rectifiable curve Γ has exactly two such parametrizations, one in each direction, and standard terminology calls either of these *the* normalized constant-speed parametrization $f : [0,1] \to \mathbb{R}^n$ of Γ. The constant-speed parametrization is also called the *parametrization by arclength* when it is reformulated as a function $f : [0,L] \to \mathbb{R}^n$ that moves with constant speed 1 along Γ.) Since the constant-speed parametrization does not retrace any part of the curve, our main theorem implies that this classical theorem is not entirely constructive. Even when a simple, rectifiable curve has an efficiently computable parametrization, the constant-speed parametrization need not be computable.

In addition to our main theorem, we prove that every simple, rectifiable curve Γ in $\mathbb{R}^n$ with a computable parametrization has the following two properties.

I. The length of Γ is lower semicomputable.
II. The constant-speed parametrization of Γ is computable relative to the length of Γ.

These two things are not hard to prove if the computable parametrization is one-to-one, (in fact, they follow from results of Müller and Zhao [19] in this case) but our results hold even when the computable parametrization retraces portions of the curve many times.

Taken together, I and II have the following two consequences.

1. The curve $\mathbf{\Gamma}$ of our main theorem has a finite length that is lower semi-computable but not computable. (The existence of polynomial-time computable curves with this property was first proven by Ko [15].)
2. Every simple, rectifiable curve Γ in $\mathbb{R}^n$ with a computable parametrization has a constant-speed parametrization that is Δ_2^0-computable, i.e., computable relative to the halting problem. Hence, the existence of a constant-speed parametrization, while not entirely constructive, is constructive relative to the halting problem.

2 Length, Computability, and Complexity of Curves

In this section we summarize basic terminology and facts about curves. As we use the terms here, a *curve* is the range Γ of a continuous function $f : [a, b] \to \mathbb{R}^n$ for some $a < b$. The function f is called a *parametrization* of Γ. Each curve clearly has infinitely many parametrizations.

A curve is *simple* if it has a parametrization that is one-to-one, i.e., the curve "does not intersect itself". The length of a simple curve Γ is defined as follows. Let $f : [a, b] \overset{1-1}{\to} \mathbb{R}^n$ be a one-to-one parametrization of Γ. For each *disection* $\vec{t}$ of $[a, b]$, i.e., each tuple $\vec{t} = (t_0, \ldots, t_m)$ with $a = t_0 < t_1 < \ldots < t_m = b$, define the *$f$-$\vec{t}$-approximate length* of Γ to be

$$\mathcal{L}_{\vec{t}}^{f}(\Gamma) = \sum_{i=0}^{m-1} |f(t_{i+1}) - f(t_i)|.$$

Then the *length* of Γ is

$$\mathcal{L}(\Gamma) = \sup_{\vec{t}} \mathcal{L}_{\vec{t}}^{f}(\Gamma),$$

where the supremum is taken over all dissections $\vec{t}$ of $[a, b]$. It is easy to show that $\mathcal{L}(\Gamma)$ does not depend on the choice of the one-to-one parametrization f, i.e. that the length is an intrinsic property of the pointset Γ.

In sections 4 and 5 of this paper we use a more general notion of length, namely, the 1-dimensional Hausdorff measure $\mathcal{H}^1(\Gamma)$, which is defined for *every* set $\Gamma \subseteq \mathbb{R}^n$. We refer the reader to [7] for the definition of $\mathcal{H}^1(\Gamma)$. It is well known that $\mathcal{H}^1(\Gamma) = \mathcal{L}(\Gamma)$ holds for every simple curve Γ.

A curve Γ is *rectifiable*, or *has finite length* if $\mathcal{L}(\Gamma) < \infty$. In sections 4 and 5 we use the notation $\mathcal{RC}$ for the set of all rectifiable simple curves.

Definition. Let $f : [a, b] \to \mathbb{R}^n$ be continuous.

1. For $m \in \mathbb{Z}^+$, f has *m-fold retracing* if there exist disjoint, closed subintervals $I_0, \ldots, I_m$ of $[a, b]$ such that the curve $\Gamma_0 = f(I_0)$ has positive length and $f(I_i) = \Gamma_0$ for all $1 \leq i \leq m$.
2. f is *non-retracing* if f does not have 1-fold retracing.
3. f has *bounded retracing* if there exists $m \in \mathbb{Z}^+$ such that f does not have m-fold retracing.
4. f has *unbounded retracing* if f does not have bounded retracing, i.e., if f has m-fold retracing for all $m \in \mathbb{Z}^+$.

We now review the notions of computability and complexity of a real-valued function. An *oracle* for a real number t is any function $O_t : \mathbb{N} \to \mathbb{Q}$ with the property that $|O_t(s) - t| \leq 2^{-s}$ holds for all $s \in \mathbb{N}$. A function $f : [a,b] \to \mathbb{R}^n$ is *computable* if there is an oracle Turing machine M with the following property. For every $t \in [a,b]$ and every precision parameter $r \in \mathbb{N}$, if M is given r as input and *any* oracle O_t for t as its oracle, then M outputs a rational point $M^{O_t}(r) \in \mathbb{Q}^n$ such that $|M^{O_t}(r) - f(t)| \leq 2^{-r}$. A function $f : [a,b] \to \mathbb{R}^n$ is *computable in polynomial time* if there is an oracle machine M that does this in time polynomial in $r + l$, where l is the maximum length of the query responses provided by the oracle.

An *oracle* for a function $f : [a,b] \to \mathbb{R}^n$ is any function $\mathcal{O}_f : ([a,b] \cap \mathbb{Q}) \times \mathbb{N} \to \mathbb{Q}^n$ with the property that $|\mathcal{O}_f(q,r) - f(q)| \leq 2^{-r}$ holds for all $q \in [a,b] \cap \mathbb{Q}$ and $r \in \mathbb{N}$. A decision problem A is *Turing reducible* to a function $f : [a,b] \to \mathbb{R}^n$, and we write $A \leq_{\mathrm{T}} f$, if there is an oracle Turing machine M such that, for every oracle $\mathcal{O}_f$ for f, $M^{\mathcal{O}_f}$ decides A. It is easy to see that, if f is computable, then $A \leq_{\mathrm{T}} f$ if and only if A is decidable.

A curve is *computable* if it has a parametrization $f : [a,b] \to \mathbb{R}^n$, where $a, b \in \mathbb{Q}$ and f is computable. A curve is *computable in polynomial time* if it has a parametrization that is computable in polynomial time.

3 An Efficiently Computable Curve That Must Be Retraced

This section presents our main theorem, which is the existence of a smooth, rectifiable, simple plane curve $\mathbf{\Gamma}$ that is parametrizable in polynomial time but not computably parametrizable in any amount of time without unbounded retracing. Intuitively, our curve $\mathbf{\Gamma}$ has, for each $n \in \mathbb{N}$, a section of the form illustrated in Figure 3.1. The height $h(n)$ is positive, and the halting problem K is encoded into the width $w(n)$. Oversimplifying a bit, $w(n)$ is $2^{-(n+\tau(n))}$, where $\tau(n)$ is the number of steps executed by the nth Turing machine on input n. Thus $w(n)$ is 0 if $n \in K$, and $w(n)$ is so small as to be "indistinguishable" from 0 if $n \notin K$. The smallness of $w(n)$ implies that we can efficiently compute a parametrization that is retracing when $w(n)$ is 0. However, as we show in Lemma 3.12, a *nonretracing* parametrization must have a vertical component that distinguishes the case $w(n) = 0$ from the case $w(n) > 0$, and hence must solve the halting problem. It follows that no nonretracing parametrization is computable.

We now give a precise construction of the curve $\mathbf{\Gamma}$, followed by a brief discussion of how the construction achieves the intuition that we have just described. The rest of the section is devoted to proving that $\mathbf{\Gamma}$ has the desired properties.

Construction 3.1 *(1) For each $a, b \in \mathbb{R}$ with $a < b$, define the functions $\varphi_{a,b}, \xi_{a,b} : [a,b] \to \mathbb{R}$ by*

$$\varphi_{a,b}(t) = \frac{b-a}{4} \sin \frac{2\pi(t-a)}{b-a}$$

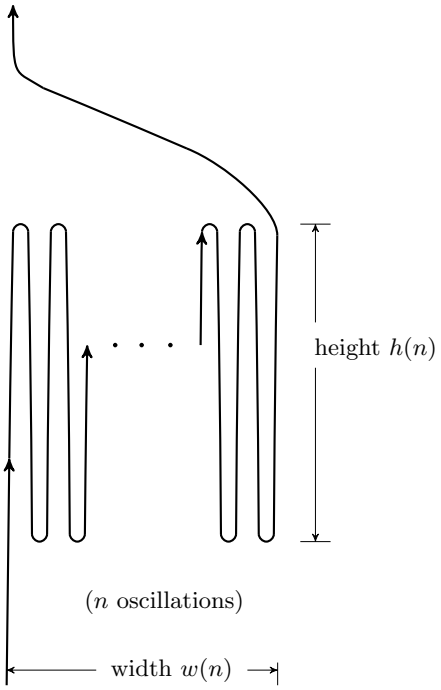


Fig. 3.1. Schematic view of the n^{th} section of $\mathbf{\Gamma}$

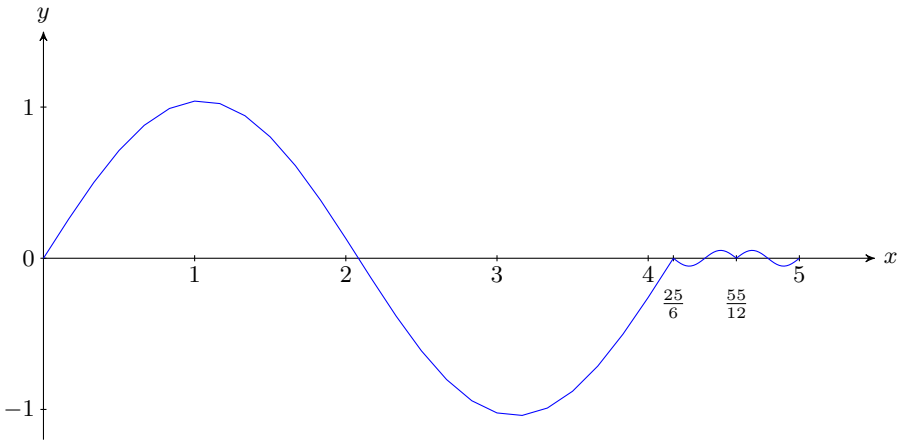


Fig. 3.2. $\psi_{0,5,1}$

and

$$\xi_{a,b}(t) = \begin{cases} -\varphi_{a,\frac{a+b}{2}}(t) & \text{if } a \le t \le \frac{a+b}{2} \\ \varphi_{\frac{a+b}{2},b}(t) & \text{if } \frac{a+b}{2} \le t \le b. \end{cases}$$

(2) For each $a, b \in \mathbb{R}$ with $a < b$ and each positive integer n, define the function $\psi_{a,b,n} : [a,b] \to \mathbb{R}$ by

$$\psi_{a,b,n}(t) = \begin{cases} \varphi_{a,d_0}(t) & \text{if } a \le t \le d_0 \\ \xi_{d_{i-1},d_i}(t) & \text{if } d_{i-1} \le t \le d_i, \end{cases}$$

where

$$d_i = \frac{a+5b}{6} + i\frac{b-a}{6n}$$

for $0 \le i \le n$. (See Figure 3.2.)

(3) Fix a standard enumeration $M_1, M_2, \ldots$ of (deterministic) Turing machines that take positive integer inputs. For each positive integer n, let $\tau(n)$ denote the number of steps executed by M_n on input n. It is well known that the diagonal halting problem

$$K = \left\{ n \in \mathbb{Z}^+ \mid \tau(n) < \infty \right\}$$

is undecidable.

(4) Define the horizontal and vertical acceleration functions $a_x, a_y : [0,1] \to \mathbb{R}$ as follows. For each $n \in \mathbb{N}$, let

$$t_n = \int_0^n e^{-x} dx = 1 - e^{-n},$$

noting that $t_0 = 0$ and that t_n converges monotonically to 1 as $n \to \infty$. Also, for each $n \in \mathbb{Z}^+$, let

$$t_n^- = \frac{t_{n-1} + 4t_n}{5}, \quad t_n^+ = \frac{6t_n - t_{n-1}}{5},$$

noting that these are symmetric about t_n and that $t_n^+ \le t_{n+1}^-$.

(i) For $0 \le t \le 1$, let

$$a_x(t) = \begin{cases} -2^{-(n+\tau(n))}\xi_{t_n^-,t_n^+}(t) & \text{if } t_n^- \le t < t_n^+ \\ 0 & \text{if no such } n \text{ exists}, \end{cases}$$

where $2^{-\infty} = 0$.

(ii) For $0 \le t < 1$, let

$$a_y(t) = \psi_{t_{n-1},t_n,n}(t),$$

where n is the unique positive integer such that $t_{n-1} \le t < t_n$.

(iii) Let $a_y(1) = 0$.

(5) Define the horizontal and vertical velocity and position functions $v_x, v_y, s_x, s_y : [0,1] \to \mathbb{R}$ *by*

$$v_x(t) = \int_0^t a_x(\theta)d\theta, \quad v_y(t) = \int_0^t a_y(\theta)d\theta,$$
$$s_x(t) = \int_0^t v_x(\theta)d\theta, \quad s_y(t) = \int_0^t v_y(\theta)d\theta.$$

(6) Define the vector acceleration, velocity, and position functions $\vec{a}, \vec{v}, \vec{s} : [0,1] \to \mathbb{R}^2$ *by*

$$\vec{a}(t) = (a_x(t), a_y(t)),$$
$$\vec{v}(t) = (v_x(t), v_y(t)),$$
$$\vec{s}(t) = (s_x(t), s_y(t)).$$

(7) Let $\mathbf{\Gamma} = \text{range}(\vec{s})$.

Intuitively, a particle at rest at time $t = a$ and moving with acceleration given by the function $\varphi_{a,b}$ moves forward, with velocity increasing to a maximum at time $t = \frac{a+b}{2}$ and then decreasing back to 0 at time $t = b$. The vertical acceleration function a_y, together with the initial conditions $v_y(0) = s_y(0) = 0$ implied by (5), thus causes a particle to move generally upward (i.e., $s_y(t_0) < s_y(t_1) < \cdots$), coming to momentary rests at times $t_1, t_2, t_3, \ldots$. Between two consecutive such stopping times t_{n-1} and t_n, the particle's vertical acceleration is controlled by the function $\psi_{t_{n-1},t_n,n}$. This function causes the particle's vertical motion to do the following between times t_{n-1} and t_n.

(i) From time t_{n-1} to time $\frac{t_{n-1}+5t_n}{6}$, move upward from elevation $s_y(t_{n-1})$ to elevation $s_y(t_n)$.
(ii) From time $\frac{t_{n-1}+5t_n}{6}$ to time t_n, make n round trips to a lower elevation $s \in (s_y(t_{n-1}), s_y(t_n))$.

In the meantime, the horizontal acceleration function a_x, together with the initial conditions $v_x(0) = s_x(0) = 0$ implied by (5), ensure that the particle remains on or near the y-axis. The deviations from the y-axis are simply described: The particle moves to the right from time $\frac{t_{n-1}+4t_n}{5}$ through the completion of the n round trips described in (ii) above and then moves to the y-axis between times t_n and $\frac{6t_n-t_{n-1}}{5}$. The amount of lateral motion here is regulated by the coefficient $2^{-(n+\tau(n))}$. If $\tau(n) = \infty$, then there is no lateral motion, and the n round trips in (ii) are retracings of the particle's path. If $\tau(n) < \infty$, then these n round trips are "forward" motion along a curvy part of $\mathbf{\Gamma}$. In fact, $\mathbf{\Gamma}$ contains points of arbitrarily high curvature, but the particle's motion is kinematically realistic in the sense that the acceleration vector $\vec{a}(t)$ is polynomial time computable, hence continuous and bounded on the interval $[0,1]$. Figure 3.3 illustrates the path of the particle from time t_{n-1} to t_{n+1} with $n = 1$ and hypothetical (model dependent!) values $\tau(1) = 1$ and $\tau(2) = 2$.

The rest of this section is devoted to proving the following theorem concerning the curve $\mathbf{\Gamma}$.

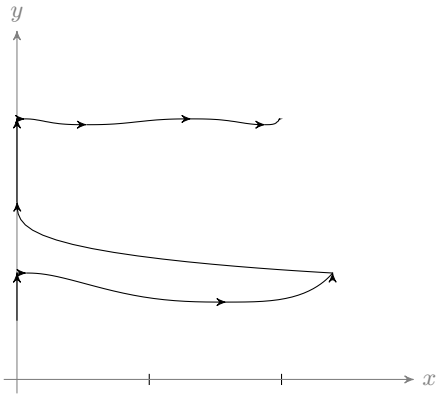


Fig. 3.3. Example of $\vec{s}(t)$ from t_0 to t_2

Theorem 3.2. *(main theorem). Let $\vec{a}, \vec{v}, \vec{s}$, and $\mathbf{\Gamma}$ be as in Construction 3.1.*

1. *The functions $\vec{a}, \vec{v}$, and $\vec{s}$ are Lipschitz and computable in polynomial time, hence continuous and bounded.*
2. *The total length, including retracings, of the parametrization $\vec{s}$ of $\mathbf{\Gamma}$ is finite and computable in polynomial time.*
3. *The curve $\mathbf{\Gamma}$ is simple, rectifiable, and smooth except at one endpoint.*
4. *Every computable parametrization $f : [a, b] \to \mathbb{R}^2$ of $\mathbf{\Gamma}$ has* unbounded retracing.

For the remainder of this section, we use the notation of Construction 3.1.

The following two observations facilitate our analysis of the curve $\mathbf{\Gamma}$. The proofs are routine calculations.

Observation 3.3 *For all $n \in \mathbb{Z}^+$, if we write*

$$d_i^{(n)} = \frac{t_{n-1} + 5t_n}{6} + i\frac{t_n - t_{n-1}}{6n}$$

and

$$e_i^{(n)} = d_i^{(n)} + \frac{t_n - t_{n-1}}{12n}$$

for all $0 \leq i < n$, then

$$t_{n-1} < t_n^- < d_0^{(n)} < e_0^{(n)} < d_1^{(n)} < e_1^{(n)} < \cdots < d_{n-1}^{(n)} < e_{n-1}^{(n)} < t_n < t_n^+ < t_{n+1}^-.$$

Observation 3.4 *For all $a, b \in \mathbb{R}$ with $a < b$,*

$$\int_a^b \int_a^t \varphi_{a,b}(\theta) d\theta dt = \frac{(b-a)^3}{8\pi}.$$

We now proceed with a quantitative analysis of the geometry of $\mathbf{\Gamma}$. We begin with the horizontal component of $\vec{s}$.

Lemma 3.5 *1. For all* $t \in [0,1] - \bigcup_{n\in K}(t_n^-, t_n^+)$, $v_x(t) = s_x(t) = 0$.
2. For all $n \in K$ *and* $t \in (t_n^-, t_n)$, $v_x(t) > 0$.
3. For all $n \in K$ *and* $t \in (t_n, t_n^+)$, $v_x(t) < 0$.
4. For all $n \in \mathbb{Z}^+$, $s_x(t_n) = \frac{(e-1)^3}{1000\pi e^{3n}} 2^{-(n+\tau(n))}$.
5. $s_x(1) = 0$.

The following lemma analyzes the vertical component of $\vec{s}$. We use the notation of Observation 3.3, with the additional proviso that $d_n^{(n)} = t_n$.

Lemma 3.6 *1. For all* $n \in \mathbb{Z}^+$ *and* $t \in (t_{n-1}, d_0^{(n)})$, $v_y(t) > 0$.
2. For all $n \in \mathbb{Z}^+$, $0 \le i < n$, *and* $t \in (d_i^{(n)}, e_i^{(n)})$, $v_y(t) < 0$.
3. For all $n \in \mathbb{Z}^+$, $0 \le i < n$, *and* $t \in (e_i^{(n)}, d_{i+1}^{(n)})$, $v_y(t) > 0$.
4. For all $n \in \mathbb{Z}^+$, $0 \le i < n$, *and* $t \in \{e_i^{(n)}, d_i^{(n)}, t_n\}$, $v_y(t) = 0$.
5. For all $n \in \mathbb{Z}^+$ *and* $0 \le i \le n$, $s_y(d_i^{(n)}) = s_y(d_0^{(n)})$.
6. For all $n \in \mathbb{Z}^+$ *and* $0 \le i < n$, $s_y(e_i^{(n)}) = s_y(e_0^{(n)})$.
7. For all $n \in \mathbb{N}$, $s_y(t_n) = \frac{5^3(e-1)^3}{6^3\cdot 8\pi} \sum_{i=1}^{n} \frac{1}{e^{3i}}$.
8. For all $n \in \mathbb{Z}^+$, $s_y(e_0^{(n)}) = s_y(t_n) - \frac{(e-1)^3}{12^3 n^3 8\pi e^{3n}}$.
9. $s_y(1) = \frac{5^3(e-1)^3}{6^3\cdot 8\pi(e^3-1)}$.

By Lemmas 3.5 and 3.6, we see that $\vec{s}$ parametrizes a curve from $\vec{s}(0) = (0,0)$ to $\vec{s}(1) = (0, \frac{5^3(e-1)^3}{6^3 8\pi(e^3-1)})$.

It is clear from Observation 3.3 and Lemmas 3.5 and 3.6 that the curve Γ does not intersect itself. We thus have the following.

Corollary 3.7 $\mathbf{\Gamma}$ *is a simple curve from* $\vec{s}(0) = (0,0)$ *to* $\vec{s}(1) = (0, \frac{5^3(e-1)^3}{6^3 8\pi(e^3-1)})$.

Lemma 3.8 *The functions* $\vec{a}, \vec{v}$, *and* $\vec{s}$ *are Lipschitz, hence continuous, on* $[0,1]$.

Since every Lipschitz parametrization has finite total length [1], and since the length of a curve cannot exceed the total length of any of its parametrizations, we immediately have the following.

Corollary 3.9 *The total length, including retracings, of the parametrization* $\vec{s}$ *is finite. Hence the curve* $\mathbf{\Gamma}$ *is rectifiable.*

Lemma 3.10 *The curve* $\mathbf{\Gamma}$ *is smooth except at the endpoint* $\vec{s}(1)$.

Lemma 3.11 *The functions* $\vec{a}, \vec{v}$, *and* $\vec{s}$ *are computable in polynomial time. The total length including retracings, of* $\vec{s}$ *is computable in polynomial time.*

Definition. A *modulus of uniform continuity* for a function $f : [a, b] \to \mathbb{R}^n$ is a function $h : \mathbb{N} \times \mathbb{N}$ such that, for all $s, t \in [a, b]$ and $r \in \mathbb{N}$,

$$|s - t| \leq 2^{-h(r)} \implies |f(s) - f(t)| \leq 2^{-r}.$$

It is well known (e.g., see [14]) that every computable function $f : [a, b] \to \mathbb{R}^n$ has a modulus of uniform continuity that is computable.

Lemma 3.12 *Let $f : [a, b] \to \mathbb{R}^2$ be a parametrization of $\mathbf{\Gamma}$. If f has bounded retracing and a computable modulus of uniform continuity, then $\mathrm{K} \leq_{\mathrm{T}} f_y$, where f_y is the vertical component of f.*

4 Lower Semicomputability of Length

In this section we prove that every computable curve Γ has a lower semicomputable length. Our proof is somewhat involved, because our result holds even if every computable parametrization of Γ is retracing.

Construction 4.1 *Let $f : [0, 1] \to \mathbb{R}^n$ be a computable function. Given an oracle Turing machine M that computes f and a computable modulus $m : \mathbb{N} \to \mathbb{N}$ of the uniform continuity of f, the (M, m)-cautious polygonal approximator of* range(f) *is the function $\pi_{M,m} : \mathbb{N} \to \{$polygonal paths$\}$ computed by the following algorithm.*

input $r \in \mathbb{N}$;
$S := \{\}$; // S may be a multi-set
for i:=0 **to** $2^{m(r)}$ **do**
  $a_i := i2^{-m(r)}$;
  use M to compute x_i with
    $|x_i - f(a_i)| \leq 2^{-(r+m(r)+1)}$;
  add x_i to S;
output a longest path inside a minimum spanning tree of S.

Definition. Let (X, d) be a metric space. Let $\Gamma \subseteq X$ and $\epsilon > 0$. Let

$$\Gamma(\epsilon) = \left\{ p \in X \;\middle|\; \inf_{p' \in \Gamma} d(p, p') \leq \epsilon \right\}$$

be the *Minkowski sausage* of Γ with radius ϵ.

Let $d_{\mathrm{H}} : \mathcal{P}(X) \times \mathcal{P}(X) \to \mathbb{R}$ be such that for all $\Gamma_1, \Gamma_2 \in \mathcal{P}(X)$

$$d_{\mathrm{H}}(\Gamma_1, \Gamma_2) = \inf \{\epsilon \mid \Gamma_1 \subseteq \Gamma_2(\epsilon) \text{ and } \Gamma_2 \subseteq \Gamma_1(\epsilon)\}.$$

Note that d_{H} is the *Hausdorff distance* function.

Let $\mathcal{K}(X)$ be the set of nonempty compact subsets of X. Then $(\mathcal{K}(X), d_{\mathrm{H}})$ is a metric space [6].

Theorem 4.2. *(Frink [8], Michael [18]). Let (X, d) be a compact metric space. Then $(\mathcal{K}(X), d_{\mathrm{H}})$ is a compact metric space.*

Definition. Let $\mathcal{RC}$ be the set of all simple rectifiable curves in $\mathbb{R}^n$.

Theorem 4.3. *([21] page 55). Let $\Gamma \in \mathcal{RC}$. Let $\{\Gamma_n\}_{n\in\mathbb{N}} \subseteq \mathcal{RC}$ be a sequence of rectifiable curves such that $\lim_{n\to\infty} d_{\mathrm{H}}(\Gamma_n, \Gamma) = 0$. Then $\mathcal{H}^1(\Gamma) \leq \liminf_{n\to\infty} \mathcal{H}^1(\Gamma_n)$.*

This theorem has the following consequence.

Theorem 4.4. *Let $\Gamma \in \mathcal{RC}$. For all $\epsilon > 0$, there exists $\delta > 0$ such that for all $\Gamma' \in \mathcal{RC}$, if $d_{\mathrm{H}}(\Gamma, \Gamma') < \delta$, then $\mathcal{H}^1(\Gamma') > \mathcal{H}^1(\Gamma) - \epsilon$.*

Theorem 4.5. *Let $\Gamma \in \mathcal{RC}$ such that $\Gamma = \gamma([0, 1])$, where γ is a continuous function. (Note that γ may not be one-one.) Let $S(a) = \{\gamma(a_i) \mid a_i \in a\}$ for all dissection a. Let $\{a_n\}_{n\in\mathbb{N}}$ be a sequence of dissections of Γ such that*

$$\lim_{n\to\infty} \mathrm{mesh}(a_n) = 0.$$

Then

$$\lim_{n\to\infty} \mathcal{H}^1(LMST(a_n)) = \mathcal{H}^1(\Gamma),$$

where $LMST(a)$ is the longest path inside the Minimum Euclidean Spanning Tree of $S(a)$.

This result implies that when the sampling density is high, the number of leaves in the minimum spanning tree is asymptotically smaller than the total number of nodes.

We now have the machinery to prove the main result of this section.

Theorem 4.6. *Let $\gamma : [0, 1] \to \mathbb{R}^n$ be computable such that $\Gamma = \gamma([0, 1]) \in \mathcal{RC}$. Then $\mathcal{H}^1(\Gamma)$ is lower semicomputable.*

5 Δ_2^0-Computability of the Constant-Speed Parametrization

In this section we prove that every computable curve Γ has a constant speed parametrization that is Δ_2^0-computable.

Theorem 5.1. *Let $\Gamma = \gamma^*([0, 1]) \in \mathcal{RC}$. ($\gamma^*$ may not be one-one.) Let $l = \mathcal{H}^1(\Gamma)$ and O_l be an oracle such that for all $n \in \mathbb{N}$, $|O_l(n) - l| \leq 2^{-n}$. Let f be a computation of γ^* with modulus m. Let γ be the constant speed parametrization of Γ. Then γ is computable with oracle O_l.*

Corollary 5.2 *Let Γ be a curve with the property described in property 5 of Theorem 3.2. Then the length of Γ – $\mathcal{H}^1(\Gamma)$ is not computable.*

Acknowledgment. We thank anonymous referees for their valuable comments.

References

1. T. M. Apostol. *Introduction to Analytic Number Theory.* Undergraduate Texts in Mathematics. Springer-Verlag, 1976.
2. V. Brattka and K. Weihrauch. Computability on subsets of Euclidean space I: Closed and compact subsets. *Theoretical Computer Science*, 219:65–93, 1999.
3. M. Braverman. On the complexity of real functions. In *Forty-Sixth Annual IEEE Symposium on Foundations of Computer Science*, 2005.
4. M. Braverman and S. Cook. Computing over the reals: Foundations for scientific computing. *Notices of the AMS*, 53(3):318–329, 2006.
5. M. Braverman and M. Yampolsky. *Computability of Julia Sets.* Springer, 2008.
6. G. A. Edgar. *Measure, topology, and fractal geometry.* Springer-Verlag, 1990.
7. K. Falconer. *Fractal Geometry: Mathematical Foundations and Applications.* Wiley, second edition, 2003.
8. O. Frink, Jr. Topology in lattices. *Transactions of the American Mathematical Society*, 51(3):569–582, 1942.
9. A. Grzegorczyk. Computable functionals. *Fundamenta Mathematicae*, 42:168–202, 1955.
10. X. Gu, J. H. Lutz, and E. Mayordomo. Points on computable curves. In *Proceedings of the Forty-Seventh Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006)*, pages 469–474. IEEE Computer Society Press, 2006.
11. J. Hershberger and S. Suri. An optimal algorithm for euclidean shortest paths in the plane. *SIAM Journal on Computing*, 28(6):2215–2256, 1999.
12. S. Kapoor and S. N. Maheshwari. Efficient algorithms for euclidean shortest path and visibility problems with polygonal obstacles. In *Proceedings of the fourth annual symposium on computational geometry*, pages 172–182, New York, NY, USA, 1988. ACM Press.
13. K. Ko and H. Friedman. Computational complexity of real functions. *Theoretical Computer Science*, 20:323–352, 1982.
14. K.-I. Ko. *Complexity Theory of Real Functions.* Birkhäuser, Boston, 1991.
15. K.-I. Ko. A polynomial-time computable curve whose interior has a nonrecursive measure. *Theoretical Computer Science*, 145:241–270, 1995.
16. C. Kreitz and K. Weihrauch. Complexity theory on real numbers and functions. In *Theoretical Computer Science*, volume 145 of *Lecture Notes in Computer Science.* Springer, 1982.
17. D. Lacombe. Extension de la notion de fonction recursive aux fonctions d'une ou plusiers variables reelles, and other notes. *Comptes Rendus*, 240:2478-2480; 241:13-14, 151-153, 1250-1252, 1955.
18. E. Michael. Topologies on spaces of subsets. *Transactions of the American Mathematical Society*, 71(1):152–182, 1951.
19. N. T. Müller and X. Zhao. Jordan areas and grids. In *Proceedings of the Fifth International Conference on Computability and Complexity in Analysis*, pages 191–206, 2008.
20. M. B. Pour-El and J. I. Richards. *Computability in Analysis and Physics.* Springer-Verlag, 1989.
21. C. Tricot. *Curves and Fractal Dimension.* Springer-Verlag, 1995.
22. K. Weihrauch. *Computable Analysis. An Introduction.* Springer-Verlag, 2000.

Effective Dispersion in Computable Metric Spaces

Zvonko Iljazović

University of Zagreb, Croatia

Abstract. We investigate the relationship between computable metric spaces (X, d, α) and (X, d, β), where (X, d) is a given metric space. In the case of Euclidean space, α and β are equivalent up to isometry, which does not hold in general. We introduce the notion of effectively dispersed metric space. This notion is essential in the proof of the main result of this paper: (X, d, α) is effectively totally bounded if and only if (X, d, β) is effectively totally bounded, i.e. the property that a computable metric space is effectively totally bounded (and in particular effectively compact) depends only on the underlying metric space.

1 Introduction

Let $k \in \mathbb{N}$, $k \geq 1$. We say that a function $f : \mathbb{N}^k \to \mathbb{Q}$ is **recursive** if there exist recursive functions $a, b, c : \mathbb{N}^k \to \mathbb{N}$ such that $f(x) = (-1)^{c(x)} \frac{a(x)}{b(x)+1}$, $\forall x \in \mathbb{N}^k$. A function $f : \mathbb{N}^k \to \mathbb{R}$ is said to be **recursive** if there exists a recursive function $F : \mathbb{N}^{k+1} \to \mathbb{Q}$ such that $|f(x) - F(x, i)| < 2^{-i}$, $\forall x \in \mathbb{N}^k$, $\forall i \in \mathbb{N}$.

A tuple (X, d, α) is said to be a **computable metric space** if (X, d) is a metric space and $\alpha : \mathbb{N} \to X$ is a sequence dense in (X, d) such that the function $\mathbb{N}^2 \to \mathbb{R}$, $(i, j) \mapsto d(\alpha(i), \alpha(j))$ is recursive. We say that α is an **effective separating sequence** in (X, d) (cf. [3]). If (X, d, α) is a computable metric space, then a sequence (x_i) in X is said to be **recursive** in (X, d, α) if there exists a recursive function $F : \mathbb{N}^2 \to \mathbb{N}$ such that $d(x_i, \alpha_{F(i,k)}) < 2^{-k}$, $\forall i, k \in \mathbb{N}$ and a point $a \in X$ is said to be **recursive** in (X, d, α) if the constant sequence $a, a, \ldots$ is recursive. For example, if $q : \mathbb{N} \to \mathbb{Q}$ is a recursive surjection, then $(\mathbb{R}, d, q)$ is a computable metric space, where d is the Euclidean metric on $\mathbb{R}$. A sequence (x_i) is recursive in this computable metric space if and only if (x_i) is a recursive sequence of real numbers and $a \in \mathbb{R}$ is a recursive point in this space if and only if a is a recursive number.

Let (X, d) be a metric space and let $\mathcal{S}$ be a nonempty set whose elements are sequences in X. We say that $\mathcal{S}$ is a **computability structure** on (X, d) (cf. [3]) if the following three properties hold:

(i) if $(x_i), (y_j) \in \mathcal{S}$, then the function $\mathbb{N}^2 \to \mathbb{R}$, $(i, j) \mapsto d(x_i, y_j)$ is recursive;
(ii) if $(x_i)_{i \in \mathbb{N}} \in \mathcal{S}$, then $(x_{f(i)})_{i \in \mathbb{N}} \in \mathcal{S}$ for any recursive function $f : \mathbb{N} \to \mathbb{N}$;
(iii) if (y_i) is a sequence in X such that $d(y_i, x_{F(i,k)}) < 2^{-k}$, $\forall i, k \in \mathbb{N}$, where $F : \mathbb{N}^2 \to \mathbb{N}$ is a recursive function and $(x_i) \in \mathcal{S}$, then $(y_i) \in \mathcal{S}$.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 161-172
http://drops.dagstuhl.de/opus/volltexte/2009/2268

Let (X,d) be a metric space. If α is an effective separating sequence in (X,d), then the set $\mathcal{S}_\alpha$ of all recursive sequences in (X,d,α) is an example of a computability structure on (X,d). Suppose now that α and β are effective separating sequences in (X,d). We say that α is **equivalent** to β, $\alpha \sim \beta$, if α is a recursive sequence in (X,d,β). It follows easily that $\alpha \sim \beta$ if and only if $\mathcal{S}_\alpha = \mathcal{S}_\beta$.

A closed subset S of a computable metric space (X,d,α) is said to be *recursively enumerable* if $\{i \in \mathbb{N} \mid I_i \cap S \neq \emptyset\}$ is an r.e. set, where (I_i) is some effective enumeration of all open rational balls in (X,d,α), *co-recursively enumerable* if $X \setminus S = \cup_{i\in\mathbb{N}} I_{f(i)}$, where $f : \mathbb{N} \to \mathbb{N}$ is a recursive function and *recursive* if it is both r.e. and co-r.e. ([1]). It is not hard to see that if $\alpha \sim \beta$, then S is r.e. (co-r.e.) in (X,d,α) if and only if S is r.e. (co-r.e.) in (X,d,β) and consequently S is recursive in (X,d,α) if and only if S is recursive in (X,d,β). Hence the notions of recursive enumerability, co-recursive enumerability and recursiveness of a set are examples of notions which depend only on the induced computability structure and not on particular α which induces that structure.

If α and β are effective separating sequences in a metric space (X,d), then α and β need not be equivalent. For example, if $c \in \mathbb{R}$ is a nonrecursive number and (α_i) a recursive sequence of real numbers dense in $(\mathbb{R},d)$, where d is the Euclidean metric, then $(\alpha_i + c)$ is an effective separating sequence in $(\mathbb{R},d)$, c is a recursive point in $(\mathbb{R},d,(\alpha_i + c))$ and c is not recursive in $(\mathbb{R},d,(\alpha_i))$. Hence (α_i) and $(\alpha_i + c)$ are not equivalent.

Let $(X,d,(\alpha_i))$ be a computable metric space and f an isometry of (X,d). By an isometry of (X,d) we mean a surjective map $f : X \to X$ such that $d(f(x),f(y)) = d(x,y)$, $\forall x,y \in X$. Then $(X,d,(f(\alpha_i)))$ is also a computable metric space and in general the sequences (α_i) and $(f(\alpha_i))$ are not equivalent by the previous example. Note that f "maps" the computability structure induced by (α_i) on the computability structure induced by $(f(\alpha_i))$, i.e.

$$\mathcal{S}_{(f(\alpha_i))} = \{(f(x_i)) \mid (x_i) \in \mathcal{S}_{(\alpha_i)}\}.$$

In particular, if A is the set of all recursive points in $(X,d,(\alpha_i))$ and B the set of all recursive points in $(X,d,(f(\alpha_i)))$, then $f(A) = B$.

We say that effective separating sequences (α_i) and (β_i) in a metric space (X,d) are **equivalent up to isometry** if $(\alpha_i) \sim (f(\beta_i))$ for some isometry f of (X,d). It is easy to see that this relation is an equivalence relation on the set of all effective separating sequences in (X,d).

If (X,d,α) is a computable metric space, then clearly the metric space (X,d) is totally bounded if and only if for each $k \in \mathbb{N}$ there exists $m \in \mathbb{N}$ such that $X = \cup_{0\leq i\leq m} B(\alpha_i, 2^{-k})$. Here $B(x,r)$ for $x \in X$ and $r > 0$ denotes the open ball of radius r centered at x. We say that a computable metric space (X,d,α) is **effectively totally bounded** if there exists a recursive function $f : \mathbb{N} \to \mathbb{N}$ such that

$$X = \bigcup_{i=0}^{f(k)} B(\alpha_i, 2^{-k}),$$

$\forall k \in \mathbb{N}$ ([3]).

Example 1. If S is a recursive nonempty compact subset of $\mathbb{R}^n$, then there exists a recursive sequence (x_i) in S and a recursive function $f : \mathbb{N} \to \mathbb{N}$ such that $S \subseteq \cup_{0 \leq i \leq f(k)} B(x_i, 2^{-k})$, $\forall k \in \mathbb{N}$ ([4]) and therefore $(S, d, (x_i))$ is an effectively totally bounded computable metric space, where d is the Euclidean metric on S.

Example 2. Let $\omega : \mathbb{N} \to \mathbb{Q}$ be a recursive sequence which converges to a nonrecursive number $\gamma \in \mathbb{R}$ and such that $\omega(0) = 0$, $\omega(i) < \omega(i+1)$, $\forall i \in \mathbb{N}$. It is easy to construct a recursive sequence of rational numbers α which is dense in $[0, \gamma]$. Then the tuple $([0, \gamma], d, \alpha)$ is a computable metric space, where d is the Euclidean metric on $[0, \gamma]$. Suppose that $([0, \gamma], d, \alpha)$ is effectively totally bounded. Then $[0, \gamma] = \cup_{0 \leq i \leq f(k)} B(\alpha_i, 2^{-k})$, $\forall k \in \mathbb{N}$, for some recursive function $f : \mathbb{N} \to \mathbb{N}$. If $h : \mathbb{N} \to \mathbb{Q}$ is defined by $h(k) = \max\{\alpha_i \mid 0 \leq i \leq f(k)\}$, $k \in \mathbb{N}$, then h is a recursive function and $|\gamma - h(k)| < 2^{-k}$, $\forall k \in \mathbb{N}$ which contradicts the fact that γ is a nonrecursive number. Hence the computable metric space $([0, \gamma], d, \alpha)$ is not effectively totally bounded, although the metric space $([0, \gamma], d)$ is totally bounded.

It is not hard to check that if α and β are equivalent effective separating sequences in a metric space (X, d), then (X, d, α) is effectively totally bounded if and only if (X, d, β) is effectively totally bounded. Furthermore, if f is an isometry of (X, d) and (α_i) an effective separating sequence, then $(X, d, (\alpha_i))$ is effectively totally bounded if and only if $(X, d, (f(\alpha_i)))$ is effectively totally bounded. This follows immediately from the fact that $f(B(x, r)) = B(f(x), r)$, $\forall x \in X$, $\forall r > 0$. Therefore, if α and β are effective separating sequences equivalent up to isometry, then (X, d, α) is effectively totally bounded if and only if (X, d, β) is effectively totally bounded.

There exist totally bounded metric spaces with effective separating sequences nonequivalent up to isometry (Section 2). Nevertheless, the equivalence

$$(X, d, \alpha) \text{ effectively totally bounded } \Leftrightarrow (X, d, \beta) \text{ effectively totally bounded} \tag{1}$$

holds in general and that is the main result of this paper which will be proved in Section 3 where we introduce the notion of effectively dispersed metric space. In Section 2 we also prove that each two effective separating sequence in Euclidean space $\mathbb{R}^n$ are equivalent up to isometry.

2 Isometries and computability structures

Let $n \geq 1$ and let d be the Euclidean metric on $\mathbb{R}^n$. The main step in proving that every two effective separating sequences in $(\mathbb{R}^n, d)$ are equivalent up to isometry is the following proposition.

Proposition 1. *Let $a_0, \ldots, a_n$ be recursive points in $\mathbb{R}^n$ which are geometrically independent (i.e. $a_1 - a_0, \ldots, a_n - a_0$ are linearly independent vectors) and let (x_i) be a sequence in $\mathbb{R}^n$ such that $(d(x_i, a_k))_{i \in \mathbb{N}}$ is a recursive sequence of real numbers for each $k \in \{0, \ldots, n\}$. Then (x_i) is a recursive sequence.*

Proposition 1 is essentially a consequence of the fact that we can compute each component of x_i by certain formula which involves addition, subtraction, multiplication and division of numbers $d(x_i, a_0), \ldots, d(x_i, a_n)$ and components of the points $a_0, \ldots, a_n$. It follows from Proposition 1 that for geometrically independent recursive points $a_0, \ldots, a_n$ in $\mathbb{R}^n$ and $x \in \mathbb{N}$ the following implication holds:

$$\text{the numbers } d(x, a_0), \ldots, d(x, a_n) \text{ are recursive } \Rightarrow \text{ the point } x \text{ is recursive.} \tag{2}$$

However, in a general computable metric space it is not possible to find $n \in \mathbf{N}$ and recursive points $a_0, \ldots, a_n$ such that the implication (2) holds. This shows the following example.

Example 3. Let p be the metric on $\mathbb{R}^2$ given by $p((x_1, y_1), (x_2, y_2)) = \max\{|x_2 - x_1|, |y_2 - y_1|\}$. If (α_i) is a recursive dense sequence in $\mathbb{R}^2$, then $(\mathbb{R}^2, p, (\alpha_i))$ is a computable metric space and the induced computability structure coincides with the usual computability structure on $\mathbb{R}^2$. Suppose $(x_0, y_0), \ldots, (x_k, y_k)$ are any recursive points in $\mathbb{R}^2$. Let $M > 0$ be some upper bound of the set $\{|x_0|, |y_0|, \ldots, |x_k|, |y_k|\}$. Let $a, b \in \mathbb{R}$ be such that $a > 3M$, $|b| < M$ and such that a is a recursive, and b a nonrecursive number. Then $p((a, b), (x_0, y_0)), \ldots$ $p((a, b), (x_k, y_k))$ are recursive numbers, but (a, b) is a nonrecursive point.

The following corollary is an immediate consequence of Proposition 1.

Corollary 1. *Suppose $(\mathbb{R}^n, d, \alpha)$ is a computable metric space, $f : \mathbb{R}^n \to \mathbb{R}^n$ an isometry and $a_0, \ldots, a_n$ recursive points in $(\mathbb{R}^n, d, \alpha)$ which are geometrically independent and such that $f(a_0), \ldots, f(a_n)$ are recursive points in $\mathbb{R}^n$ in the usual sense. Then $f \circ \alpha$ is a recursive sequence in the usual sense.*

The next step in proving that every two effective separating sequences in $(\mathbb{R}^n, d)$ are equivalent up to isometry is the following lemma.

Lemma 1. *Let $a_0, \ldots, a_n$ be geometrically independent points in $\mathbb{R}^n$ such that $d(a_i, a_j)$ is a recursive number for all $i, j \in \{0, \ldots, n\}$. Then there exists an isometry $f : \mathbb{R}^n \to \mathbb{R}^n$ such that $f(a_0), \ldots, f(a_n)$ are recursive points.*

The idea in the proof of Lemma 1 is to find an isometry $f : \mathbb{R}^n \to \mathbb{R}^n$ such that $f(a_0) = (0, \ldots, 0)$, $f(a_i) \in \{(t_1, \ldots, t_i, 0, \ldots, 0) \mid t_1, \ldots, t_i \in \mathbb{R}, t_i \neq 0\}$, $\forall i \in \{1, \ldots, n\}$ and then to show that the points $f(a_0), \ldots, f(a_n)$ are recursive.

Proposition 2. *Let (α_i) be an effective separating sequence in $\mathbb{R}^n$. Then there exists an isometry $f : \mathbb{R}^n \to \mathbb{R}^n$ such that $(f(\alpha_i))$ is a recursive sequence in $\mathbb{R}^n$.*

Proof. Let $i_0, \ldots, i_n \in \mathbb{N}$ be such that $\alpha_{i_0}, \ldots, \alpha_{i_n}$ are geometrically independent points. By Lemma 1 there exists an isometry $f : \mathbb{R}^n \to \mathbb{R}^n$ such that $f(a_{i_0}), \ldots, f(a_{i_n})$ are recursive points. The claim of the theorem now follows from Corollary 1. □

Note the following: if (x_i) and (y_i) are recursive dense sequences in $\mathbb{R}^n$, then (x_i) and (y_i) are equivalent as effective separating sequences. This and Proposition 2 imply the following.

Theorem 1. *If α and β are effective separating sequences in $(\mathbb{R}^n, d)$, then α and β are equivalent up to isometry.*

Euclidean space $\mathbb{R}^n$ is not totally bounded, but each open (or closed) ball in $\mathbb{R}^n$ is totally bounded. We say that a computable metric space (X, d, α) **can be exhausted effectively by totally bounded balls** if there exists $\tilde{x} \in X$ and a recursive function $F : \mathbb{N}^2 \to \mathbb{N}$ such that

$$B(\tilde{x}, m) \subseteq \bigcup_{i=0}^{F(k,m)} B(\alpha_i, 2^{-k}),$$

$\forall k, m \in \mathbb{N}$. It is clear that if such a function F exists for one $\tilde{x} \in X$, then it exists for each $\tilde{x} \in X$. It is obvious that each effectively totally bounded computable metric space can be exhausted effectively by totally bounded balls. Furthermore, if α is some recursive dense sequence in $\mathbb{R}^n$, then $(\mathbb{R}^n, d, \alpha)$ can be exhausted effectively by totally bounded balls. It is easy to conclude from this and Theorem 1 that any computable metric space of the form $(\mathbb{R}^n, d, \alpha)$ can be exhausted effectively by totally bounded balls.

In the contrast to the fact that the equivalence (1) holds in general, which will be proved later, the equivalence

(X, d, α) can be exhausted effectively by totally bounded balls

$$\Updownarrow$$

(X, d, β) can be exhausted effectively by totally bounded balls

does not hold in general, as the following example shows.

Example 4. Let the number γ be as in Example 2. It is easy to construct a recursive sequence of rational numbers α' which is dense in $\langle -\infty, \gamma]$. Let d be the Euclidean metric on $\langle -\infty, 0]$ and let (x_i) be some recursive sequence of real numbers which is dense in $\langle -\infty, 0]$. Then the computable metric space $(\langle -\infty, 0], d, (x_i))$ can be exhausted effectively by totally bounded balls. On the other hand, if $\alpha : \mathbb{N} \to \langle -\infty, 0]$ is defined by $\alpha(i) = \alpha'(i) - \gamma$, then α is an effective separating sequence in $(\langle -\infty, 0], d)$ and the computable metric space $(\langle -\infty, 0], d, \alpha)$ cannot be exhausted effectively by totally bounded balls which can be deduced from the fact that 0 is not a recursive point in this space.

The previous example also shows that effective separating sequences in a metric space (X, d) need not be equivalent up to isometry. The following two examples show that effective separating sequences in (X, d) need not be equivalent up to isometry even when (X, d) is totally bounded.

Example 5. Let $([0,\gamma],d,\alpha)$ be the computable metric space of Example 2. Let $\alpha' : \mathbb{N} \to \mathbb{R}$ be defined by $\alpha'(2i) = \frac{\alpha(i)}{2}$, $\alpha'(2i+1) = -\frac{\alpha(i)}{2}$, $i \in \mathbb{N}$ and let $\alpha'' : \mathbb{N} \to [0,\gamma]$ be defined by $\alpha''(i) = \alpha'(i) + \frac{\gamma}{2}$. Then α'' is an effective separating sequence in $([0,\gamma],d)$. Since the point $\frac{\gamma}{2}$ is recursive in $([0,\gamma],d,\alpha'')$, but not in $([0,\gamma],d,\alpha)$, and since $\frac{\gamma}{2}$ is a fixed point of each isometry of $([0,\gamma],d)$ (namely the only isometries are the identity and the map $t \mapsto \gamma - t$, $t \in [0,\gamma]$), we conclude that effective separating sequences α and α'' are not equivalent.

Example 6. Let S be the unit circle in $\mathbb{R}^2$ and let d be the Euclidean metric on S. Since S is a recursive set, there exists a recursive sequence (x_i) in S such that $(S,d,(x_i))$ is an effectively totally bounded computable metric space (Example 1). Let $f : \mathbb{R}^2 \to \mathbb{R}^2$ be a rotation with the center $(0,0)$ such that $f(1,0)$ is a nonrecursive point. Then $(f(x_i))$ is an effective separating sequence in (S,d) nonequivalent to (x_i). Let $A = \{x_i \mid i \in \mathbb{N}\} \cup \{f(x_i) \mid i \in \mathbb{N}\}$, let $T = \{(x,y) \in S \mid x \leq 0 \text{ or } (x,y) \in A\}$ and let d' be the Euclidean metric on T. Then (x_i) and (y_i) are effective separating sequences in (T,d') and it follows easily that they are not equivalent up to isometry in this metric space.

3 Effective dispersion

Let (X,d) be a metric space. A nonempty subset S of X is said to be $r-$**dense** in (X,d), where $r \in \mathbb{R}$, $r > 0$, if $X = \cup_{s \in S} B(s,r)$. Note that a set S is dense in (X,d) if and only if S is $r-$dense in (X,d) for all $r > 0$. We say that a finite sequence $x_0,\ldots,x_n$ of points in X is $r-$dense in (X,d) if the set $\{x_0,\ldots,x_n\}$ is $r-$dense in (X,d). Hence (X,d) is totally bounded if and only if for each $\varepsilon > 0$ there exists a finite sequence of points in X which is $\varepsilon-$dense in (X,d).

Let $s \in \mathbb{R}$. A nonempty subset S of X is said to be $s-$**dispersed** in (X,d) if $d(x,y) > s$, $\forall x,y \in S$, $x \neq y$. A finite sequence $x_0,\ldots,x_n$ of points in X is said to be $s-$**dispersed** in (X,d) if $d(x_i,x_j) > s$, $\forall i,j \in \{0,\ldots,n\}$, $i \neq j$. Note that if $x_0,\ldots,x_n$ is an s-dispersed finite sequence, then $\{x_0,\ldots,x_n\}$ is an $s-$dispersed set, while converse does not hold in general.

Proposition 3. *Let (X,d) be a totally bounded metric space and let $s > 0$. Then the set $A = \{k \in \mathbb{N} \mid$ there exists a finite sequence $x_1,\ldots,x_k$ which is $s-$dispersed in $(X,d)\}$ is finite.*

Proof. Let $y_0,\ldots,y_p$ be an $\frac{s}{2}-$dense finite sequence in (X,d). Suppose that a finite sequence $x_1,\ldots,x_k$ is $s-$dispersed. For each $i \in \{1,\ldots,k\}$ let $j_i \in \{0,\ldots,p\}$ be such that $x_i \in B(y_{j_i},\frac{s}{2})$. If $i,i' \in \{1,\ldots,k\}$, $i \neq i'$, then $j_i \neq j_{i'}$ since $d(x_i,x_{i'}) > s$. Therefore we have an injection $\{1,\ldots,k\} \to \{0,\ldots,p\}$, hence $k < p$. This shows that A is finite. □

Let (X,d) be a totally bounded metric space. If $S \subseteq X$, $S \neq \emptyset$, and $s > 0$, then, by Proposition 3, the set $\{k \in \mathbb{N} \mid$ there exists a finite sequence $x_1,\ldots,x_k$ of points in S which is $s-$dispersed in $(X,d)\}$ is finite. We denote the maximum of this set by $\rho(S,s)$. If $x_0,\ldots,x_n$ is a finite sequence in X, then we will write $\rho(x_0,\ldots,x_n;s)$ instead of $\rho(\{x_0,\ldots,x_n\},s)$.

Example 7. With the Euclidean metric on $[0,3]$ we have $\rho([0,1],s)=1$ if $s\geq 1$, $\rho([0,1],s)=2$ if $s\in\left[\frac{1}{2},1\right\rangle$ and $\rho(0,1,3;s)=\begin{cases}1,\ 3\leq s,\\ 2,\ 1\leq s<3,\\ 3,\ 0<s<1.\end{cases}$

Suppose (X,d) is a totally bounded metric space, $s>0$ and $n=\rho(X,\frac{s}{2})$. Then there exists a finite sequence $x_0,\ldots,x_{n-1}$ which is $\frac{s}{2}-$dispersed in (X,d) and such that the finite sequence $a,x_0,\ldots,x_{n-1}$ is not $\frac{s}{2}-$dispersed for each $a\in X$. Therefore for each $a\in X$ there exists $i\in\{0,\ldots,n-1\}$ such that $d(a,x_i)<s$. Hence the finite sequence $x_0,\ldots,x_{n-1}$ is $s-$dense.

Now, let α and β be effective separating sequences in (X,d) such that the computable metric space (X,d,α) is effectively totally bounded. In order to prove that (X,d,β) is also effectively totally bounded, it would be enough to prove that for each $k\in\mathbb{N}$ we can effectively find the number $\rho(X,2^{-k})$. Namely, in that case for any $k\in\mathbb{N}$ we can effectively find $i_1,\ldots,i_n\in\mathbb{N}$ such that the finite sequence $\beta_{i_1},\ldots,\beta_{i_n}$ is $2^{-(k+1)}-$dispersed, where $n=\rho(X,2^{-(k+1)})$ and then this finite sequence of points (and consequently the finite sequence $\beta_0,\ldots,\beta_{\max\{i_1,\ldots,i_n\}}$) must be $2^{-k}-$dense. However, the number $\rho(X,2^{-k})$ cannot be found effectively in general, as the following example shows.

Example 8. Let (λ_i) be a recursive sequence of real numbers such that $\lambda_i\geq 0$, $\forall i\in\mathbb{N}$ and such that the set $\{i\in\mathbb{N}\mid\lambda_i=0\}$ is not recursive ([2]). We may assume $\lambda_i<4^{-i}$, $\forall i\in\mathbb{N}$. Let $t_i=4^{-i}+\lambda_i$, $i\in\mathbb{N}$, $X=\{t_i\mid i\in\mathbb{N}\}\cup\{0\}$ and let d be the Euclidean metric on X. Then $(X,d,(t_i))$ is an effectively totally bounded computable metric space. Let $i\in\mathbb{N}$. It is straightforward to check that $\rho(X,4^{-i})=i+1$ if $\lambda_i=0$ and $\rho(X,4^{-i})=i+2$ if $\lambda_i>0$. Therefore the function $\mathbb{N}\to\mathbb{N}$, $i\mapsto\rho(X,2^{-i})$ is not recursive.

Although $\rho(X,2^{-k})$ cannot be found effectively in general, we are going to prove that for $k\in\mathbb{N}$ we can effectively find numbers $a_k\in\langle 0,2^{-k}\rangle\cap\mathbb{Q}$ and $\rho(X,a_k)$ and this will imply that (X,d,β) if effectively totally bounded.

Suppose (X,d) is a totally bounded metric space, $S\subseteq X$, $S\neq\emptyset$ and $s>0$. It is immediate from the definition of the number $\rho(S,s)$ that there exists $r>0$ such that $\rho(S,s)=\rho(S,s+2r)$. Here, of course, r depends on S and s. In the following lemma we claim that s and r can be chosen so that $\rho(S,s)=\rho(S,s+2r)$ holds whenever S is in certain family of subsets of X.

Lemma 2. *Let (X,d) be a totally bounded metric space and let $s_0>0$. Then there exists $r_0>0$ such that for each $r\in\langle 0,r_0\rangle$ and each finite sequence $x_0,\ldots,x_p$ which is $r-$dense in (X,d) there exists $s\in[s_0,s_0+r\rangle\cap\mathbb{Q}$ and $m_1,\ldots,m_n\in\{0,\ldots,p\}$ such that the finite sequence $x_{m_1},\ldots,x_{m_n}$ is $(s+2r)-$dispersed, $d(x_i,x_j)\neq s$, $\forall i,j\in\{0,\ldots,p\}$ and $\rho(x_0,\ldots,x_p;s)=n$.*

Proof. Let $n=\rho(X,s_0)$ and $y_1,\ldots,y_n$ be a finite sequence which is s_0-dispersed in (X,d). Since $d(y_i,y_j)>s_0$, $\forall i,j\in\{1,\ldots,n\}$, $i\neq j$, there exists $r_0>0$ such that

$$d(y_i,y_j)>s_0+5r_0,\ \forall i,j\in\{1,\ldots,n\},\ i\neq j.$$

Let $r \in \langle 0, r_0\rangle$ and let $x_0, \ldots, x_p$ be an $r-$dense sequence in (X,d). For $i \in \{1,\ldots,n\}$ let $m_i \in \{0,\ldots,p\}$ be such that $y_i \in B(x_{m_i}, r)$. If $i,j \in \{1,\ldots,n\}$, $i \neq j$, then

$$d(y_i,y_j) \leq d(y_i,x_{m_i}) + d(x_{m_i},x_{m_j}) + d(x_{m_j},y_j) < 2r + d(x_{m_i},x_{m_j})$$

and therefore

$$d(x_{m_i},x_{m_j}) > s_0 + 3r, \tag{3}$$

$\forall i,j \in \{1,\ldots,n\}$, $i \neq j$. Let $s \in [s_0, s_0+r\rangle \cap \mathbb{Q}$ be such that $d(x_i,x_j) \neq s$, $\forall i,j \in \{0,\ldots,p\}$. From (3) we get that $d(x_{m_i},x_{m_j}) > s+2r$, $\forall i,j \in \{1,\ldots,n\}$, $i \neq j$, hence the finite sequence $x_{m_1},\ldots,x_{m_n}$ is $s+2r-$dispersed. This implies $\rho(x_0,\ldots,x_p;s) \geq n$. On the other hand

$$\rho(x_0,\ldots,x_p;s) \leq \rho(X,s) \leq \rho(X,s_0) = n.$$

Therefore $\rho(x_0,\ldots,x_p;s) = n$. □

The next lemma provides conditions under which equality $\rho(X,s+2r) = \mathrm{card}(T)$ holds, where $T \subseteq X$ and $s,r > 0$.

Lemma 3. *Let (X,d) be a totally bounded metric space, $r,s > 0$ and let S be an $r-$dense subset of X such that there exists a finite nonempty subset T of S which is $s+2r$ dispersed and such that $\rho(S,s) = \mathrm{card}(T)$. Then $\rho(X,s+2r) = \mathrm{card}(T)$.*

Proof. Certainly $\rho(X,s+2r) \geq \mathrm{card}(T)$. On the other hand, let $x_1,\ldots,x_n$ be an $(s+2r)-$dispersed sequence in (X,d). For each $i \in \{1,\ldots,n\}$ let $y_i \in S$ be such that $d(x_i,y_i) < r$. For all $i,j \in \{1,\ldots,n\}$, $i \neq j$, we have

$$s+2r < d(x_i,x_j) \leq d(x_i,y_i) + d(y_i,y_j) + d(y_j,x_j) < d(y_i,y_j) + 2r$$

which implies $s < d(y_i,y_j)$. Hence $y_1,\ldots,y_n$ is an $s-$dispersed sequence and therefore $\rho(S,s) \geq n$, i.e. $\mathrm{card}(T) \geq n$. We conclude that $\mathrm{card}(T) \geq \rho(X,s+2r)$ and it follows $\rho(X,s+2r) = \mathrm{card}(T)$. □

Lemma 3, together with Lemma 2, gives the idea how to compute the number $\rho(X,s+2r)$, $s,r > 0$. The next step is to include effectiveness into consideration. We first state the following lemma.

Lemma 4. *Let $F : \mathbb{N}^4 \to \mathbb{R}$ be a recursive function. Let S be the set of all $(k,n,l,p) \in \mathbb{N}^4$ such that $F(i,j,n,l) \neq 0$, $\forall i,j \in \{0,\ldots,k\}$ and such that*

$$\mathrm{card}\{(i,j) \in \{0,\ldots,k\} \times \{0,\ldots,k\} \mid F(i,j,n,l) > 0\} = p.$$

Then S is a recursively enumerable set.

Let $\sigma : \mathbb{N}^2 \to \mathbb{N}$ and $\eta : \mathbb{N} \to \mathbb{N}$ be some fixed recursive functions with the following property: $\{(\sigma(j,0),\ldots,\sigma(j,\eta(j))) \mid j \in \mathbb{N}\}$ is the set of all finite sequences in $\mathbb{N}$, i.e. the set $\{(a_0,\ldots,a_n) \mid n \in \mathbb{N},\ a_0,\ldots,a_n \in \mathbb{N}\}$. Such functions, for instance, can be defined using the Cantor pairing function. We are

going to use the following notation: $(j)_i$ instead of $\sigma(j,i)$ and $\overline{j}$ instead of $\eta(j)$. Hence

$$\{((j)_0,\ldots,(j)_{\overline{j}}) \mid j \in \mathbb{N}\}$$

is the set of all finite sequences in $\mathbb{N}$.

Suppose (X,d) is a metric space, (γ_i) a sequence in X such that the function $\mathbb{N}^2 \to \mathbb{R}$, $(i,j) \mapsto d(\gamma_i,\gamma_j)$ is recursive and (s_n) a recursive sequence of real numbers. Then the function $\mathbb{N}^2 \to \mathbb{N}$, $(k,n) \mapsto \rho(\gamma_0,\ldots,\gamma_k;s_n)$ need not be recursive and we see this similarly as in Example 8. However, we have the following lemma.

Lemma 5. *Let (X,d) be a metric space, (γ_i) a sequence in X such that the function $\mathbb{N}^2 \to \mathbb{R}$, $(i,j) \mapsto d(\gamma_i,\gamma_j)$ is recursive and (s_n) a recursive sequence of real numbers.*

(i) The set

$$S = \{(k,n,p) \in \mathbb{N}^3 \mid d(\gamma_i,\gamma_j) \neq s_n,\ \forall i,j \in \{0,\ldots,k\},\ \rho(\gamma_0,\ldots,\gamma_k;s_n) = p\}$$

is recursively enumerable.

(ii) The set

$$T = \{(l,n) \in \mathbb{N}^2 \mid \textit{the finite sequence } \gamma_{(l)_0},\ldots,\gamma_{(l)_{\overline{l}}} \textit{ is } s_n - \textit{dispersed}\}$$

is recursively enumerable.

Proof. (i) We apply Lemma 4 to the function $F : \mathbb{N}^4 \to \mathbb{R}$ defined by

$$F(i,j,n,l) = d(\gamma_i,\gamma_j) - s_n,$$

$i,j,n,l \in \mathbb{N}$ and we get that the set

$$\{(k,n,l,p) \in \mathbb{N}^4 \mid d(\gamma_i,\gamma_j) \neq s_n,\ \forall i,j \in \{0,\ldots,k\},\ \rho(\gamma_0,\ldots,\gamma_k;s_n) = p\}$$

is r.e. which implies that S is r.e.

(ii) Let $F : \mathbb{N}^4 \to \mathbb{R}$ be given by $F(i,j,n,l) = d(\gamma_{(l)_i},\gamma_{(l)_j}) - s_n$. Let T' be the set associated to F as in Lemma 4, hence $T' = \{(k,n,l,p) \in \mathbb{N}^4 \mid d(\gamma_{(l)_i},\gamma_{(l)_j}) \neq s_n,\ \forall i,j \in \{0,\ldots,k\}$ and $\rho(\gamma_{(l)_0},\ldots,\gamma_{(l)_k};s_n) = p\}$. Then for all $l,n \in \mathbb{N}$ we have $(l,n) \in T$ if and only if

$$d(\gamma_{(l)_i},\gamma_{(l)_j}) \neq s_n, \forall i,j \in \{0,\ldots,\bar{l}\},\ \rho(\gamma_{(l)_0},\ldots,\gamma_{(l)_{\overline{l}}};s_n) = \bar{l}+1$$

and this holds if and only if $(\bar{l},n,l,\bar{l}+1) \in T'$. Therefore T is r.e. □

A totally bounded metric space (X,d) is said to be **effectively dispersed** if there exists a recursive function $a : \mathbb{N} \to \mathbb{Q}$ such that $a(i) \in \langle 0, 2^{-i}\rangle$, $\forall i \in \mathbb{N}$ and such that the function $\mathbb{N} \to \mathbb{N}$, $i \mapsto \rho(X,a(i))$ is recursive.

Theorem 2. *Let (X,d,α) be an effectively totally bounded computable metric space. Then (X,d) is effectively dispersed.*

Proof (Sketch). Let $f : \mathbb{N} \to \mathbb{N}$ be a recursive function such that $\alpha_0, \ldots, \alpha_{f(k)}$ is a $2^{-k}-$dense sequence for each $k \in \mathbb{N}$. Let $q : \mathbb{N} \to \mathbb{Q}$ be some fixed recursive function whose image is $\mathbb{Q} \cap \langle 0, \infty \rangle$.

Let $i \in \mathbb{N}$. Let s_0 be some positive number such that $s_0 < 2^{-i}$. By Lemma 2 there exist $k, n, l \in \mathbb{N}$ such that $s_0 + 3 \cdot 2^{-k} < 2^{-i}$, $q_n \in [s_0, s_0 + 2^{-k}\rangle$ and such that the following holds:

$$\alpha_{(l)_0}, \ldots, \alpha_{(l)_{\bar{l}}} \text{ is } (q_n + 2 \cdot 2^{-k}) - \text{dispersed finite sequence}, \tag{4}$$

$$d(\alpha_i, \alpha_j) \neq q_n, \forall i, j \in \{0, \ldots, f(k)\},\ \rho(\alpha_0, \ldots, \alpha_{f(k)}; q_n) = \bar{l} + 1, \tag{5}$$

and

$$\{(l)_0, \ldots, (l)_{\bar{l}}\} \subseteq \{0, \ldots, f(k)\}. \tag{6}$$

Since (4) and (5) are r.e. relations (Lemma 5) and (6) is recursive, we can express n, k and l recursively in i. The claim of the theorem follows from

$$\rho(X, q_n + 2 \cdot 2^{-k}) = \bar{l} + 1,$$

and this equality can be deduced from Lemma 3. □

Theorem 3. *Let (X, d, α) be a computable metric space such that (X, d) is effectively dispersed. Then (X, d, α) is effectively totally bounded.*

The idea in the proof of Theorem 3 is that for a given $i \in \mathbb{N}$ we effectively find $i_1, \ldots, i_n \in \mathbb{N}$ such that the finite sequence $\alpha_{i_1}, \ldots, \alpha_{i_n}$ is $s-$dispersed, where $s \in \langle 0, 2^{-(i+1)}\rangle \cap \mathbb{Q}$ and $n = \rho(X, s)$. Then the finite sequence of points $\alpha_{i_1}, \ldots, \alpha_{i_n}$ must be $2^{-i}-$dense which shows that (X, d, α) is effectively totally bounded.

Let (X, d, α) be a computable metric space. Theorem 2 and Theorem 3 give the following equivalence:

$$(X, d, \alpha) \text{ is effectively totally bounded } \Leftrightarrow (X, d) \text{ is effectively dispersed .}$$

Corollary 2. *Let α and β be effective separating sequences in a metric space (X, d). Then (X, d, α) is effectively totally bounded if and only if (X, d, β) is effectively totally bounded.*

A computable metric space (X, d, α) is said to be **effectively compact** if (X, d, α) is effectively totally bounded and (X, d) is compact (cf. [3]). If α and β are effective separating sequences in a metric space (X, d), then, by Corollary 2, (X, d, α) is effectively compact if and only if (X, d, β) is effectively compact.

Acknowledgements

The author would like to thank the anonymous referee for his valuable suggestions and corrections. Furthermore, the author is grateful to Professor Ivan Ivanšić for his comments.

References

1. Brattka, V., Presser, G.: Computability on subsets of metric spaces. Theoretical Computer Science 305 (2003) 43-76.
2. Pour-El, M.B., Richards, I.: Computability in Analysis and Physics. Springer-Verlag, Berlin-Heielberg-New York, 1989.
3. Yasugi, M., Mori, T., Tsujji, Y.: Effective properties of sets and functions in metric spaces with computability structure. Theoretical Computer Science 219 (1999) 467-486.
4. Zhou, Q.: Computable Real-Valued Functions on Recursive Open and Closed Subsets of Euclidean Space. Math. Log. Quart. 42 (1996) 379-409.

On Oscillation-free ε-random Sequences II

Jöran Mielke and Ludwig Staiger

Institute for Computer Science
Martin-Luther-University
Halle(Saale), Germany

Abstract. It has been shown (see [10]), that there are strongly Martin-Löf-ε-random ω-words that behave in terms of complexity like random ω-words. That is, in particular, the *a priori* complexity of these ε-random ω-words is bounded from below and above by linear functions with the same slope ε. In this paper we will study the set of these ω-words in terms of Hausdorff measure and dimension.
Additionally we find upper bounds on *a priori* complexity, monotone and simple complexity for a certain class of ω-power languages.

1 Introduction

The present paper is a continuation of [10] where it has been shown that oscillation-free ε-random sequences exist, for every computable ε, $0 < \varepsilon < 1$. To this end two methods were developed. The first one, by diluting random sequences, led to a method for a general "construction" of ε-random sequences from random sequences whereas the second one exhibited ε-random sequences as maximally complex sequences in certain computably definable sets of sequences (ω-languages).

Here we address mainly two questions. The first one is about the Hausdorff dimension and the Hausdorff measure of the set of oscillation-free ε-random sequences. As every random sequence is also ε-random the set of ε-random sequences has Hausdorff dimension 1. We prove a result analogous to Ryabko's estimate of the dimension of the set of sequences of a certain asymptotic relative complexity (cf. [6, 9]). We show that the set of oscillation-free ε-random sequences has Hausdorff dimension ε and infinite ε-dimensional Hausdorff measure.

The second problem we address is the one of obtaining oscillation-free ε-random sequences in so-called ω-power languages. Here we generalise the results for ω-powers of regular languages obtained in [10] to more general classes of ω-powers of computably enumerable languages.

2 Notation and Preliminary Results

In this section we briefly recall the concepts of Hausdorff measure and complexity of finite and infinite words used in this paper. For more detailed information the reader is referred to the textbooks [2] and [4]. In the following X is

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 173-184
http://drops.dagstuhl.de/opus/volltexte/2009/2269

a finite alphabet with cardinality $|X| = r$. By X^* we denote the set (monoid) of words on X, including the *empty word* e, and X^ω is the set of infinite words (ω-words) over X. For $w \in X^*$ and $\eta \in X^* \cup X^\omega$ let $w \cdot \eta$ be their concatenation. We extend this concatenation in the obvious way to subsets $W \subseteq X^*$ and $B \subseteq X^* \cup X^\omega$. For a language W let $W^* := \bigcup_{n\in\mathbb{N}} W^n$ be the *submonoid* of X^* generated by W, and by $W^\omega := \{w_1 \cdots w_n \cdots \mid w_n \in W\setminus\{e\}\}$ we denote the subset of X^ω formed by concatenating words of W. We call $V/w := \{v \mid w \cdot v \in V\}$ the *left derivative* of V by w. Furthermore $|w|$ is the *length* of the word $w \in X^*$ and $\underline{l}(V) := \min\{|v| \mid v \in V\}$ denotes the length of the shortest word contained in V. For a set $B \subseteq X^* \cup X^\omega$ the set of all finite prefixes of strings in B is $\mathbf{pref}(B)$, we abbreviate $w \in \mathbf{pref}(\{\eta\})$ by $w \sqsubseteq \eta$. By $\xi[0..n]$ we denote the prefix of $\xi \in X^* \cup X^\omega$ of length n.

A real number α is right-computable (left-computable) if and only if there is a computable sequence α_i, $i \in \mathbb{N}$, of rational numbers with $\alpha_i \geq \alpha_{i+1}$ ($\alpha_i \leq \alpha_{i+1}$) for all $i \in \mathbb{N}$ and $\lim_{i\to\infty} \alpha_i = \alpha$. A number α is called computable if and only if α is left- and right-computable. A function $f : X^* \to \mathbb{R}$ is called right-computable (left-computable) if and only if there is a computable function $h : X^* \times \mathbb{N} \to \mathbb{R}$ such that $\lim_{t\to\infty} h(w,t) = f(w)$, for every $w \in X^*$, and h is decreasing (increasing) with respect to t.

A language $V \subseteq X^*$ is called a *code* provided every $w \in V^*$ has a unique factorisation $w = v_1 \dots v_n$ with $v_i \in V$ ($1 \leq i \leq n$). If $e \notin V$ and for all $v, w \in V$ the relation $v \sqsubseteq w$ implies $v = w$ then V is called *prefix code*.

It is useful to consider the set X^ω as a metric space (Cantor space) (X^ω, ρ) of all ω-words over the alphabet X where the metric is ρ is defined as follows

$$\rho(\xi,\eta) := \inf\{r^{-|w|} \mid w \sqsubseteq \xi \wedge w \sqsubseteq \eta\}$$

The open (and simultaneously closed) balls in (X^ω, ρ) are the sets of the form $w \cdot X^\omega$, where $w \in X^*$. The diameter of these balls is $d(w \cdot X^\omega) = r^{-|w|}$. The closure of a set $F \subseteq X^\omega$ in (X^ω, ρ) is the set $\mathcal{C}(F) := \{\xi \mid \mathbf{pref}(\xi) \subseteq \mathbf{pref}(F)\}$.

We define Hausdorff measure and Hausdorff dimension for subsets of (X^ω, ρ). For every language $F \subseteq X^\omega$ and every $0 \leq \varepsilon \leq 1$ the equation

$$\mathbb{L}_\varepsilon(F) := \lim_{n\to\infty} \inf\left\{\sum_{v\in V} r^{-\varepsilon\cdot|v|} \mid F \subseteq V \cdot X^\omega \wedge \underline{l}(V) \geq n\right\}$$

defines the ε-dimensional Hausdorff measure of F. The measure $\mathbb{L}_1$ is the usual Lebesgue measure. The following property of the Hausdorff measure is well-known.

Corollary 1. *Let $F \subseteq X^\omega$. If $\mathbb{L}_\varepsilon(F) < \infty$ then for every $\delta > 0$ it holds $\mathbb{L}_{\varepsilon+\delta}(F) = 0$ and if $0 < \mathbb{L}_\varepsilon(F)$ then for every $\delta > 0$ it holds $\mathbb{L}_{\varepsilon-\delta}(F) = \infty$.*

The Hausdorff dimension of F is defined as follows

$$\dim F = \sup\{\varepsilon \mid \mathbb{L}_\varepsilon(F) = \infty \vee \varepsilon = 0\} = \inf\{\varepsilon \mid \mathbb{L}_\varepsilon(F) = 0\}$$

Next we introduce the complexities used in this paper. Consider a semi-measure m on X^*, that is, a function $m : X^* \to \mathbb{R}$ which satisfies $m(\varepsilon) \leq 1$ and

$m(w) \geq \sum_{x \in X} m(wx)$, for $w \in X^*$. If $m(w) = \sum_{x \in X} m(wx)$ the function m is called a measure. In [13] Levin proved the existence of a universal left-computable semi-measure $\mathbf{M}$, that is, a left-computable semi-measure which satisfies

$$\exists c_m \, \forall w \in X^* \qquad m(w) \leq c_m \cdot \mathbf{M}(w), \tag{1}$$

for all left-computable semi-measures m. Then the *a priori* complexity is defined as $\mathrm{KA}(w) = \lfloor -\log_r \mathbf{M}(w) \rfloor$ (cf. [4, 11]).

For the definition of the *monotone complexity* Km we refer the reader to [4, 12]. Here we need only the following property.

Corollary 2 ([4]). *Let μ be a computable measure on X^*. Then there is a constant c_μ such that*

$$\mathrm{Km}(w) \leq -\log \mu(w) + c_\mu$$

holds for all $w \in X^*$.

Plain (cf. [4]) or simple (cf. [11]) program size complexity defines the complexity of a finite string to be the length of a shortest program which prints the string. Let $\varphi : X^* \to X^*$ be a partial computable function. The complexity of a word $w \in X^*$ with respect to φ is defined as

$$\mathrm{K}_\varphi(w) := \inf\{|\pi| \mid \pi \in X^* \wedge \varphi(\pi) = w\}\,. \tag{2}$$

It is well-known that there is an optimal partial computable function $\mathfrak{U}$, that is, a function satisfying

$$\exists c_\varphi \forall w (w \in X^* \to \mathrm{K}_\mathfrak{U}(w) \leq \mathrm{K}_\varphi(w) + c_\varphi) \tag{3}$$

for every partial computable function φ. In the sequel we fix an optimal function $\mathfrak{U}$ and denote the complexity with respect to this function by KS.

The complexity of an infinite word ξ is a function mapping natural numbers n to the complexity of the n-length prefix of ξ.

Definition 1. *Let $\xi \in X^\omega$.*

1. *The function* $\mathrm{KS}(\xi[\cdot]) : \mathbb{N} \to \mathbb{N}$ *is called* plain *or* simple complexity *of ξ.*
2. *The function* $\mathrm{Km}(\xi[\cdot]) : \mathbb{N} \to \mathbb{N}$ *is called* monotone complexity *of ξ.*
3. *The function* $\mathrm{KA}(\xi[\cdot]) : \mathbb{N} \to \mathbb{N}$ *is called* a priori complexity *of ξ.*

We follow here, except for the monotone complexity, the notation of Uspensky and Shen in [11]. In [1] strongly Martin-Löf-ε-random ω-words were introduced as follows.

Definition 2. *A computably enumerable set $\mathcal{V} \subseteq X^* \times \mathbb{N}$ is referred to as a strong* Martin-Löf-ε*-test provided*

1. $\forall i (V_{i+1} \cdot X^\omega \subseteq V_i \cdot X^\omega)$, *where* $V_i := \{v \mid (v, i) \in \mathcal{V}\}$ *and*
2. $\forall i \, \forall C (C \subseteq V_i \wedge C$ *is prefix code* $\to \sum_{v \in C} r^{-\varepsilon \cdot |v|} < r^{-i})$.

An ω-word $\xi \in X^\omega$ is called strongly Martin-Löf-ε-random *if and only if $\xi \notin \bigcap_{i \in \mathbb{N}} V_i \cdot X^\omega$ for all strong* Martin-Löf-ε*-tests.*

We mention the following equivalence between strong Martin-Löf-ε-tests and *a priori* complexity.

Lemma 1 ([1]). *Let $0 < \varepsilon \leq 1$ be a computable number. Then an ω-word $\xi \in X^\omega$ is strongly* Martin-Löf-ε*-random if and only if* $\mathrm{KA}(\xi[0..n]) \geq_{\text{a.e.}} \varepsilon \cdot n - O(1)$.

Ryabko showed in [6] the following result on the set of ω-words having a bounded asymptotic lower complexity (see also [7]).

Theorem 1 ([6]).

$$\dim \left\{ \xi \mid \xi \in X^\omega \wedge \liminf_{n\to\infty} \frac{\mathrm{KA}(\xi[0..n])}{n} \leq \varepsilon \right\} = \varepsilon$$

Depending on the ε-dimensional measure of an ω-language we obtain a lower bound on the complexity of the most complex ω-words in that ω-language.

Theorem 2 ([5]). *Let $F \subseteq X^\omega$ and $\mathbb{L}_\varepsilon(F) > 0$. Then for all $c > -\log \mathbb{L}_\varepsilon(F)$ there is a $\xi_c \in F$ such that* $\mathrm{KA}(\xi_c[0..n]) \geq_{\text{a.e.}} \varepsilon \cdot n - c$.

ω-words which, analogously to random ω-words, satisfy also a linear upper bound for *a priori* complexity are referred to as oscillation-free.

Definition 3 ([10]). *An ω-word ξ is called* oscillation-free strongly Martin-Löf-ε-random *if and only if ξ is strongly* Martin-Löf-ε*-random and there is a constant c such that* $\mathrm{KA}(\xi[0..n]) \leq \varepsilon \cdot n + c$ *holds.*

3 The Measure of the Set of ε-random Sequences

We start with mappings that preserve some properties of the measure of a language and the behaviour of the complexity-function of an ω-word.

Definition 4. *A computable function $\varphi : X^* \to X^*$ is called* dilution function *provided φ is prefix-monotone, one-to-one and $|\varphi(w)| = |\varphi(w')|$ for all $w, w' \in X^n$. A function $g : \mathbb{N} \to \mathbb{N}$ is called* modulus-function *for φ provided $|\varphi(w)| = g(|w|)$ for every $w \in X^*$.*

Every dilution function φ defines a mapping $\overline{\varphi} : X^\omega \to X^\omega$ in the following way: $\mathbf{pref}(\overline{\varphi}(\xi)) = \mathbf{pref}(\varphi(\mathbf{pref}(\xi)))$. The following is an example of a dilution function.

Example 1. Dilution functions can be defined inductively by inserting a fixed string in front of every letter. Let $X = \{0, 1\}$. Then $\varphi(e) := e$ and $\varphi(wx) := \varphi(w)0x$ for every $w \in X^*$ and $x \in X$ defines a dilution function with $\frac{1}{2}$-modulus.

In this paper we are interested in the following dilution functions.

Definition 5. *Let ε with $0 < \varepsilon < 1$ be a computable real. A computable function g is called* ε-modulus *if and only if there is a constant c such that $|\varepsilon \cdot g(n) - n| \leq c$, for all $n \in \mathbb{N}$.*

The mapping $g(n) := \lceil \frac{n}{\varepsilon} \rceil$ is an example for an ε-modulus. If φ is a dilution function with ε-modulus then for every $w \in X^*$ holds

$$-c \leq \varepsilon \cdot |\varphi(w)| - |w| \leq c\,.$$

We obtain our first result on the relation of the measure of a language and its image.

Lemma 2. *Let $F \subseteq X^\omega$, $0 < \varepsilon < 1$ computable and $\varphi : X^* \to X^*$ a dilution function with ε-modulus $g : \mathbb{N} \to \mathbb{N}$. There are constants $c_1, c_2 > 0$, such that*

$$c_1 \cdot \mathbb{L}(F) \leq \mathbb{L}_\varepsilon(\overline{\varphi}(F)) \leq c_2 \cdot \mathbb{L}(F)\,.$$

Proof. The first inequality is shown as follows. Let $W \subseteq X^*$ cover $\overline{\varphi}(F)$, that is, $\overline{\varphi}(F) \subseteq W \cdot X^\omega$ and let $\underline{l}(W) \geq n$. For every $w \in W$ we define v_w as the unique word with $\varphi(v_w) \sqsubseteq w \sqsubset \varphi(v_w x)$, for some $x \in X$. Since φ has an ε-modulus, we have the following:

$$|v_w| - c \leq \varepsilon \cdot |w| \leq |v_w x| + c = |v_w| + 1 + c$$

Then the set $V = \{v_w \mid w \in W\}$ covers F. Now we obtain a bound of the ε-dimensional measure of $\overline{\varphi}(F)$ by the 1-dimensional measure of F:

$$\begin{aligned}\sum_{w \in W} r^{-\varepsilon \cdot |w|} &\geq \sum_{w \in W} r^{-|v_w|-1-c} \geq r^{-1-c} \sum_{v \in V} r^{-|v|} \\ &\geq r^{-1-c} \cdot \inf\left\{ \sum_{v \in V} r^{-|v|} \mid F \subseteq V \cdot X^\omega \wedge \underline{l}(V) \geq \varepsilon \cdot n - c - 1 \right\}\end{aligned}$$

Taking the limit $n \to \infty$ we get our intended inequality $\mathbb{L}_\varepsilon(\overline{\varphi}(F)) \geq c_1 \cdot \mathbb{L}(F)$.

To prove the second inequality we consider a set V with minimum length $\underline{l}(V) \geq n$ that covers F. Now the set $W = \{\varphi(v) \mid v \in V\}$ covers $\overline{\varphi}(F)$ and we can estimate

$$\begin{aligned}\sum_{v \in V} r^{-|v|} &\geq r^{-c} \cdot \sum_{w \in W} r^{-\varepsilon \cdot |w|} \\ &\geq r^{-c} \cdot \inf\left\{ \sum_{w \in W} r^{-\varepsilon \cdot |w|} \mid \overline{\varphi}(F) \subseteq W \cdot X^\omega \wedge \underline{l}(W) \geq g(n) - c \right\}\end{aligned}$$

Again, the limit $n \to \infty$ yields the announced inequality.

Since the constants c_1 and c_2 in Lemma 2 are positive, the following equivalence of the 1-dimensional measure of F and the ε-dimensional measure of $\overline{\varphi}(F)$ holds true.

Corollary 3. *Let $F \subseteq X^\omega$, $0 < \varepsilon < 1$ computable and $\varphi : X^* \to X^*$ a dilution function with ε-modulus $g : \mathbb{N} \to \mathbb{N}$. The measures $\mathbb{L}(F)$ and $\mathbb{L}_\varepsilon(\overline{\varphi}(F))$ are simultaneously zero, positive or infinite, respectively.*

To derive our main theorem we still need the following result from [10]. It states that the *a priori* complexity of the $\varepsilon \cdot n$-length prefix of an ω-word and the n-length prefix of its image differ not too much.

Corollary 4 ([10]). *Let ε, $0 < \varepsilon < 1$, be a computable number. Then there is a dilution function $\varphi : X^* \to X^*$ with strictly increasing ε-modulus g such that*

$$|\mathrm{KA}(\overline{\varphi}(\xi)[0..n]) - \mathrm{KA}(\xi[0..\varepsilon \cdot n])| \leq O(1) \text{ for all } \xi \in X^\omega \text{ and all } n \in \mathbb{N}.$$

If the ω-word ξ is chosen to be random then $\overline{\varphi}(\xi)$ is an oscillation-free Martin-Löf-ε-random ω-word.

As every (1-)random ω-word is also strongly ML-ε-random the Hausdorff dimension of the set of all strongly ML-ε-random ω-words is 1. Theorem 1 shows that the Hausdorff dimension of the set of all oscillation-free strongly Martin-Löf-ε-random ω-words is bounded from above by ε. The next theorem calculates its Hausdorff dimension and the corresponding measure.

Theorem 3. *Let $0 < \varepsilon < 1$ computable. The set F_ε of all oscillation-free strongly* Martin-Löf-ε*-random sequences has* Hausdorff *dimension ε and infinite ε-dimensional measure.*

Proof. Theorem 1 implies $\dim F_\varepsilon \leq \varepsilon$, since $\mathrm{KA}(\xi[0..n]) \leq \varepsilon \cdot n + c$ for every $\xi \in F_\varepsilon$. On the other hand, let φ be a dilution function with ε-modulus and F_1 the set of all (1-)random sequences. Then, according to Corollary 4, $\overline{\varphi}(F_1) \subseteq F_\varepsilon$. Since F_1 has positive, finite 1-dimensional measure, $\overline{\varphi}(F_1)$ has positive, finite ε-dimensional measure. Thus $\varepsilon = \dim \overline{\varphi}(F_1) \leq \dim F_\varepsilon$.

To show that the ε-dimensional measure of F_ε is infinite, we find an infinite family of pairwise disjoint subsets of F_ε for which the ε-dimensional measure of every set of the family is bounded from below by the same positive constant. Let $a, b \in X$, $a \neq b$ and $k : \mathbb{N} \to \mathbb{N}$. For every $w \in X^*$ and $x \in X$ the function φ_i is defined as follows: $\varphi_i(e) = e$ and

$$\varphi_i(wx) = \begin{cases} \varphi_i(w)a^{k(|w|)}x & \text{, if } |w| \neq i \\ \varphi_i(w)b^{k(|w|)}x & \text{, if } |w| = i \end{cases}$$

Here the function k is to be defined in a way that all φ_i become computable functions with ε-modulus. Since $\varepsilon < 1$, the set $K := \{i \mid k(i) > 0\}$ is infinite. Moreover for all $i, j \in K$, $i \neq j$, the sets $\overline{\varphi}_i(X^\omega)$ and $\overline{\varphi}_j(X^\omega)$ are disjoint. Lemma 2 shows that there is a constant $c > 0$ such that $\mathbb{L}_\varepsilon(\overline{\varphi}_i(F_1)) > c$ for every $i \in \mathbb{N}$. Now we obtain

$$\mathbb{L}_\varepsilon(F_\varepsilon) \geq \mathbb{L}_\varepsilon(\bigcup_{i \in K} \overline{\varphi}_i(F_1)) = \sum_{i \in K} \mathbb{L}_\varepsilon(\overline{\varphi}_i(F_1)) = \infty\,.$$

4 Complexity Bounds for ω-power Languages

In [8] for certain ω-power languages a necessary and sufficient condition to be of non-null α-dimensional Hausdorff measure was derived. In this respect, for a language $V \subseteq X^*$, the *α-residue of V derived by w*, the value $\sum_{v \in V/w} r^{-\alpha|v|}$, plays a special rôle.

Theorem 4 ([8]). *Let $V \subseteq X^*$ be a prefix code and $\sum_{v\in V} r^{-\alpha|v|} = 1$. Then $\alpha = \dim V^\omega$, and, moreover $\mathbb{I}\!\mathrm{L}_\alpha(V^\omega) > 0$ if and only if the α-residues of V are bounded from above.*

Thus in view of Theorem 2 such V^ω contain sequences ξ having a linear lower complexity bound $\alpha\cdot n - c$. It is interesting now to observe that bounding the α-residues of V from below yields a linear upper bound of slope α on the complexity of ω-words in the closure $\mathcal{C}(V^\omega)$.

Lemma 3. *Let $V \subseteq X^*$ be a computably enumerable prefix code. Let α be right-computable such that $\sum_{v\in V} r^{-\alpha\cdot|v|} = a \leq 1$ and the α-residues of V derived by $w \in \mathbf{pref}(V)$ are bounded from below. Then there is a constant c such that for every $\xi \in \mathcal{C}(V^\omega)$*

$$\mathrm{KA}(\xi[0..n]) \leq \alpha\cdot n + c\,.$$

Proof. In the same way as in the proof of Lemma 3.9 of [10] we construct a left-computable semi-measure μ such that $\mu(w) \geq c\cdot r^{-\alpha\cdot|w|}$ and use Eq. (1). We have only to ensure that the construction works also in the case $a < 1$. The construction is as follows.

$$\mu(w) = \begin{cases} 0 & ,\text{if } w \notin \mathbf{pref}(V^*) \\ \sum_{wv\in V} r^{-\alpha|wv|} & ,\text{if } w \in \mathbf{pref}(V) \\ r^{-\alpha\cdot|w|} & ,\text{if } w \in V^* \\ \mu(u)\cdot\mu(v) & ,\text{if } w = u\cdot v \text{ with } u \in V\cdot V^* \wedge v \in \mathbf{pref}(V)\setminus V \end{cases} \tag{4}$$

Since V is a prefix code, the decomposition in the last line of the construction is unique. The equation $\mu(w) = \sum_{x\in X}\mu(wx)$ for every $w \in \mathbf{pref}(V)\setminus V$ follows directly from the second case of the construction. For $w \in V$ we have the inequality

$$\sum_{x\in X}\mu(wx) = \mu(w)\cdot\sum_{x\in X}\sum_{xv\in V} r^{-\alpha|xv|} = \mu(w)\cdot\sum_{v\in V} r^{-\alpha|v|} = \mu(w)\cdot a \leq \mu(w) \tag{5}$$

The inductive construction in the last line yields the inequality in the remaining cases. To show that μ is left-computable we successively approximate the value $\mu(w)$ from below. Let V_i be the set of the first i elements in the enumeration of V and α_i the ith approximation of α from the right. We start with $\mu^{(0)}(w) := 0$ and $\mu^{(j)}(e) = 1$ for $j > 0$. Suppose that the jth approximation $\mu^{(j)}$ for all words shorter than w is already computed. If there is a $v \in V_j$ with $w = v\cdot w'$, $w' \neq e$, then $\mu^{(j)}(w) = \mu^{(j)}(v)\cdot\mu^{(j)}(w')$. Otherwise $\mu^{(j)}(w) = \sum_{v\in V_j\wedge w\sqsubseteq v} r^{-\alpha_j\cdot|v|}$.

Let $c_{\inf} := \inf\left\{\sum_{v\in V/w} r^{-\alpha\cdot|v|} \mid w \in \mathbf{pref}(V)\right\}$. Since μ is a left-computable semi-measure, the following inequality holds true.

$$\mathbf{M}(w)\cdot c_\mu \;\geq\; \mu(w) \;=\; r^{-\alpha|w|}\cdot\sum_{v\in V/w} r^{-\alpha|v|} \;\geq\; r^{-\alpha|w|}\cdot c_{\inf}$$

Taking the negative logarithm on both sides of the inequality we obtain $\mathrm{KA}(w) \leq \alpha\cdot|w| + \log\frac{c_\mu}{c_{\inf}}$ for every $w \in \mathbf{pref}(V^*)$.

The following example shows, that in Lemma 3 we cannot omit the condition that the α-residues are bounded from below. To this end we use a computable prefix code constructed in Example (6.4) of [7].

Example 2. Let X=$\{0,1\}$ and consider $W := \bigcup_{i\in\mathbb{N}} 0^{i+1}\cdot 1\cdot X^{i+1}\cdot 0^{4\cdot i+3}$. The language W is a prefix code. Its ω-power, W^ω, satisfies $\alpha = \dim W^\omega = \dim \mathcal{C}(W^\omega) = \frac{1}{3}$ and $\mathbb{L}_\alpha(W^\omega) = \mathbb{L}_\alpha(\mathcal{C}(W^\omega))$. For every $w \in \bigcup_{i\in\mathbb{N}} 0^{i+1}\cdot 1\cdot X^{i+1}$ we have $W/w = \{0^{4\cdot i+3}\}$. Thus $\sum_{v\in W/w} r^{-\alpha\cdot|v|} = r^{-\alpha\cdot(4\cdot i+3)}$ and, consequently, $\inf\{\sum_{v\in W/w} r^{-\alpha\cdot|v|} \mid w\in \mathbf{pref}(W)\} = 0$.
Now, in Eq. (6.13) and Proposition 6.15 of [7] it is shown that $\sup_{\xi\in W^\omega} \limsup_{n\to\infty} \frac{\mathrm{KA}(\xi[0..n])}{n} \geq \frac{1}{2} > \frac{1}{3} = \dim W^\omega$.

In connection with Theorem 4 our Lemma 3 yields a sufficient condition for ω-powers to contain oscillation-free α-random ω-words.

Corollary 5. *Let $V\subseteq X^*$ be a computably enumerable prefix code and α right-computable such that $\sum_{v\in V} r^{-\alpha\cdot|v|} = 1$ and the α-residues of V derived by $w\in \mathbf{pref}(V)$ are bounded from above and below. Then there is an oscillation-free ML-α-random ω-word in V^ω.*

The results of Section 3.2 of [10] show that Corollary 5 is valid for prefix codes which are regular languages. The subsequent example verifies that there are also non-regular prefix codes which satisfy the hypotheses of Corollary 5.

Example 3. Let $X=\{0,1\}$ and consider the Łukasiewicz language L defined by the identity $L = 0\cup 1\cdot L^2$. This language is a prefix code and Kuich [3] showed that $\sum_{w\in L} 2^{-|w|} = 1$. Thus the language V defined by $V = 00\cup 11\cdot V^2$ is also a prefix code and satisfies $\sum_{v\in V} 2^{-\frac{1}{2}\cdot|w|} = 1$. By induction one shows that for $v\in\mathbf{pref}(V)$ we have $V/v = w'\cdot V^k$ for suitable $k\in\mathbb{N}$ and $|w'|\leq 1$. Therefore the α-residues of V derived by $v\in\mathbf{pref}(V)$ are bounded from above and below.

For the monotone complexity Km a result similar to Lemma 3 can be obtained for a smaller class of ω-languages. We start with an auxiliary result.

Proposition 1. *1. If V is computably enumerable and $\sum_{v\in V} r^{-\alpha|v|} = 1$ then α is left-computable.*
2. If V is computably enumerable, α is right-computable and $\sum_{v\in V} r^{-\alpha|v|} = 1$ then V is computable.

Proof. The proof of part 1 is obvious. To prove part 2 we present an algorithm to decide whether a word w is in V or not.

Let V_j be the set of the first j elements in the enumeration of V and α_j the jth approximation of α from the right.

```
Input w
j := 0
   repeat
      j := j + 1
      if w ∈ V_j then accept and exit
   until r^{-α_j|w|} + Σ_{v∈V_j} r^{-α_j|v|} > 1
reject
```

If $w \notin V$ then the repeat until loop terminates as soon as $\sum_{v\in V_j} r^{-\alpha_j|v|} > 1 - r^{-\alpha_j|w|} \geq 1 - r^{-\alpha|w|}$ because $\sum_{v\in V_j} r^{-\alpha_j|v|} \to 1$ for $j \to \infty$.

Now we can prove our result on monotone complexity.

Lemma 4. *Let $V \subseteq X^*$ be a computably enumerable prefix code. If α is right-computable such that $\sum_{v\in V} r^{-\alpha\cdot|v|} = 1$ and the α-residues of V derived by $w \in \mathbf{pref}(V)$ are bounded from below then there is a constant c such that for every $\xi \in \mathcal{C}(V^\omega)$*

$$\mathrm{Km}(\xi[0..n]) \leq \alpha \cdot n + c\,.$$

Proof. Because of Proposition 1 we can assume that α is a computable real number and V is computable. We use Eq. (4) to construct μ as in the proof of Lemma 3. Since $a = 1$, equality holds in Eq. (5). Thus μ is a measure and for every $v \in V^*$ the number $\mu(v)$ is computable. Since V is a computable prefix code, for every $w \in X^*$ we can compute the unique decomposition $w = v \cdot w'$ with $v \in V^*$ and $w' \notin V \cdot X^*$. Now

$$\mu(w) = \mu(v) \cdot \left(1 - \sum_{v'\in V \wedge w \sqsubseteq vv'} r^{-\alpha|v'|}\right)$$

shows that μ is right-computable. If $w' \notin \mathbf{pref}(V)$ then the last factor is zero.

Again let $c_{\inf} := \inf\left\{\sum_{v\in V/w} r^{-\alpha\cdot|v|} \mid w \in \mathbf{pref}(V)\right\}$. In view of Corollary 2 we get the bound

$$\mathrm{Km}(w) \leq -\log\mu(w) + c_\mu \leq \alpha\cdot|w| + c_\mu - \log c_{\inf}$$

for every $w \in \mathbf{pref}(V^*)$.

5 Plain Complexity

In this section we prove results analogous to Lemma 3 for the complexity KS. First we derive a preparatory result. A similar lemma, for length-conditional plain description complexity, is known from [7, 13].

Lemma 5. *Let $W \subseteq X^*$ be computably enumerable, ε, $0 < \varepsilon < 1$, be a computable real number and let $|W \cap X^l| \leq c \cdot r^{\varepsilon\cdot l}$ for some constant $c > 0$ and all $l \in \mathbb{N}$. Then*

$$\exists C\big(C \in \mathbb{N} \wedge \forall w(w \in W \to \mathrm{KS}(w) \leq \varepsilon\cdot|w| + C)\big)$$

Proof. Let $X = \{0, 1, \ldots, r-1\}$ consist of r letters. Since ε is computable, $g(n) := \lceil \frac{n}{\varepsilon} \rceil$ is a computable function. Define a partial computable function $\varphi : X^* \to X^*$ as follows.

$$\varphi(0^k 1 v) := \text{ the } v\text{th word of length } g(|v|) - k \text{ in the enumeration of } W\,. \quad (6)$$

Here we interpret a word $v \in X^n$ as a number between 0 and $r^n - 1$.

As W has at most $r^{\varepsilon\cdot(l_o+l)}$ words of length l, this enumeration process yields $\{\varphi(0^k 1v) : v \in X^n\} \supseteq W \cap X^l$ as soon as $n \geq \varepsilon\cdot(l_0+g(n)-k) = \varepsilon\cdot g(n) - \varepsilon\cdot(k-l_0)$.

Hence, $\mathrm{KS}(w) \leq \varepsilon \cdot |w| + O(1)$ for all $w \in W$.

In order to apply Lemma 5 to languages V satisfying the conditions of Lemma 3 we show that a positive lower bound to the α-residues of V implies the upper bound $|\mathbf{pref}(V^*) \cap X^l| \leq c \cdot r^{\alpha\cdot l}$ for some constant $c > 0$ and all $l \in \mathbb{N}$.

Lemma 6. *Let $V \subseteq X^*$ be a code, $\sum_{v\in V} r^{-\alpha|v|} \leq 1$ and $\sum_{v\in V/w} r^{-\alpha|v|} \geq c' > 0$ for all $w \in \mathbf{pref}(V)$. Then $|\mathbf{pref}(V^*) \cap X^l| \leq c \cdot r^{\alpha\cdot l}$ for some constant $c > 0$ and all $l \in \mathbb{N}$.*

Proof. First observe that $w \in V^*$ if and only if $w \in V^l$ for some $l \leq |w|$. Thus $\mathbf{pref}(V^*) \cap X^l = \mathbf{pref}(V^l) \cap X^l$.

Let $a := \sum_{v\in V} r^{-\alpha|v|}$. Since V is a code, we have $a^l = \sum_{v\in V^l} r^{-\alpha|v|} = \sum_{|w|=l, w\in\mathbf{pref}(V^*)} \big(r^{-\alpha\cdot l} \cdot \sum_{v\in V^l/w} r^{-\alpha|v|}\big)$.

Now, $V^l/w \supseteq V^{l-i_w+1}/w' \supseteq (V/w' \cdot V^{l-i_w})$ where $w = v_1 \cdots v_{i_w-1} \cdot w'$ with $v_j \in V$ and $w' \in \mathbf{pref}(V)$.

Thus, $\sum_{v\in V^l/w} r^{-\alpha|v|} \geq \sum_{v\in V/w'} r^{-\alpha|v|} \cdot a^{l-i_w} \geq c' \cdot a^{l-i_w} \geq c' \cdot a^l$ and we obtain $a^l \geq r^{-\alpha\cdot l} \cdot |\mathbf{pref}(V^*) \cap X^l| \cdot c' \cdot a^l$ what proves our assertion.

Now, the fact that $\mathbf{pref}(V^*)$ is computably enumerable if only V is computably enumerable yields our result.

Lemma 7. *Let $V \subseteq X^*$ be a computably enumerable code, α be right-computable and $\sum_{v\in V} r^{-\alpha\cdot|v|} = a \leq 1$. If $\inf\left\{\sum_{v\in V/w} r^{-\alpha\cdot|v|} \mid w \in \mathbf{pref}(V)\right\} > 0$ then there is a constant c such that*

$$\mathrm{KS}(\xi[0..n]) \leq \alpha \cdot n + c \text{ for every } \xi \in \mathcal{C}(V^\omega)\,.$$

References

1. Calude, C. S. , L. Staiger and S. A. Terwijn, *On partial randomness*, Annals of Pure and Applied Logic **138** (2006), 20–30.
2. Falconer, K. J. , "Fractal Geometry: Mathematical Foundations and Applications", John Wiley & Sons, 1990.
3. Kuich W., *On the Entropy of Context-Free Languages*, Information and Control, **16** (1970), 2, 173 – 200.
4. Li, M. and P. Vitányi, "An Introduction to Kolmogorov Complexity and Its Applications", 2nd Edition, Springer Verlag, New York, 1997.
5. Mielke, J. , *Refined Bound on Kolmogorov Complexity for ω-languages*, `in:` Proceedings of the Fifth International Conference on Computability and Complexity in Analysis (CCA 2008), (V. Brattka, R. Dillhage, T. Grubba and A. Klutsch Eds.), Electronic Notes in Theoretical Computer Science, Vol. 221 (2008) 181 – 189.
6. Ryabko, B. Ya, *Coding of combinatorial sources and Hausdorff dimension*, Soviet. Math. Dokl. **30** (1984), 1, 219–222.

7. Staiger, L. , *Kolmogorov complexity and Hausdorff dimension*, Inform. and Comput. **103** (1993), 2, 159–194.
8. Staiger, L. , *Infinite Iterated Function Systems in Cantor Space and the Hausdorff Measure of ω-Power Languages*, Int. J. of Foundations of Computer Science, **16** (2005), 4, 787–802.
9. Staiger, L., *The Kolmogorov complexity of infinite words*, Theor. Comput. Sci. Vol. 383 (2007) 187 – 199.
10. Staiger, L. , *On Oscillation-free ε-random Sequences*, `in:` Proceedings of the Fifth International Conference on Computability and Complexity in Analysis (CCA 2008), (V. Brattka, R. Dillhage, T. Grubba and A. Klutsch Eds.), Electronic Notes in Theoretical Computer Science, Vol. 221 (2008) 287 – 297.
11. Uspensky, V. A. and A. Shen, *Relations Between Varieties of Kolmogorov Complexities*, Math. Systems Theory **29** (1996), 271–292.
12. Uspensky, V. A., Semenov, A. L. and A. Shen *Can an (individual) sequence of zeroes and ones be random?*, Uspekhi Mat. Nauk **45** (1990), 1, 105–162 [Russian].
13. Zvonkin, A. K. and L. A. Levin, *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** (1970), 6, 83–124.

Computability of Probability Distributions and Distribution Functions

Takakazu Mori *, Yoshiki Tsujii **, and Mariko Yasugi ***

Kyoto Sangyo University

Abstract. We define the computability of probability distributions on the real line as well as that of distribution functions. Mutual relationships between the computability notion of a probability distribution and that of the corresponding distribution function are discussed. It is carried out through attempts to effectivize some classical fundamental theorems concerning probability distributions. We then define the effective convergence of probability distributions as an effectivization of the classical vague convergence. For distribution functions, computability and effective convergence are naturally defined as real functions. A weaker effective convergence is also defined as an effectivization of pointwise convergence.

1 Introduction

In this article, we investigate computability aspects of probability distributions on the real line $\mathbb{R}$ in relation to their distribution functions. We will proceed as follows.

In Section 2, we briefly review some elementary notions of computability on the real line and some fundamentals of the classical theory of probability distributions on the real line.

In Section 3, we define the computability of probability distributions as well as that of distribution functions. Our central interest is the relation between those two computabilities. Meanwhile, we prove that the "vague sequential computability" is equivalent to the "weak sequential computability" for probability distributions.

In Section 4, we consider mutual relationships between effective convergence of probability distributions and that of distribution functions. If we restrict ourselves to the case where a probability distribution has a bounded density function, then the corresponding distribution function becomes effectively uniformly continuous, and we can prove the equivalence of the two effective convergences.

* Faculty of Science, Kyoto Sangyo University: morita@cc.kyoto-su.ac.jp. This work has been supported in part by Research Grant from KSU(2008, 339).

** Faculty of Science, Kyoto Sangyo University: tsujiiy@cc.kyoto-su.ac.jp. This work has been supported in part by Research Grant from KSU (2008, 282).

*** Kyoto Sangyo University: yasugi@cc.kyoto-su.ac.jp. This work has been supported in part by JSPS Grant-in-Aid No. 20540143.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 185-196
http://drops.dagstuhl.de/opus/volltexte/2009/2270

In the general case, we need to define notions of computability and effective convergence for bounded monotonically increasing right continuous functions. Such a function may be discontinuous at most countably many points.

Computability of the probability distribution has been treated by many authors. For example, Weihrauch ([10]) and Schröder and Simpson ([9]) have treated computability of probability distributions on the unit interval from the stand point of the representation theory. We develop a theory along the Pour-El and Richards line.

2 Preliminaries

Here, we briefly review the introductory part of the computability theory on the real line developed by Pour-El and Richards [6] as well as some basics of probability distributions on the real line. A sequence of rational numbers $\{r_n\}$ is said to be *recursive* if there exist recursive functions α, β and γ such that $r_n = (-1)^{\gamma(n)}\frac{\beta(n)}{\alpha(n)}$. A sequence of real numbers $\{x_{m,n}\}$ is said to *converge effectively* to $\{x_m\}$ if there exists a recursive function $\alpha(m,k)$ such that $n \geqslant \alpha(m.k)$ implies $|x_{m,n} - x_m| < 2^{-k}$. A sequence of real numbers $\{x_m\}$ is said to be *computable* if there exists a recursive double sequence of rational numbers, which converges effectively to $\{x_m\}$.

We adopt the definition of computability of continuous real functions by Pour-El and Richards in Chapter 0 of [6].

A sequence of (real) functions $\{f_m\}$ is said to be *computable*, if it is (i) *sequentially computable*, that is, $\{f_m(x_n)\}$ is computable for all computable sequences of reals $\{x_n\}$, and (ii) *effectively continuous*, that is, there exists a recursive function $\alpha(m,n,k)$ such that $x, y \in [-n,n]$ and $|x-y| < 2^{-\alpha(m,n,k)}$ imply $|f_m(x) - f_m(y)| < 2^{-k}$. $\alpha(m,n,k)$ is called an effective modulus of continuity of $\{f_m\}$.

A sequence of (real) functions $\{f_m\}$ is said to be *uniformly computable*, if it is (i) sequentially computable and (ii) *effectively uniformly continuous*, that is, there exists a recursive function $\alpha(m,k)$ such that $|x-y| < 2^{-\alpha(m,k)}$ implies $|f_m(x) - f_m(y)| < 2^{-k}$.

For a probability distribution μ on the real line $\mathbb{R}$, its distribution function F is defined by $F(x) = \mu((-\infty, x])$. Such a distribution function is characterized by the following three properties: (i) monotonically increasing; (ii) right continuous; (iii) $F(\infty) = \lim_{x\to\infty} F(x) = 1$ and $F(-\infty) = \lim_{x\to-\infty} F(x) = 0$. A distribution function may be discontinuous, but the set of discontinuous points is at most countable.

It is well known that the above correspondence between probability distributions and distribution functions is one to one and onto.

In the following we denote the integral with respect to a probability distribution μ, $\int_{\mathbb{R}} f(x)\mu(dx)$, with $\mu(f)$.

Let $\{\mu_n\}$ be a sequence of probability distributions on $\mathbb{R}$ and let μ be a probability distribution on $\mathbb{R}$ with corresponding distribution functions $\{F_n\}$

and F respectively. Convergence of $\{\mu_n\}$ to μ is defined to be the convergence of $\{\mu_n(f)\}$ to $\mu(f)$ for all continuous functions with compact support. This convergence is called *vague convergence* and is equivalent to each of the following convergences.

Weak convergence: $\{\mu_n(f)\}$ converges to $\mu(f)$ for all bounded continuous functions f on $\mathbb{R}$.

Convergence of distribution functions: $\{F_n(x)\}$ converges to $F(x)$ at every continuous point x of $F(x)$.

We refer the reader to [1], [3], [4] and [7] for details of fundamentals of probability theory.

Since we adopt the notion of computability of functions by Pour-El and Richards, we will plan to confine ourselves to continuous distribution functions. A sufficient condition for continuity of a distribution function is the following.

Definition 1. (Absolute continuity of probability distributions) *A probability distribution μ is said to be absolutely continuous if there exists a nonnegative integrable function $\xi(x)$ which satisfies that $\mu(A) = \int_A \xi(x)dx$ for all measurable set $A \subset \mathbb{R}$.*

The function ξ is called a density (function) of μ. We also say that the corresponding distribution function F has a density ξ.

Remark 1. If μ is absolutely continuous, then the corresponding distribution function F is continuous, and equalities $\mu([a,b]) = \mu((a,b]) = \mu([a,b)) = \mu((a,b))$ $= F(b) - F(a)$ hold.

3 Computability of probability distributions

In this section, we define the computability of probability distributions on $\mathbb{R}$ and discuss its relation to the computability of distribution function.

Let $\{f_n\}$ be a sequence of continuous functions with compact support. We say that $\{f_n\}$ is a computable sequence of functions with compact support if it is a computable sequence of functions in the sense of Pour-El and Richards and furthermore there exists a recursive function $K(n)$ such that $f_n(x) = 0$ if $|x| \geqslant K(n)$.

We obtain the following lemma.

Lemma 1. *A computable sequence of functions with compact support is uniformly computable.*

Proof Let $\{f_m\}$ be a computable sequence of functions with compact support with respect to recursive functions $\alpha(m,n,k)$ and $K(m)$.

Define $\beta(m,k) = \alpha(m, K(m), k+1)$ and assume that $|x-y| < 2^{-\beta(m,k)}$.

If both x and y are in $[-K(m), K(m)]$, then it holds that $|f_m(x) - f_m(y)| < 2^{-(k+1)}$; otherwise, one of them, say, x is in $[-K(m), K(m)]$ and the other, say, y is not in $[-K(m), K(m)]$. Then $y < -K(m) \leqslant x$ or $y > K(m) \geqslant x$ and

$|x + K(m)| < 2^{-\beta(m,k)}$ or $|x - K(m)| < 2^{-\beta(m,k)}$ accordingly. So, $|f_m(x) - f_m(y)| = |f_m(x) - f_m(\pm K(m))| < 2^{-k}$, since $f_m(y) = f_m(\pm K(m)) = 0$. This proves that $\{f_m\}$ is effectively uniformly continuous with respect to $\beta(m,k)$. □

Definition 2. (Computability of probability distributions) *We say that a sequence of probability distributions $\{\mu_m\}$ is computable if it satisfies the following* vague sequential computability: *$\{\mu_m(f_n)\}$ is computable for all computable sequence of functions with compact support $\{f_n\}$.*

Remark 2. If we regard the integral $\mu(f)$ as a function on the set of all bounded continuous functions $\mathcal{C}_b(\mathbb{R})$ with sup-norm $||\ ||$, Definition 2 only asserts sequential computability. For a probability distribution μ, it holds that $|\mu(f) - \mu(g)| \leqslant \mu(|f - g|) \leqslant ||f - g||$. This makes $\mu(f)$ effectively uniformly continuous as a function on $\mathcal{C}_b(\mathbb{R})$.

Let a and b with $a < b$ be computable numbers. For a computable function f on the interval $[a, b]$, its definite integral $\int_a^b f(x)dx$ is a computable number (cf. [6]). We can generalize this fact as follows.

Proposition 1. *Let $\{a_m\}$ and $\{b_m\}$ be computable sequences of reals with $a_m < b_m$ for each m, and let $\{f_n\}$ be a computable sequence of functions on $\mathbb{R}$. Then, $\{\int_{a_m}^{b_m} f_n(x)dx\}$ is a computable (double) sequence of real numbers.*

This proposition yields that, if a sequence of distributions has a computable sequence of density functions, then it is computable and the corresponding sequence of distribution functions is also computable.

We frequently use the following Lemma.

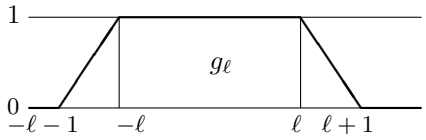
Lemma 2. (Monotone convergence [6]) *Let $\{x_{n,k}\}$ be a computable sequence of reals which converges monotonically to $\{x_n\}$ as k tends to infinity for each n. Then $\{x_n\}$ is computable if and only if the convergence is effective.*

We say that a sequence of functions $\{f_n\}$ is effectively bounded if there exists a recursive function $B(n)$ such that $|f_n(x)| \leqslant 2^{B(n)}$ for each $n, x \in \mathbb{R}$. We give some examples of probability distributions which have bounded density (Example 1).

Proposition 2. *If $\{\mu_m\}$ is vaguely sequentially computable, then it is* weakly sequentially computable, *that is, $\{\mu_m(f_n)\}$ is a computable sequence for all effectively bounded computable sequence of functions $\{f_n\}$.*

Proof. Let $\{f_n\}$ be an effectively bounded computable sequence of functions with an effective bound $B(n)$, and define $g_\ell(x)$ by:

$$g_\ell(x) = \begin{cases} 0 & \text{if } \ x \leqslant -\ell - 1 \\ (x + \ell) + 1 & -\ell - 1 \leqslant x \leqslant -\ell \\ 1 & \text{if } \ -\ell \leqslant x \leqslant \ell \\ -(x - \ell) + 1 & \text{if } \ \ell \leqslant x \leqslant \ell + 1 \\ 0 & \text{if } \ x \geqslant \ell + 1. \end{cases}$$



It is obvious that $\{g_\ell\}$ is a computable sequences of functions with compact support.

Since $g_\ell \uparrow 1$ pointwise, $\mu_m(g_\ell) \uparrow 1$ as ℓ tends to infinity by the bounded convergence theorem for each m, where $\uparrow$ means monotonically increasing convergence. Moreover, $\{\mu_m(g_\ell)\}$ is a computable sequence of reals by vague sequential computability of $\{\mu_m\}$ and the limit 1 is a computable number. So, the convergence of $\mu_m(g_\ell)$ to 1 is effective by Monotone Convergence Lemma 2. Therefore, we obtain a recursive function $N(m,k)$ such that $\mu_m([-\ell-1,\ell+1]^C) \leqslant 1-\mu_m(g_\ell) < 2^{-k}$ if $\ell \geqslant N(m,k)$, where A^C denotes the complement of the set A.

On the other hand, $\{\mu_m(f_n g_\ell)\}$ is a computable triple sequence of reals and

$$|\mu_m(f_n) - \mu_m(f_n g_{\ell+1})| = |\textstyle\int_{[-\ell-1,\ell+1]^C} (1-g_{\ell+1})\, f_n\, \mu_m(dx)|$$
$$\leqslant 2^{B(n)}\, \mu_m([-\ell-1,\ell+1]^C) < 2^{-k}$$

if $\ell \geqslant N(m, B(n)+k)$. This means that $\{\mu_m(f_n g_\ell)\}$ converges effectively to $\{\mu_m(f_n)\}$. Hence $\{\mu_m(f_n)\}$ is a computable sequence of reals. □

Proposition 3. *For a sequentially computable sequence of distribution functions $\{F_m\}$, effective continuity implies effective uniform continuity.*

Proof. By sequential computability of $\{F_m\}$, $\{F_m(n)\}$ and $\{F_m(-n)\}$ are computable sequences of reals. Since, F_m's are distribution functions, $F_m(n) \uparrow 1$ and $F_m(-n) \downarrow 0$ as n tends to infinity for each m. By Lemma 2, there exists a recursive function $N(m,k)$ such that $1-F_m(x) \leqslant 1-F_m(N(m,k)) < 2^{-k}$ for $x > N(m,k)$ and $F_m(x) \leqslant F_m(-N(m,k)) < 2^{-k}$ for $x < -N(m,k)$.

On the other hand, effective continuity of $\{F_m\}$ implies that there exists a recursive function $\alpha(m,n,k)$ such that $x,y \in [-n,n]$ and $|x-y| < 2^{-\alpha(m,n,k)}$ imply $|F_m(x)-F_m(y)| < 2^{-k}$.

If we put $\beta(m,k) = \alpha(m, N(m,k+2), k+2)$ and assume that $|x-y| < 2^{-\beta(m,k)}$, then the following four cases are possible.

The first case: Both x and y are in $[-N(m,k+2), N(m,k+2)]$. In this case, $|F_m(x)-F_m(y)| < 2^{-(k+2)}$.

The second case: Both x and y are in $(N(m,k+2),\infty)$. In this case, $|F_m(x)-F_m(y)| \leqslant |1-F_m(x)| + |1-F_m(y)| < 2^{-(k+1)}$.

The third case: Both x and y are in $(-\infty, -N(m,k+2))$. In this case, $|F_m(x)-F_m(y)| \leqslant |F_m(x)| + |F_m(y)| < 2^{-(k+1)}$.

The last case: One is in $[-N(k+2), N(k+2)]$ and the other is not. Suppose $x < -N(k+2) \leqslant y$, then

$$|F_m(x)-F_m(y)| \leqslant |F_m(x)| + |F_m(-N(k+2))| + |F_m(-N(k+2)) - F_m(y)|$$

$< 3 \cdot 2^{-(k+2)} < 2^{-k}$.

Therefore, we have shown that $\{F_m\}$ is effectively uniformly continuous with respect to $\beta(m,k)$. □

Theorem 1. If a sequence of distribution functions $\{F_m\}$ is sequentially computable, then the corresponding sequence of distributions $\{\mu_m\}$ is computable.

Proof. We prove that $\mu(f)$ is computable if f is a computable function with compact support. For such a function f, there exists an integer m such that $f(x) = 0$ if $|x| \geqslant m$.

Put

$g_p(x) = f(-m + 2^{-p})\chi_{[-m,-m+2^{-p}]}(x)$
$+ \sum_{\ell=-m2^p+1}^{m2^p-1} f((\ell+1)2^{-p})\chi_{(\ell 2^{-p},(\ell+1)2^{-p}]}(x)$.

Then, $\mu(g_p) = \int_{[-m,m]} g_p \mu(dx)$

$= \sum_{\ell=0}^{2m2^p-1} f(-m + (\ell+1)2^{-p})\big(F(-m + (\ell+1)2^{-p}) - F(-m + \ell 2^{-p})\big)$

form a computable sequence of reals by sequential computability of F.

By Lemma 1, f is uniformly computable. So, there exists a recursive function $\alpha(k)$ such that $|f(x)-f(y)| < 2^{-k}$ if $|x-y| < 2^{-\alpha(k)}$. We note that $\{g_p\}$ converges effectively uniformly to f. More precisely, if $p \geqslant \alpha(k)$, then $||f - g_p|| \leqslant 2^{-k}$.

Therefore, for the above α, $p \geqslant \alpha(k+1)$ implies

$$|\mu(f) - \mu(g_p)| \leqslant ||f - g_p|| \leqslant 2^{-k}.$$

This proves the effective convergence of $\{\mu(g_p)\}$ to $\mu(f)$, and hence, $\mu(f)$ is computable. The proof goes through for a sequence $\{F_m\}$. □

If a probability distribution has a bounded density ξ with a bound M, then the corresponding distribution function F satisfies $|F(b)-F(a)| = |\int_a^b \xi(x)dx| \leqslant M|b-a|$. So, we obtain the following lemma.

Lemma 3. *If a sequence of densities of probability distributions is effectively bounded, then the corresponding sequence of distribution functions is effectively uniformly continuous.*

From the lemma above follows that, if a sequence of probability distributions has an effectively bounded sequence of densities, then uniform computability of the corresponding sequence of distribution functions is equivalent to sequential computability.

In the rest of this section, we assume the existence of bounded densities.

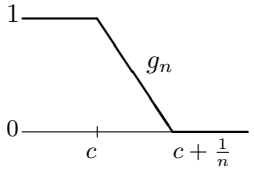
Proposition 4. *Let $\{\mu_m\}$ be a computable sequence of probability distributions which has effectively bounded densities. Then the corresponding sequence of distribution functions $\{F_m\}$ is sequentially computable.*

Proof. We prove that a single distribution function F is uniformly computable if the corresponding probability distribution μ is computable and there exists an integer M such that $|\xi(x)| \leqslant M$ for all x, where ξ is a density of μ.

By Lemma 3, F is effectively uniformly continuous.

We prove that $F(c)$ is computable if c is computable. First, we define the functions $\{g_n\}$ by

$$g_n(x) = \begin{cases} 1 & \text{if } x \leqslant c \\ -n(x-c)+1 & \text{if } c \leqslant x \leqslant c+\frac{1}{n} \\ 0 & \text{if } x \geqslant c+\frac{1}{n} \end{cases}.$$



Then, $\{g_n\}$ is a computable sequence and the following classical properties hold:

$\{g_n\}$ is monotonically decreasing, that is $m < n$ implies $g_m(x) \geqslant g_n(x)$ for all x.

$F(c) \leqslant \mu(g_n) \leqslant F(c+\frac{1}{n})$.

$F(c) = \lim_{n\to\infty} \mu(g_n)$ holds by the bounded convergence theorem.

On the other hand, $\{\mu(g_n)\}$ is a computable sequence of reals by the assumption and Proposition 2.

We obtain $0 \leqslant \mu(g_n) - F(c) = \int_c^{c+\frac{1}{n}} g_n(x)\xi(x)dx \leqslant \frac{M}{n}$.

Therefore, the convergence of $\mu(g_n)$ to $F(c)$ is effective, and hence $F(c)$ is computable.

This proof is also valid for a sequence $\{c_\ell\}$. The entire argument can be extended to a sequence $\{\mu_m\}$. □

We obtain the following theorem by Theorem 1, Lemma 3 and Proposition 4.

Theorem 2. *If a sequence of distributions $\{\mu_n\}$ has effectively bounded densities, then the computability of $\{\mu_n\}$ is equivalent to the uniform computability of the corresponding sequence of distribution functions.*

Example 1. In this example, μ denotes a probability distribution on $\mathbb{R}$, F denotes the corresponding distribution function and ξ denotes the corresponding density.

(1) Uniform distribution on $[0,1]$:

$$\xi(x) = \chi_{[0,1]}(x); \quad F(x) = \begin{cases} 0 \text{ if } x \leqslant 0 \\ x \text{ if } 0 \leqslant x \leqslant 1 \\ 1 \text{ if } x \geqslant 1 \end{cases}.$$

$\xi(x)$ is bounded, but not continuous. On the other hand, $F(x)$ is continuous and indeed uniformly computable.

(2) Gaussian distribution: $\xi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$, $F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{1}{2}y^2} dy$.

(3) Exponential distribution: $\xi(x) = e^{-x}$, $F(x) = 1 - e^{-x}$.

In (2) and (3), both ξ and F are computable. □

Example 2. (Translated Unit Distribution) The translated unit distribution δ_c is defined by

$$\delta_c(A) = \begin{cases} 1 \text{ if } c \in A \\ 0 \text{ otherwise} \end{cases}.$$

The corresponding distribution function is

$$F(x) = \chi_{[c,\infty)}(x) = \begin{cases} 0 \ (x < c) \\ 1 \ (x \geqslant c) \end{cases}.$$

The translated unit distribution is computable if c is a computable number. Its distribution function is not continuous. □

4 Convergence of probability distributions and distribution functions

We define effective convergence of probability distributions as an effectivization of classical vague convergence of probability distributions.

Definition 3. (Effective convergence of a sequence of probability distributions)
A sequence of probability distributions $\{\mu_m\}$ is said to effectively converge to a probability distribution μ if $\{\mu_m(f_n)\}$ converges effectively to $\{\mu(f_n)\}$ for any computable sequence of functions with compact support $\{f_n\}$.

It is well known that the set of all uniformly computable functions on a closed interval $[a, b]$ is dense in the set of all continuous functions on $[a, b]$ for any pair of computable numbers a and b with $a < b$. So, effective convergence of a sequence of probability distributions implies classical vague convergence.

The following proposition follows immediately.

Proposition 5. *If a computable sequence of probability distributions $\{\mu_n\}$ effectively converges to a probability distribution μ, then μ is computable.*

Proposition 6. *Let $\{\mu_m\}$ be a computable sequence of probability distributions and let μ be a computable probability distribution. If $\{\mu_m\}$ converges effectively to μ, then $\{\mu_m\}$ effectively weakly converges to μ, that is, $\{\mu_m(f_n)\}$ converges effectively to $\{\mu(f_n)\}$ for all effectively bounded computable sequence of functions $\{f_n\}$.*

Proof. We prove that $\{\mu_m(f)\}$ converges effectively to $\{\mu(f)\}$ for a bounded computable function f. For such f, there exists an integer M which satisfies that $|f(x)| \leqslant 2^M$ for all x.

Let us take a computable sequence of functions $\{g_\ell\}$ with compact support which is defined in the proof of Proposition 2. Then, we obtain a recursive function $N(k)$ which satisfies that $\mu([-\ell, \ell]^C) \leqslant 1 - \mu(g_{\ell-1}) < 2^{-k}$ for $\ell \geqslant N(k)$. Moreover, by effective convergence of $\{\mu_m(g_\ell)\}$ to $\{\mu(g_\ell)\}$, there exists a recursive function $\alpha(\ell, k)$ such that $m \geqslant \alpha(\ell, k)$ implies $|\mu_m(g_\ell) - \mu(g_\ell)| < 2^{-k}$.

Therefore, we obtain $1 - \mu(g_{N(k)}) < 2^{-k}$ and $m \geqslant \alpha(N(k), k)$ implies $|1 - \mu_m(g_{N(k)})| \leqslant |\mu_m(g_{N(k)}) - \mu(g_{N(k)})| + |1 - \mu(g_{N(k)})| < 2 \cdot 2^{-k}$.

On the other hand, since $\{fg_\ell\}$ is a computable sequence of functions with compact support, $\{\mu_m(fg_\ell)\}$ converges effectively to $\{\mu(fg_\ell)\}$. So, there exists a recursive function $\beta(\ell, k)$ such that $|\mu_m(fg_\ell) - \mu(f\, g_\ell)| < 2^{-k}$ for $m \geqslant \beta(\ell, k)$.

Therefore, $m \geqslant \beta(N(k), k)$ implies $|\mu_m(fg_{N(k)}) - \mu(fg_{N(k)})| < 2^{-k}$.

If we take $j = k + M + 2$ and assume $m \geqslant \max\{\alpha(N(j), j), \beta(N(j), j)\}$, then

$$\begin{aligned}
&|\mu_m(f) - \mu(f)| \\
&\leqslant |\mu_m(f) - \mu_m(fg_{N(j)})| + |\mu_m(fg_{N(j)}) - \mu(fg_{N(j)})| \\
&\qquad + |\mu(f) - \mu(fg_{N(j)})| \\
&\leqslant 2^M(1 - \mu_m(g_{N(j)})) + |\mu_m(fg_{N(j)}) - \mu(fg_{N(j)})| + 2^M(1 - \mu(g_{N(j)})) \\
&< 2 \cdot 2^{-(k+2)} + 2^{-(k+2)} + 2^{-(k+2)} = 2^{k}.
\end{aligned}$$

This proves the effective convergence of $\{\mu_n(f)\}$ to $\mu(f)$. □

Definition 4. (Effective pointwise convergence of functions)
A sequence of functions $\{F_m\}$ is said to converge effectively pointwise to a function F if $\{F_m(x_n)\}$ converges effectively to $\{F(x_n)\}$ for all computable sequence $\{x_n\}$.

By definition, the following proposition holds.

Proposition 7. *For a computable sequence of functions $\{F_m\}$, if it converge effectively pointwise to a function F, then F is sequentially computable.*

By Lemma 3, the existence of density of F implies the effective uniform continuity of F. So, we obtain the following proposition.

Proposition 8. *Let us consider a sequentially computable sequence of distribution functions $\{F_m\}$ and a distribution function F. If $\{F_m\}$ converges effectively pointwise to F, then the sequence of corresponding probability distributions $\{\mu_m\}$ converges effectively to μ.*

Proof. We follow the classical proof and prove that $\mu_m(f)$ converges effectively to $\mu(f)$ for a computable function with compact support f. By Lemma 1, f is uniformly computable. So, there exists a recursive function $\alpha(k)$, which is an effective modulus of uniform continuity of f. We also obtain an integer N such that $f(x) = 0$ if $|x| > 2^N$ and an integer B such that $|f(x)| \leqslant 2^B$ for all x.

Define $f_n(x) = \sum_{j=-2^N 2^n+1}^{2^N 2^n} f(j2^{-n})\chi_{((j-1)2^{-n}, j2^{-n}]}(x)$.

Then, $\mu_m(f_n) = \sum_{j=-2^N 2^n+1}^{2^N 2^n} f(j2^{-n})(F_m(j2^{-n}) - F_m((j-1)2^{-n}))$

and $\mu(f_n) = \sum_{j=-2^N 2^n+1}^{2^N 2^n} f(j2^{-n})(F(j2^{-n}) - F((j-1)2^{-n}))$ hold.

We note that each of the right-hand sides of the last two equations forms a computable sequence of reals.

By the definitions of f_n and α, $|f(x) - f_{\alpha(k)}(x)| = |f(x) - f(j2^{-\alpha(k)})| < 2^{-k}$ if $x \in ((j-1)2^{-\alpha(k)}, j2^{-\alpha(k)}]$.

Hence, we obtain $|\mu_m(f_{\alpha(k)}) - \mu_m(f)| \leqslant 2^{-k}$ and $|\mu(f_{\alpha(k)}) - \mu(f)| \leqslant 2^{-k}$.

By effective pointwise convergence of $\{F_m\}$ to F, there exists a recursive function $\beta(k, n, j)$ such that $m \geqslant \beta(k, n, j)$ implies

$|F_m(j2^{-n}) - F(j2^{-n})| < 2^{-k}$.

Define $\tilde{k} = N + 1 + B + \alpha(k + 3) + k + 3$ and

$\gamma(k) = \max\{\beta(\tilde{k}, \alpha(k + 3), 0), \ldots, \beta(\tilde{k}, \alpha(k + 3), 2^{N+1}2^{\alpha(k+3)})\}$.

Assume $m \geqslant \gamma(k)$. Then,

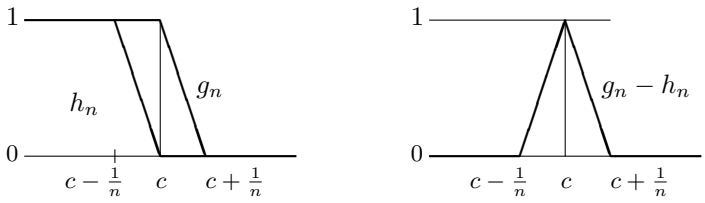
$$\begin{aligned}
&|\mu_m(f) - \mu(f)| \\
&\leqslant |\mu_m(f_{\alpha(k+3)}) - \mu(f_{\alpha(k+3)})| + |\mu(f_{\alpha(k+3)}) - \mu(f)| \\
&\qquad +|\mu_m(f_{\alpha(k+3)}) - \mu_m(f)| \\
&\leqslant \textstyle\sum_{j=-2^N 2^{\alpha(k+3)}+1}^{2^N 2^{\alpha(k+3)}} \big\{ |f(j2^{-\alpha(k+3)})|\, |F_m(j2^{-\alpha(k+3)}) - F(j2^{-\alpha(k+3)})| \\
&\qquad +|f(j2^{-\alpha(k+3)})|\, |F_m((j-1)2^{-\alpha(k+3)}) - F((j-1)2^{-\alpha(k+3)})| \big\} \\
&\qquad +2 \cdot 2^{-(k+3)} \\
&\leqslant 2(2^{N+1}2^{\alpha(k+3)})2^B 2^{-\tilde{k}} + 2 \cdot 2^{-(k+3)} < 2^{-k}.
\end{aligned}$$

This proves the effective convergence of $\{\mu_m(f)\}$ to $\mu(f)$. □

Proposition 9. *Let us consider a computable sequence of probability distributions $\{\mu_m\}$ and a computable probability distribution μ with a bounded density. If $\{\mu_m\}$ converges effectively to μ, then the sequence of the corresponding distribution functions $\{F_m\}$ converges effectively pointwise to F, the distribution function corresponding to μ.*

Proof. We prove that $\{F_m(c)\}$ converges effectively to $F(c)$ if c is computable. Let us define $h_n(x)$ by

$$h_n(x) = \begin{cases} 1 & \text{if } \ x \leqslant c - \frac{1}{n} \\ -n(x-c) & \text{if } \ c - \frac{1}{n} \leqslant x \leqslant c \\ 0 & \text{if } \ x \geqslant c \end{cases}.$$



It holds that $h_n(x) \leqslant \chi_{(-\infty,c]}(x) \leqslant g_n(x)$, where g_n is the function defined in the proof of Proposition 4. Hence, we obtain $\mu(h_n) \leqslant F(c) \leqslant \mu(g_n)$ and $\mu_m(h_n) \leqslant F_m(c) \leqslant \mu_m(g_n)$.

Meanwhile, $\{g_n\}$ and $\{h_n\}$ are effectively bounded computable sequences of functions if c is a computable real. Hence, by Proposition 6, $\{\mu_m(h_n)\}$ and $\{\mu_m(g_n)\}$ converge effectively to $\mu(h_n)$ and $\mu(g_n)$ respectively as m tends to infinity. So, there exists a recursive function $\alpha(n,k)$ such that $m \geqslant \alpha(n,k)$ implies $|\mu_m(h_n) - \mu(h_n)| < 2^{-k}$ and $|\mu_m(g_n) - \mu(g_n)| < 2^{-k}$.
Hence, $m \geqslant \alpha(n,k)$ implies $\mu(h_n) - 2^{-k} \leqslant \mu_m(h_n) \leqslant F_m(c) \leqslant \mu_m(g_n) < \mu(g_n) + 2^{-k}$.

On the other hand,

$$g_n(x) - h_n(x) = \begin{cases} 0 & \text{if } x \leqslant c - \frac{1}{n} \\ n(x-c)+1 & \text{if } c - \frac{1}{n} \leqslant x \leqslant c \\ -n(x-c)+1 & \text{if } c \leqslant x \leqslant c + \frac{1}{n} \\ 0 & \text{if } x \geqslant c + \frac{1}{n} \end{cases}.$$

If we take an integer M such that 2^M is a bound of a density of μ, then $\mu(g_n - h_n) \leqslant \frac{2 \cdot 2^M}{n}$. If we put $N = 2^{k+M+2}$, then $\mu(g_N - h_N) \leqslant 2^{-(k+1)}$. Hence, we obtain $\mu(h_N) > F(c) - 2^{-(k+1)}$ and $\mu(g_N) < F(c) + 2^{-(k+1)}$. Therefore, $m \geqslant \alpha(N, k+1)$ implies $|F_m(c) - F(c)| < 2^{-k}$.

This proves the effective convergence of $\{F_m(c)\}$ to $F(c)$.

The argument above can be modified to a computable sequence of real numbers $\{c_n\}$. □

In the case where μ has a bounded density and $\{\mu_n\}$ has effectively bounded densities, we obtain the following theorem from Propositions 4, 8 and 9.

Theorem 3. *Let us consider a computable sequence of probability distributions $\{\mu_m\}$ with effectively bounded densities and a computable distribution μ with a bounded density. We denote their distribution functions with $\{F_m\}$ and F respectively. Then, $\{\mu_m\}$ converges effectively to μ if and only if $\{F_m\}$ converges effectively pointwise to F.*

In the following examples, μ_m and μ denote probability distributions, ξ_m and ξ denote the corresponding densities (if they exist) and F_m and F denote the corresponding distribution functions.

Example 3. Let μ_m be the Gaussian distribution with mean $\frac{1}{m}$ and variance $\frac{m}{m+1}$ and μ be the Gaussian distribution with mean 0 and variance 1, that is,

$$\xi_m(x) = \frac{\sqrt{m+1}}{\sqrt{2\pi m}} e^{-\frac{m+1}{2m}(x - \frac{1}{m})^2} \text{ and } \xi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}.$$

$\{\xi_m\}$ is computable and converges effectively to ξ. It also holds that $|\xi_m(x)|$, $|\xi(x)| \leqslant 1$. So, the assumption of Theorem 3 holds. By virtue of the properties of the densities, the effective convergence of $\{\mu_m\}$ to μ and that of $\{F_m\}$ to F are the consequences of Effective Dominated Convergence Theorem (see [5]). □

Example 4. Let ξ_m be defined as follows.

$$\xi_m(x) = \begin{cases} 0 & \text{if } x \leqslant -\frac{1}{m} \\ \frac{m}{2}x + \frac{1}{2} & \text{if } -\frac{1}{m} < x < \frac{1}{m} \\ 1 & \text{if } \frac{1}{m} \leqslant x \leqslant 1 - \frac{1}{m} \\ -\frac{m}{2}(x-1) + \frac{1}{2} & \text{if } 1 - \frac{1}{m} < x < 1 + \frac{1}{m} \\ 0 & \text{if } x \geqslant 1 + \frac{1}{m} \end{cases}.$$

$\{\xi_m\}$ is a computable sequence with compact support, and $\{\mu_m\}$ converges effectively to the uniform distribution on $[0,1]$ (cf. Example 1(1)). Although the density of the uniform distribution is not continuous, it is still bounded. So, the assumption of Theorem 3 holds.

By the inequality $|\int_{\mathbb{R}} f(x)\xi_m(x)dx - \int_{\mathbb{R}} f(x)\xi_{[0,1]}(x)dx| \leqslant ||f||$, we can prove the effective convergence of $\{\mu_m\}$, and hence of $\{F_m\}$. □

Example 5. Let ξ_n be defined as follows.

$$\xi_n(x) = \begin{cases} x \leqslant -\frac{1}{n} \\ n^2x + n \quad \text{if} \quad -\frac{1}{n} \leqslant x \leqslant 0 \\ -n^2x + n \text{ if } \ 0 \leqslant x \leqslant \frac{1}{n} \\ 0 \qquad\quad \text{if} \quad x \geqslant \frac{1}{n} \end{cases} .$$

$\{\mu_n\}$ converges effectively to the unit distribution δ_0, which does not have a density. This is a case to which Theorem 3 cannot be applied. Indeed, $F_n(0) = \frac{1}{2}$ but $F(0) = 1$. So, $\{F_n(0)\}$ does not converge to $F(0) = 1$. □

References

1. Billingsley, P. *Convergence of Probability Measures*, John Wiley & Sons, 1968.
2. Feller, W. *An Introduction to Probability Theory and its Applications, Vol. I, Third Edition*, John Wiley & Sons, 1968.
3. Feller, W. *An Introduction to Probability Theory and its Applications, Vol. II*, John Wiley & Sons, 1966.
4. Ito, K. *An Introduction to Probability Theory*, Cambridge University Press, 1984 (translated by the author from *Probability Theory* (in Japanese), Iwanami Shoten, 1978, 1991).
5. Mori, T., M. Yasugi and Y. Tsujii. *Effective Fine-convergence of Walsh-Fourier series,* MLQ 54, 519-534, 2008.
6. Pour-El, M.B. and J. I. Richards. *Computability in Analysis and Physics*, Springer, 1988.
7. Rényi, A. *Probability Theory*, Dover, 2007. (North-Holland, 1970.)
8. Schröder, M. Admissible Representations of Probability Measures, ENTCS 167(2007), 61-78.
9. Schröder, M. and A. Simpson. Representing Probability Measures using Probabilistic Processes, CCA 2005 (Kyoto), 211-226.
10. Weihrauch, K. Computability on the probability measures on the Borel sets of the unit interval, TCS 219, 421-437, 1999.

How Discontinuous is Computing Nash Equilibria? (Extended Abstract)

Arno Pauly

University of Cambridge Computer Laboratory
Cambridge, UK
Arno.Pauly@cl.cam.ac.uk

Abstract. We investigate the degree of discontinuity of several solution concepts from non-cooperative game theory. While the consideration of Nash equilibria forms the core of our work, also pure and correlated equilibria are dealt with. Formally, we restrict the treatment to two player games, but results and proofs extend to the n-player case. As a side result, the degree of discontinuity of solving systems of linear inequalities is settled.

Keywords. Game Theory, Computable Analysis, Nash Equilibrium, Discontinuity

1 Introduction

Both for applications and theoretical considerations, the algorithmic task of computing Nash equilibria from certain representations of games is of immense importance. A natural mathematical formulation of game theory uses the real numbers for payoffs and for mixed strategies, while classical models for algorithms require a restriction to countable sets. By imposing suitable restrictions and modifications to obtain countable problems, the complexity of computing a Nash equilibrium for a normal form game was proven to be PPAD-complete ([1], [2]).

Here we will use another approach: Instead of limiting the problem, we will extend the theory of computation. While the TTE-framework ([3]) is perfectly capable of formulating the task of computing Nash equilibria from normal form games, we will see that even the most trivial cases are discontinuous, and hence not computable.

To gain a deeper understanding of the problem, its degree of discontinuity will be studied. Mirroring an approach in the study of game theory using classical computational complexity, we will also examine other solution concepts such as correlated equilibria. While correlated equilibria seem to be computationally easier than Nash equilibria[1], we will prove that both concepts share a degree

[1] In [4] several decision problems regarding Nash equilibria and correlated equilibria were compared, most of them turned out to be NP-hard for Nash equilibria and to be in P for correlated equilibria.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 197-208
http://drops.dagstuhl.de/opus/volltexte/2009/2271

of discontinuity. Limitation to pure strategies yields a strictly less discontinuous problem, the classical problem can be solved by a cubic algorithm[2].

Due to space restrictions, most of the proofs are omitted here. A more comprehensive version including proofs is [10].

2 Preliminaries

2.1 Game Theory

An $n \times m$ bi-matrix game is simply given by two $n \times m$ real valued matrices A and B. Two players simultaneously pick an index, row player chooses an $i \in \{1, 2, \ldots, n\}$ and column player chooses an $j \in \{1, 2, \ldots, m\}$. Row player gets A_{ij} as a reward, column player gets B_{ij}. We consider several solution concepts defined as equilibria, where no player has an incentive to change her strategy unilaterally.

Definition 1. *A pure equilibrium for a $n \times m$ bi-matrix game (A, B) is a pair $(i, j) \in \{1, \ldots, n\} \times \{1, \ldots, m\}$ satisfying $A_{ij} \geq A_{kj}$ for all $k \in \{1, \ldots, n\}$ and $B_{ij} \geq B_{il}$ for all $l \in \{1, \ldots, m\}$.*

As pure equilibria do not exist for all games, a more general notion is introduced. If both players can randomize independently over their actions, one is led to the definition of an m-mixed strategy as an m-dimensional real valued vector s with non-negative coefficients and $\sum_{j=1}^{m} s_j = 1$. The set of m-mixed strategies will be denoted by S^m.

Definition 2. *A Nash equilibrium for an $n \times m$ bi-matrix game (A, B) is a pair $(\hat{x}, \hat{y}) \in S^n \times S^m$ satisfying $\hat{x}^T A \hat{y} \geq x^T A \hat{y}$ for all $x \in S^n$ and $\hat{x}^T B \hat{y} \geq \hat{x}^T B y$ for all $y \in S^m$.*

If $(\hat{x}, \hat{y})$ is a Nash equilibrium, again neither of the players can improve her payoff by changing her mixed strategy unilaterally. A famous result by John Nash ([7]) established that Nash equilibria in bi-matrix games always exist. By identifying a pure strategy with the mixed strategy that puts weight 1 on it, pure equilibria can be considered a special case of Nash equilibria. An even more general solution concept can be obtained by allowing the individual player's randomization processes to be correlated ([8]).

Definition 3. *A correlated equilibrium for a $n \times m$ bi-matrix game is a real valued $n \times m$ matrix C with non-negative entries and $\sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij} = 1$ so that*

$$\sum_{j=1}^{m} A_{ij} C_{ij} \geq \sum_{j=1}^{m} A_{lj} C_{ij}$$

[2] There are, however, several interesting hardness results for finding pure equilibria in games ([5], [6]), originating in other representations or requiring additional properties.

holds for all $i, l \in \{1, 2, \ldots, n\}$ *and*

$$\sum_{i=1}^{n} B_{ij} C_{ij} \geq \sum_{i=1}^{n} B_{ik} C_{ij}$$

holds for all $j, k \in \{1, 2, \ldots, m\}$.

Given a Nash equilibrium (x, y), a correlated equilibrium can be constructed as $C_{ij} = x_i y_j$, while each correlated equilibrium of this form can be obtained from a Nash equilibrium, allowing us to consider Nash equilibria as special cases of correlated equilibria. Thus, finding a correlated equilibrium has to be easier than finding a Nash equilibrium, as we just presented a reduction.

Another way of creating an easier problem consists in a restriction of the games used. A zero-sum game is a bi-matrix game of the form $(A, -A)$.

2.2 Representing Games

In order to consider games as inputs to Type-2-Machines, they have to be coded into infinite sequences. The choice of the countable alphabet used is irrelevant for the theory, to simplify proofs we will use either $\{0, 1\}$ or $\mathbb{N}$, depending on the context. The degrees of discontinuity we study are those of the realizations, that is of functions turning names of instances into names of solutions. Since all occurring representations will be admissible, topological properties carry over between sets of games and sets of names for games, etc.

As games in normal form are pairs of real matrices, and (possible) equilibria pairs of real vectors (or again real matrices), one can quickly derive suitable representations by using product and coproduct representations ([3], [9]), starting from any representation of the real numbers.

The standard representation ρ of the real numbers is chosen for various reasons; it is admissible and provides a convincing class of computable functions, in contrast to some of the alternatives ([3], [11]). Additionally, as demonstrated in [12], the representation ρ is equivalent to the representation naturally arising for the results of repeated physical measurements. For defining ρ, we fix a bijection $\nu : \mathbb{N} \to \mathbb{Q}$ with $\nu(0) = 0$, so that all the usual operations on $\mathbb{Q}$ are computable w.r.t. ν.

Definition 4. *Let* $\rho(w) = x \in \mathbb{R}$ *hold for* $w \in \mathbb{N}^{\mathbb{N}}$, *if* $|\nu(w(i)) - x| \leq 2^{-i}$ *holds for all* $i \in \mathbb{N}$.

Definition 5. *Let* w *be a* Γ*-name for the bi-matrix game* (A, B), *if*

1. $w = 0^n 1^m 0 w_2$, *when* (A, B) *is an* $n \times m$ *game*
2. $w_2 = \langle w^a, w^b \rangle$, *where* $\langle\ \rangle$ *denotes the usual pairing function*
3. $w^a = \langle w^a_{11}, \ldots w^a_{n1}, w^a_{12}, \ldots, w^a_{nm} \rangle$
4. $w^b = \langle w^b_{11}, \ldots w^b_{n1}, w^b_{12}, \ldots, w^b_{nm} \rangle$
5. $\rho(w^a_{ij}) = A_{ij}$
6. $\rho(w^b_{ij}) = B_{ij}$

Representations for pure, Nash and correlated equilibria can be derived in the same fashion. Detailed definitions are omitted here.

2.3 Comparing Discontinuity

As games can have multiple equilibria, we do not consider a function assigning an equilibrium to each game, but rather a multi-valued function. We will identify a multi-valued functions with the set of its choice functions. To compare the discontinuity of such sets, Type-2-Reducibility as studied in e.g. ([14], [15], [16], [17], [25], [18], [9]) is used, as well as the Level of a function (or a set of functions), introduced in [17].

We use the following definition of Type-2-Reducibility:

Definition 6. *Let A, B be multi-valued functions. Then $A \leq_2 B$ holds, iff there are continuous partial functions F, G with $w \mapsto F(w, g(G(w))) \in A$ for each $g \in B$.*

As demonstrated in [9] (for suprema) and [13] (for infima), $\leq_2$ induces a completely distributive complete lattice. We use $\lceil P_n \rceil_{n\in\mathbb{N}}$ to denote the supremum of a countable family $(P_n)_{n\in\mathbb{N}}$. This allows to consider the degree of discontinuity of finding equilibria in any game as the supremum of the degrees of discontinuity of finding equilibria in games with fixed size.

As the Level will play only a minor role in our considerations, we refer to [9] for definitions.

3 Single Player Games and Pure Equilibria

From the perspective of game theory, single player games are trivial: The acting player chooses whatever action is best for her. As a discrete computation problem, this amounts to finding a maximum in a list of integers, a task that can be solved in linear time or logarithmic space. As the problem posed over the reals is discontinuous, we will study the problems $1\textsc{Pure}_n$ and $1\textsc{Pure}$ of finding pure equilibria in single player games with n actions and without fixed game sizes. It shall be noted that single player games can be identified with $n \times 1$ bi-matrix games, justifying their inclusion.

As every $n \times 1$ bi-matrix game has a pure equilibrium, and $C_{ij} > 0$ can only hold in a correlated equilibrium C, if the entry A_{i1} is maximal in A (and thus $(i, 1)$ is a pure equilibrium), finding pure, Nash and correlated equilibria is equivalent for single player games, so the restriction to pure equilibria does not invoke any loss of generality.

The degree of discontinuity of $1\textsc{Pure}_n$ turns out to be equivalent to another family of problems, $MLPO_n$, introduced in [14] as generalizations of the *lesser limited principle of omniscience* (LPO) studied in constructive mathematics ([19]).

Definition 7. *A function $f : \{(p_1, \ldots, p_n) \in (\mathbb{N}^\mathbb{N})^n \mid \exists i \leq n\ p_i = 0^\mathbb{N}\} \to \{1, 2, \ldots, n\}$ is in $MLPO_n$, if it fulfills $p_{f(p_1,p_2,\ldots,p_n)} = 0^\mathbb{N}$ for all valid $(p_1, p_2, ..., p_n)$.*

Theorem 1. $MLPO_n \equiv_2 1\textsc{Pure}_n$

In the next step, we extend the scope of consideration to finding pure equilibria in arbitrary bi-matrix games. The relevant problems are PURE_{nm}, where the size of the game is restricted to $n \times m$, and the general case denoted by PURE. For obtaining results, reducibility to $MLPO_n$ shall be expressed by a partition property:

Lemma 1. *Let H be a multi-valued function defined on a strongly zero-dimensional metrisable space*[3] *X. Then $H \leq_2 MLPO_n$ holds, iff there are n closed sets A_i, $i \leq n$ with $X = \bigcup_{i=1}^{n} A_i$, so that for each $i \leq n$, there is an $f^i \in H$ so that $f^i_{|A_i}$ is continuous.*

Theorem 2. $\text{PURE}_{nm} \leq_2 MLPO_{n*m}$.

Proof. Given an $n \times m$ bi-matrix game (A, B), the condition for the pair (i, j) to be a pure equilibrium is $A_{ij} \geq A_{kj}$ and $B_{ij} \geq B_{il}$ for all $k \leq n$, $l \leq m$. This implies that the set $P^{ij}_{nm} = \{(A, B) \mid (i, j) \text{ is an equilibrium of } (A, B)\} \subseteq \mathbb{R}^{nm} \times \mathbb{R}^{nm}$ is closed. Due to the admissibility of Γ, the set of corresponding names for the games is also closed. As the set of $n \times m$ bi-matrix games which have a pure strategy equilibrium is the union $\bigcup_{i \leq n, j \leq m} P^{ij}_{nm}$, an application of Lemma 1 yields the claim.

Corollary 1. $1\text{PURE} \equiv_2 \text{PURE}$.

Proof. As both problems are the respective limits, considering Theorems 1 and 2 is sufficient.

The same reasoning used to establish the equivalence of finding pure strategies in 1 player games and in 2 player games can directly be extended to any finite number of players. While Nash and correlated equilibria have the same degree of discontinuity as pure equilibria in single player games, we will continue to show that a higher degree of discontinuity emerges in the two player case.

4 Nash and correlated equilibria in bi-matrix games

We will now consider Nash and correlated equilibria in bi-matrix games. The problems CORR_{nm} and NASH_{nm} are the fixed size versions, CORR and NASH the general problems. An additional dimension of the problem is whether the games are zero-sum, yielding the problems ZCORR_{nm}, ZNASH_{nm} and the corresponding general problems. Straight-forward reasoning yields the reductions:

$$\text{ZCORR}_{nm} \leq_2 \text{CORR}_{nm} \leq_2 \text{NASH}_{nm} \quad \text{ZCORR}_{nm} \leq_2 \text{ZNASH}_{nm} \leq_2 \text{NASH}_{nm}$$

[3] Examples for such spaces are $\{0, 1\}^{\mathbb{N}}$ and $\mathbb{N}^{\mathbb{N}}$ with their standard topologies. A brief characterization of strongly zero-dimensional metrisable spaces can be found in [9], for details we refer to [17] and [20].

4.1 The discontinuity of robust division

Similar to $MLPO_n$ being representative of the kind of discontinuity we face when searching for pure equilibria, we will start with considering division, which will turn out to be typical for correlated and Nash equilibria. Computing $\frac{a}{b}$ given two real numbers a, $b \neq 0$ is continuous, of course. However, testing whether $b \neq 0$ is not. A robust variant of division, which accepts division by zero and returns an arbitrary value, is not continuous anymore:

Definition 8. *Let* rDiv *be the set of functions d defined on $\{(u,v) \mid 0 \leq \rho(u) \leq \rho(v)\}$ satisfying $\rho(d(u,v)) = \frac{\rho(u)}{\rho(v)}$ for $\rho(v) > 0$.*

While $\mathrm{Lev}(\text{rDiv}) = 2$ establishes robust division as an only slightly discontinuous problem, the following result shows that robust division introduces a new kind of discontinuity not present in finding pure equilibria.

Theorem 3. rDiv $\not\leq_2$ Pure.

We will now use modifications of the game *matching pennies* as a gadget to implement divisions in a game.

$$A = \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} \quad B = -A \quad MP(a,b) = (A,B)$$

If both $a > 0$ and $b > 0$, the unique correlated equilibrium is obtained from the unique Nash equilibrium $x = y = (\frac{b}{a+b}, \frac{a}{a+b})$. If $a = 0$, $b > 0$, then (x,y) is an equilibrium, iff $y = (1,0)$, and for $a > 0$, $b = 0$ we have $y = (0,1)$.

Theorem 4. rDiv $\leq_2$ ZCorr$_{22}$

Proof. Given a pair of ρ-names for real numbers a, b with $0 \leq a \leq b$, a name for the game $MP(a, b-a)$ can be computed. A correlated equilibrium C of $MP(a, b-a)$ has the form:

$$C = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} xy & x(1-y) \\ (1-x)y & (1-x)(1-y) \end{pmatrix}$$

Thus, one can obtain $c_{11} + c_{21} = y = \frac{a}{b}$ for $b > 0$.

Theorem 4 in conjunction with Theorem 3 implies ZCorr$_{22}$ $\not\leq_2$ Pure, so even the simplest case of finding mixed strategies is not reducible to finding pure strategies. The problem rDiv itself cannot capture the discontinuity of finding Nash equilibria, due to $\mathrm{Lev}(\text{ZNash}_{22}) = 4$ (s. Subsection 5.2), compelling us to derive a sequence of problems with increasing level from rDiv.

4.2 Products of Problems and Products of Games

The product of functions can be considered as computing all of them in parallel. This will allow us to specify exactly the degree of discontinuity of problems solvable by multiple robust divisions, once we defined products for multi-valued functions. The following definitions and results on the products of multi-valued functions and their discontinuity extend corresponding results from [18].

Definition 9. *For functions* $f : X \to Y$, $g : U \to V$, *define* $\langle f, g\rangle : (X \times U) \to (Y \times V)$ *through* $\langle f, g\rangle(x, u) = (f(x), g(u))$. *Define* $\langle f\rangle^1 = f$ *and* $\langle f\rangle^{n+1} = \langle f, \langle f\rangle^n\rangle$.

Definition 10. *For relations* P, Q, *define* $\langle P, Q\rangle = \{\langle f, g\rangle \mid f \in P, g \in Q\}$. *Define* $\langle P\rangle = P$ *and* $\langle P\rangle^{n+1} = \langle P, \langle P\rangle^n\rangle$.

$\lceil P, Q\rceil \leq_2 \langle P, Q\rangle$ holds, but the converse is false in general. If $f \leq_2 g$ holds, then also $\langle f, h\rangle \leq_2 \langle g, h\rangle$. As $\langle\ \rangle$ is associative, it can be extended to any finite number of arguments in the standard way. There is a useful distributive law for $\lceil\ \rceil$ and $\langle\ \rangle$ which we will state as $\langle P, \lceil Q_i\rceil_{i\in\mathbb{N}}\rangle \equiv_2 \lceil\langle P, Q_i\rangle\rceil_{i\in\mathbb{N}}$.

For games, our notion of a product will be inspired by the model of playing two independent games at once. This will allow us to establish a link between products of relations and products of games. We will use $[\ ]$ to denote a bijection between $\{1, 2, \ldots, n\} \times \{1, 2, \ldots, m\}$ and $\{1, 2, \ldots, nm\}$ for suitable n, m.

Definition 11. *Given an* $n_1 \times m_1$ *bi-matrix game* (A^1, B^1) *and an* $n_2 \times m_2$ *bi-matrix game* (A^2, B^2), *we define the* $(n_1 n_2) \times (m_1 m_2)$ *product game* (A, B) *through* $A_{[i_1,i_2][j_1,j_2]} = A^1_{i_1 j_1} + A^2_{i_2 j_2}$ *and* $B_{[i_1,i_2][j_1,j_2]} = B^1_{i_1 j_1} + B^2_{i_2 j_2}$.

The product of two games is a constant-sum game, iff both games are constant-sum[4]. If (x^1, y^1) is an equilibrium (either pure or Nash) of (A^1, B^1), and (x^2, y^2) is an equilibrium of (A^2, B^2), then (x, y) is an equilibrium (of the same type) of the product game where $x_{[i_1 i_2]} = x^1_{i_1} x^2_{i_2}$ and $y_{[m_1 m_2]} = y^1_{m_1} y^2_{m_2}$. Conversely, if (x, y) is an equilibrium of the product game, an equilibrium (x^1, y^1) for (A^1, B^1) can be obtained through $x^1_i = \sum_{k=1}^{n_2} x_{[i,k]}$ and $y^1_j = \sum_{l=1}^{m_2} y_{[j,l]}$, analogously an equilibrium (x^2, y^2) for (A^2, B^2) can be computed. Analogous statements hold for correlated equilibria.

As the product game can be computed from the constituent games, we can use the properties of the products of games to obtain the following results regarding the problem of finding equilibria:

Theorem 5. *Let* $\textsc{Game} \in \{\textsc{Pure}, \textsc{ZCorr}, \textsc{ZNash}, \textsc{Corr}, \textsc{Nash}\}$. *Then* $\langle \textsc{Game}_{nm}, \textsc{Game}_{kl}\rangle \leq_2 \textsc{Game}_{(nk),(ml)}$.

Theorem 6. *Let* $\textsc{Game} \in \{\textsc{Pure}, \textsc{ZCorr}, \textsc{ZNash}, \textsc{Corr}, \textsc{Nash}\}$. *Then* $\langle \textsc{Game}\rangle^n \equiv_2 \textsc{Game}$ *for all* $n \in \mathbb{N}$.

[4] As equilibria finding for constant-sum games is trivially equivalent to equilibria finding for zero-sum games, this is sufficient for our purposes.

The present paper contains two results interpretable as counterparts to Theorem 5, as they allow to reduce finding equilibria for a large game to finding equilibria in several smaller games; for mixed strategies, this will be a consequence of the main result presented in Subsection 4.3, the corresponding statement for pure strategies is given in the next theorem:

Theorem 7. $1\textsc{Pure}_{n+1} \leq_2 \langle MLPO_2 \rangle^n$.

As we have identified $MLPO_2$ (or $1\textsc{Pure}_2$) as the basic building stone in the degree of discontinuity of finding pure strategies, the following theorem will establish the missing link in the relationship between finding pure strategies and multiple robust divisions:

Theorem 8. $MLPO_2 <_2 \textsc{rDiv}$.

To sum up the results established sofar, we have:

$$\lceil \langle 1\textsc{Pure}_2 \rangle^n \rceil_{n\in\mathbb{N}} \equiv_2 1\textsc{Pure} \equiv_2 \textsc{Pure} <_2 \lceil \langle \textsc{rDiv} \rangle^n \rceil_{n\in\mathbb{N}} \leq_2 \textsc{ZCorr}$$

4.3 Problems reducible to $\lceil \langle \textsc{rDiv} \rangle^n \rceil_{n\in\mathbb{N}}$

The goal of this subsection is to present a way of designing reductions to $\lceil \langle \textsc{rDiv} \rangle^n \rceil_{n\in\mathbb{N}}$, and, in particular, to present a reduction from Nash. This equivalently can be considered as the task to design algorithms for a Type-2-Machine capable of making a finite number of independent queries to an oracle for rDiv. Due to Theorems 7, 8 also oracle calls to $MLPO_n$ are permitted.

We will start by providing a technical lemma similar to Lemma 1. Using the lemma, we can prove that the Fourier-Motzkin-algorithm ([21]) for solving systems of linear inequalities can be executed using continuous (even computable) operations and oracle calls to rDiv.

Lemma 2. *Let F be a multi-valued function defined on a strongly zero-dimensional metrisable space X. Then $F \leq_2 \lceil \langle \textsc{rDiv} \rangle^n \rceil_{n\in\mathbb{N}}$ holds, iff there are k closed sets A_i, $i \leq k$ with $X = \bigcup_{i=1}^{k} A_i$, so that for each $i \leq k$, there is a multi-valued function $G^i \leq_2 \lceil \langle \textsc{rDiv} \rangle^n \rceil_{n\in\mathbb{N}}$ with $\mathrm{dom}(G^i) = X$, so that for each $g^i \in G^i$ there is an $f^i \in F$ with $f^i_{|A_i} = g^i_{|A_i}$.*

Definition 12. *The problem $\textsc{BLinIneq}_{nm}$ asks for a ρ^m-name of a vector v of reals, so that $Av \leq b$ holds in addition to $0 \leq v \leq 1$, given a ρ^{nm}-name for a matrix A and a ρ^n-name for a vector b, provided that a solution exists. For simplicity, we assume that $Av \leq b$ always contains $0 \leq v \leq 1$.* BLinIneq *is the problem without fixed values n, m.*

Theorem 9. $\textsc{BLinIneq} \leq_2 \lceil \langle \textsc{rDiv} \rangle^z \rceil_{z\in\mathbb{N}}$.

Proof. As BLinIneq is expressible as a supremum, it suffices to prove $\text{BLinIneq}_{nm} \leq_2 \lceil\langle \text{rDiv}\rangle^z\rceil_{z\in\mathbb{N}}$ for all $n, m \in \mathbb{N}$. For this, we use induction over m. The case $m = 0$ is trivial, so we assume $\text{BLinIneq}_{n(m-1)} \leq_2 \lceil\langle \text{rDiv}\rangle^z\rceil_{z\in\mathbb{N}}$.

For each $K \subseteq \{1, \ldots, n\}$, abbreviate $K^C := \{1, \ldots, n\} \setminus K$. The set $D_K = \{(A, b) \mid \forall k \in K\ a_{k1} \geq 0 \wedge \forall l \in K^C\ a_{l1} \leq 0\}$ is closed, and the union $\bigcup_{K\subseteq\{1,\ldots,n\}} D_K$ covers the domain of BLinIneq_{nm}. So due to Lemma 2, it is sufficient to show that BLinIneq_{nm} restricted to D_K is reducible to $\lceil\langle \text{rDiv}\rangle^z\rceil_{z\in\mathbb{N}}$ for arbitrary $K \subseteq \{1, \ldots, n\}$. In the next step we assume K to be fixed. With the same argument we can assume $|a_{k1}| \geq |a_{(k+1)1}|$ by renumbering the inequalities for each fixed sequence of increasing first coefficients.

Now we rewrite the given inequalities as $a_{k1}v_1 \leq b_k - \sum_{i=2}^{m} a_{ki}v_i$ for $k \in K$ and $-b_j + \sum_{i=2}^{m} a_{ji}v_i \leq -a_{j1}v_1$ for $j \in K^C$. For each pair $k \in K$, $j \in K^C$, the corresponding inequalities can be multiplied by $-a_{j1}$ respective a_{k1}, and then contracted to:

$$a_{k1}(-b_j + \sum_{i=2}^{m} a_{ji}v_i) \leq -a_{j1}(b_k - \sum_{i=2}^{m} a_{ki}v_i)$$

Every solution to the newly created system of linear inequalities can be extended to a solution to the original system by choosing a suitable value for v_1. Due to the induction assumption, such a solution can be obtained by making oracle calls to $\lceil\langle \text{rDiv}\rangle^z\rceil_{z\in\mathbb{N}}$.

To obtain a solution for v_1, we would like to call

$$v_1 = \max(0, \min(1, \text{op}_1(\text{rDiv}(|b_1 - \sum_{i=2}^{m} a_{1i}v_i|, |a_{11}|), \text{op}_2(|\text{rDiv}(b_2 - \sum_{i=2}^{m} a_{2i}v_i|, |a_{21}|), \ldots$$

with $\text{op}_i = \min$ for $i \in K$ and $\text{op}_i = \max$ else. As the $|a_{k1}|$ are ordered as a decreasing sequence, values that arise arbitrary as result of a division by zero occur deeper inside the nested structure than significant values. While they can influence the actual value for v_1 that is chosen, it still satisfies all inequalities, if this is possible. However, the expression above contains nested calls to rDiv in form of the v_i, $2 \leq i \leq n$.

To solve the problem, one replaces v_2 with the corresponding sequence used to compute it, then v_3, and so on. By moving the max and min operators outside, and unifying all divisions, terms of the form $\text{rDiv}(P, Q)$ remain, where P is a polynomial in a_{ij}, b_j whose degree does not exceed $2n$, and Q is a polynomial in a_{ij} whose degree does not exceed n. These can be evaluated by allowed oracle calls, and the max and min operators are continuous.

As the problem BLinIneq is of considerable interest on its own, we shall note that the converse statement to Theorem 9 is also true:

Theorem 10. $\lceil\langle \text{rDiv}\rangle^z\rceil_{z\in\mathbb{N}} \leq_2 \text{BLinIneq}$.

By adapting [22, Algorithm 3.4] and applying Lemma 2 and Theorem 9 we proceed to prove the main theorem of this subsection. Again, the reasoning directly extends to more than two players.

Theorem 11. $\textsc{Nash} \leq_2 \lceil \langle \textsc{rDiv} \rangle^z \rceil_{z \in \mathbb{N}}$.

Proof. By the same reasoning as above, since NASH is the supremum $\lceil \textsc{Nash}_{nm} \rceil_{n,m \in \mathbb{N}}$, it suffices to show $\textsc{Nash}_{nm} \leq_2 \lceil \langle \textsc{rDiv} \rangle^z \rceil_{z \in \mathbb{N}}$ for arbitrary $n, m \in \mathbb{N}$.

By the best response condition ([22, Proposition 3.1]), a pair of mixed strategies (x, y) is a Nash equilibrium of a game if each pure strategy played with positive probability in x (in y) is a best response against y (against x). This condition can be formalized by noting that the following set is the set of games and their Nash equilibria with support in I, J:

$$\hat{G}_{I,J} = \{(A,B,x,y) \mid j,k \in J \; l \notin J \; (x^T B)_j = (x^T B)_k \geq (x^T B)_l \; y_l = 0 \; i,p \in I \\ q \notin I \; (Ay)_i = (Ay)_p \geq (Ay)_q \; x_q = 0\}$$

The set $\hat{G}_{I,J}$ is closed, and so is its projection $G_{I,J} = \{(A,B) \mid \exists x,y \; (A,B,x,y) \in \hat{G}_{I,J}\}$.

As every game has a Nash equilibrium, the sets $G_{I,J}$ cover the domain of NASH, so we can apply Lemma 2. To recover the Nash equilibrium (x, y) from I, J the corresponding system of linear inequalities has to be solved, which is reducible to $\lceil \langle \textsc{rDiv} \rangle^z \rceil_{z \in \mathbb{N}}$ as established in Theorem 9.

Corollary 2. $\textsc{ZCorr} \equiv_2 \textsc{Corr} \equiv_2 \textsc{ZNash} \equiv_2 \textsc{Nash} \equiv_2 \lceil \langle \textsc{rDiv} \rangle^n \rceil_{n \in \mathbb{N}}$.

The same technique applied in the proof of Theorem 9 can also be used to show that Gaussian Elimination can be reduced to $\lceil \langle \textsc{rDiv} \rangle^n \rceil_{n \in \mathbb{N}}$. This shows that the reduction of Gaussian Elimination to the rank of a matrix given in [23] is strict, taking into consideration Corollary 4.

5 Additional Results

5.1 Nash and *Sep*

To shed further light on the degree of discontinuity of NASH, we will compare it to the problem *Sep* studied in [24].

Definition 13. *$f \in Sep$ holds, iff f is a function from*

$$\{(p,q) \in \mathbb{N}^\mathbb{N} \times \mathbb{N}^\mathbb{N} \mid \forall n,m \in \mathbb{N} \; p(n) \neq q(m)\}$$

to $\mathbb{N}^\mathbb{N}$ satisfying $f(p(n)) = 0$ and $f(q(n)) = 1$ for all $n \in \mathbb{N}$.

The problem *Sep* was shown to be equivalent to finding an infinite path in an infinite binary tree and extending a linear functional from a subspace of a Banach space to the complete space following the Hahn-Banach Theorem. *Sep* can be

reduced to $\{C_1\}$, which is defined through $C_1(p)(n) = 1$, iff there is an $i \in \mathbb{N}$ with $p(i) = n$ and $C_1(p)(n) = 0$ else. The function C_1 has been introduced in [16]. In [25, Theorem 5.5], it was proven that a function is $\sum_2^0$-measurable, iff it is reducible to C_1.

In [24], $\{cf\} \not\leq_2 Sep$ was shown, which can directly to extended to prove $\{f\} \not\leq_2 Sep$ for all discontinuous functions f. In the following, we will prove that $Nash$ is strictly reducible to Sep, thereby obtaining a lower bound for Sep. For this aim, we need the level of Sep.

Theorem 12. $\mathrm{Lev}^2(Sep)$ *does not exist.*

Due to the behaviour of the level under formation of products ([18]) and suprema ([9], [17]), we know $\mathrm{Lev}^2(\textsc{Nash}) = \omega$, where ω is the smallest infinite ordinal. This is sufficient to establish $Sep \not\leq_2 \textsc{Nash}$ by [9, Theorem 5.7].

Theorem 13. $\textsc{rDiv} \leq_2 Sep$.

Theorem 14. $\langle Sep, Sep\rangle \equiv_2 Sep$.

Corollary 3. $\textsc{Nash} <_2 Sep$.

Corollary 4. $\{f\} \not\leq_2 \textsc{Nash}$ *for all discontinuous functions* f.

5.2 The Level of $\textsc{Nash}_{22}$

The simplest non-trivial bi-matrix games, 2×2 games, have already been investigated from a constructive point of view in [26]. Among other results, [26] contains the constructive analogue to the reduction $MLPO_2 \leq_2 \textsc{Nash}_{22}$, and the constructive analogue to determine a subset of $\mathcal{L}_0(\textsc{Nash}_{22}) \setminus \mathcal{L}_1(\textsc{Nash}_{22})$, that is the set where Nash equilibria are continuous. We will produce the TTE-counterpart by investigating the Level of $\textsc{Nash}_{22}$.

Theorem 15. $\mathrm{Lev}(\textsc{Nash}_{22}) = 4$.

References

1. Papadimitriou, C.H.: On the complexity of the parity argument and other inefficient proofs of existence. Journal of Computer and Systems Science **48** (1994) 498–532
2. Chen, X., Deng, X.: Settling the complexity of 2-player nash-equilibrium. Technical Report 134, Electronic Colloquium on Computational Complexity (2005)
3. Weihrauch, K.: Computable Analysis. Springer-Verlag (2000)
4. Gilboa, I., Zemel, E.: Nash and correlated equilibria: Some complexity considerations. Games and Economic Behavior (1989)
5. Gottlob, G., Greco, G., Scarcello, F.: Pure nash equilibria: Hard and easy games. Journal of Artificial Intelligence Research **24** (2005) 357–406
6. Fabrikant, A., Papadimitriou, C., Talwar, K.: The complexity of pure nash equilibria. In: STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing, New York, NY, USA, ACM (2004) 604–612

7. Nash, J.F.: Equilibrium Points in N-Person Games. Proceedings of the National Academy of Sciences **36** (1950) 48–49
8. Aumann, R.: Subjectivity and correlation in randomized strategies. Journal of Mathematical Economics **1** (1974) 67–96
9. Pauly, A.: On the (semi)lattices induced by continuous reducibilities. arXiv:0903.2177v1 (2009)
10. Pauly, A.: How discontinuous is Computing Nash Equilibria? arXiv:0907.1482v1 (2009)
11. Weihrauch, K.: The degrees of discontinuity of some translators between representations of the real numbers. Informatik Berichte 129, FernUniversität Hagen, Hagen (1992)
12. Pauly, A.: Representing measurement results. Journal of Universal Computer Science **15** (2009) 1280–1300
13. Brattka, V., Gherardi, G.: Weihrauch degrees, omniscience principles and weak computability. arXiv:0905.4679v1 (2009)
14. Weihrauch, K.: The TTE-interpretation of three hierarchies of omniscience principles. Informatik Berichte 130, FernUniversität Hagen, Hagen (1992)
15. Mylatz, U.: Vergleich unstetiger Funktionen in der Analysis. Diplomarbeit, Fachbereich Informatik, FernUniversität Hagen (1992)
16. Stein, T.v.: Vergleich nicht konstruktiv lösbarer Probleme in der Analysis. Diplomarbeit, Fachbereich Informatik, FernUniversität Hagen (1989)
17. Hertling, P.: Unstetigkeitsgrade von Funktionen in der effektiven Analysis. PhD thesis, Fernuniversität, Gesamthochschule in Hagen (1996)
18. Pauly, A.: Methoden zum Vergleich der Unstetigkeit von Funktionen. Master's Thesis, FernUniversität Hagen (2007)
19. Bishop, E., Bridges, D.: Constructive Analysis. Springer Verlag, Berlin, Heidelberg (1985)
20. Engelking, R.: General Topology. Heldermann, Berlin (1989)
21. Keler, C.W.: Parallel fourier–motzkin elimination. In: In Proc. 2nd Int. Euro-Par Conference, Springer (1996) 66–71
22. von Stengel, B.: Equilibrium computation for two-player games in strategic and extensive form. In Nisan, N., Roughgarden, T., va Tardos, Vazirani, V., eds.: Algorithmic Game Theory. Cambridge University Press (2007) 53–78
23. Ziegler, M., Brattka, V.: Computability in linear algebra. Theoretical Computer Science **326** (2004) 187–211
24. Gherardi, G., Marcone, A.: How much incomputable is the separable Hahn-Banach Theorem? In Brattka, V., Dillhage, R., Grubba, T., Klutsch, A., eds.: Conference on Computability and Complexity in Analysis. Number 348 in Informatik Berichte, FernUniversität Hagen (2008) 101 – 117
25. Brattka, V.: Effective borel measurability and reducibility of functions. Mathematical Logics Quaterly **51** (2005) 19–44
26. Bridges, D.: First steps in constructive game theory. Mathematical Logic Quaterly **50** (2004) 501–506
27. Kalai, E., Zemel, E.: On the order of eliminating dominated strategies. Discussion Papers 789, Northwestern University, Center for Mathematical Studies in Economics and Management Science (1988)

Towards the Complexity of Riemann Mappings (Extended Abstract)

Robert Rettinger[1]

Department of Mathematics and Computer Science
University of Hagen, Germany

Abstract. We show that under reasonable assumptions there exist Riemann mappings which are as hard as tally ♯-P even in the non-uniform case. More precisely, we show that under a widely accepted conjecture from numerical mathematics there exist single domains with simple, i.e. polynomial time computable, smooth boundary whose Riemann mapping is polynomial time computable if and only if tally ♯-P equals P. Additionally, we give similar results without any assumptions using tally UP instead of ♯-P and show that Riemann mappings of domains with polynomial time computable analytic boundaries are polynomial time computable.

1 Introduction

In this paper we will prove lower bounds on the complexity of Riemann mappings, i.e. conformal mappings of a simply connected domain onto $\mathbb{D}$. Though the existence of such mappings is well known, computability results or even complexity results were unknown for a long time. Despite the fact that constructive proof methods were known for the problem (see [Hen86])) before, a characterization of those domains which have computable Riemann mappings was proven not before [Her99]. In a recent paper, Binder, Braverman and Yampolsky [BBY07] gave sharp bounds on the complexity of the corresponing functor, i.e. the functor which maps domains to their Riemann mappings: This functor is ♯-P complete. (Actually the authors showed that this functor is ♯-P hard and belongs to PSPACE. Using similar techniques, however, even a sharp upper bound of ♯-P can be proven (see [Ret08a]).)

Using the proof techniques of [BBY07] it is not hard to show that this functor remains ♯-P complete even if we restrict the class of domains to those domains which have analytic boundaries. On the other hand, the Riemann mapping of any domain with polynomial time computable analytic boundary can be computed in polynomial time as we will show in Section 4. This underlines that hardness of the functor does not necessarily imply hardness of the mappings themselves and raises the question on the complexity of Riemann mappings in general. In Section 5 we will prove, however, that even the complexity of a single Riemann mapping can be as hard as tally ♯-P under reasonable assumptions. Furthermore we will give a new proof on the (uniform) lower bound of Riemann mappings.

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 209-220
http://drops.dagstuhl.de/opus/volltexte/2009/2272

Our proofs in the non-uniform case will heavily depend on this proof. Besides, this new proof might be of some interest itself as we will use only potential theoretic techniques. Some basic notations of complex analysis, complexity and Type-2 theory are given in the following section.

2 Preliminaries

We denote the set of natural, integer, rational, dyadic, real and complex numbers by $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{Y}$, $\mathbb{R}$ and $\mathbb{C}$, respectively. Here, a dyadic number is a number of the form $i/2^j$ with i and j integers. As we quite often use the symbol i as an index, we denote the imaginary unit $\sqrt{-1}$ by $\hat{\imath}$ instead. The imaginary and real part of a complex number z are denoted by $\Im(z)$ and $\Re(z)$, respectively. We identify $\mathbb{C}$ and $\mathbb{R} \times \mathbb{R}$ in the usual sense and denote the distance between two numbers z, z' by $d(z, z') = |z - z'|$ and the (Hausdorff)-distance between two sets M, N by $d(M, N) = \sup\{d(z', N), d(z'', M) | z' \in M, z'' \in N\}$, where $d(z, M) = d(M, z) = \inf\{d(z, z') | z' \in M\}$. Furthermore let $\mathbb{D}_\varepsilon(z_0)$ denote the open disc of radius ε with center z_0. To simplify notation we use $\mathbb{D}_\varepsilon := \mathbb{D}_\varepsilon(0)$ and $\mathbb{D} = \mathbb{D}_1$.

For an open subset G of $\mathbb{C}$, a function $f : G \to \mathbb{C}$ is called holomorphic iff its complex derivative f' exists throughout G. A holomorphic function f is called conformal on A iff $|f'(z)| > 0$ for all $z \in A$. If f is conformal throughout its domain we simply say that f is conformal.

Beside functions we allow also multi-functions, denoted by $f :\subseteq M \rightrightarrows N$ and $f : M \rightrightarrows N$ for partial and total multi-function, respectively. We will use both notations $f(x) = y$ and $f(x) \ni y$ to denote that y belongs to the image of x under a multi-function f. Furthermore, for a function (or multi-function) $f :\subseteq G \to G'$ and $H \subseteq G$ we denote the restriction of f to H by $f|_H$.

Before turning to Type-2 objects, we will recall some notions of discrete complexity theory. For more details see e.g. [DK00] or [Sip97]. We denote by FP and P the class of polynomial time computable functions $f : \Sigma^* \to \Sigma^*$ and polynomial time decidable languages $L \subseteq \Sigma^*$, respectively. (Σ denotes here and later on a finite alphabet.) Restricting the alphabet Σ to a single symbol, say 0, leads to tally functions and sets. The corresponding classes will be denoted by the subscript 1, i.e. FP_1, P_1, etc.

Beside we will also need the classes $\sharp\mathrm{P}$ and UP. $\sharp\mathrm{P}$ denotes the class of functions $h : \Sigma^* \to \Sigma^*$ so that there exists some $L \in \mathrm{P}$ and polynomial p with $h(u) = |\{v \in \Sigma^* \mid |v| = p(|u|) \wedge (u, v) \in L\}|$ for all $u \in \Sigma^*$. UP denotes the class of languages L so that there exist $\hat{L} \in \mathrm{P}$ and polynomials p with $L = \{v \in \Sigma^* | \exists! u. |u| \leq p(|v|) \wedge (u, v) \in \hat{L}\}$ and $\Sigma^* \setminus L = \{v \in \Sigma^* | \forall u. (u, v) \notin \hat{L}\}$, where $\exists! u$ denotes as usual the fact that there exists exactly one u.

The usual notation of separating complexity classes (or classes in general) is to simply ask for a language, which belongs to the first but not to the second class. Another notion used in literature is separation almost everywhere which can be expressed by the related notion of immune languages (see [DK00]). We will need in this paper a stronger separation notion than the usual one provides. On

the other hand we do not need the full power of almost everywhere separation. We will therefore introduce next a kind of separation, which lies between the usual separation and almost everywhere separation. We define this for function classes over the alphabet $\{0\}$ only.

Definition 1. *Let $\Sigma = \{0\}$. Then a function $f : \Sigma^* \to \Sigma^*$ is called selectively separable by a function $s : \mathbb{N} \to \mathbb{N}$ from a class K iff for every $g \in K$, $g : \Sigma^* \to \Sigma^*$ there exists some $i \in \mathbb{N}$ so that $g(0^{s(i)}) \neq f(0^{s(i)})$.*

Furthermore we say that we can separate two classes K_1 and K_2 selectively ($K_1 \neq_{sel} K_2$) iff for every strictly monotone time constructible function $s : \mathbb{N} \to \mathbb{N}$ there exists a function $f : \Sigma^* \to \Sigma^*$ in K_1 which is selectively separable by s from K_2 or vice versa.

Next, let Σ^{**} denote the set $(\Sigma^*)^{\Sigma^*}$, i.e. the set of total functions $f : \Sigma^* \to \Sigma^*$. We fix some standard tuple function $\langle\cdot\rangle$ on $(\Sigma^{**})^n$ mapping products to Σ^{**}.

To give a natural notion of complexity we extend the Type-2-Turing machine model by allowing some kind of indirect access to the input tapes. Formally we realize this by a new definition of representations and the usage of oracle machines, where oracles are elements of Σ^{**}, i.e. functions rather than languages. Queries to the oracle are here answered by the function value of the string on the oracle tape. An oracle Turing machine M computes a function $f_M :\subseteq \Sigma^{**} \to \Sigma^{**}$ in the following sense: $f_M(\alpha)$ is defined to be β iff for each $w \in \Sigma^*$ the machine M together with the oracle α outputs $\beta(w)$. For fixed α we can define the time complexity as usually. We denote the class of such functions of polynomial time complexity (independently of α) by FP_*. In a similar way even relative computations with respect to some oracles can be defined. Details can be found in [Ret08a].

To introduce complexity on more general Type-2 objects we fix a set of standard representations, i.e. surjective functions $\nu :\subseteq \Sigma^* \to M$ or $\nu :\subseteq \Sigma^{**} \to M$ onto the represented set M, next. A (multi)function $g :\subseteq M \to N$ is then called polynomially time computable if there exists a polynomially time computable realization, i.e. a function $f :\subseteq A \to B$ for some $A, B \in \Sigma^{**}$ so that $f \circ \nu_M = \nu_N \circ g$ on $\mathrm{dom}(\nu_M)$ where ν_M and ν_N denote the standard representation of M, N, respectively. We will denote the corresponding complexity class again by FP_*.

Dyadics will be given by their dual representation, denoting the decimal point by ., i.e. $\nu_{\mathbb{Y}}(w.v) = \nu_{\mathrm{dual}}(w) + \nu_{\mathrm{dual}}(v) \cdot 2^{-|v|}$ and $\nu_{\mathbb{Y}}(-w.v) = -(\nu_{\mathrm{dual}}(w) + \nu_{\mathrm{dual}}(v) \cdot 2^{-|v|})$ for $w, v \in \{0,1\}^*$, $w[0] \neq 0$, where ν_{dual} denotes the dual notation of natural numbers. Complex dyadics are represented by pairs of dyadics: $\nu_{\mathbb{Y}[i]}(\langle d_0, d_1\rangle) = \nu_{\mathbb{Y}}(d_0) + i\nu_{\mathbb{Y}}(d_1)$ for all $d_0, d_1 \in \mathrm{dom}(\nu_{\mathbb{Y}})$.

A real number x is represented as a sequence of dyadics, which converges fast to x, i.e. $\nu_{\mathbb{R}}(f) = x \Leftrightarrow \forall w \in \Sigma^*.|\nu_{\mathbb{Y}}(f(w)) - x| < 2^{-|w|}$ for all $f \in \Sigma^{**}$. Finally, by identifying $\mathbb{C}$ and $\mathbb{R} \times \mathbb{R}$, we get our standard representation of $\mathbb{C}$ by $\nu_{\mathbb{C}} = \nu_{\mathbb{R}\times\mathbb{R}}$.

Now let G, G' be subsets of $\mathbb{C}$. Then the standard representation $\nu_{\mathcal{A}} :\subseteq \Sigma^{**} \to \mathcal{A}$ of a subclass $\mathcal{A}$ of $\mathrm{Cont}(G, G') = \{g : G \to G' \mid g \text{ continuous}\}$ is

defined by

$$g \in \nu_{\mathcal{A}}(f) \Leftrightarrow \forall z \in \nu^{-1}_{\mathbb{Y}[\hat{\iota}]}(G).\forall n \in \mathrm{dom}(\nu_{\mathbb{N}}).|f(\langle n, z\rangle) - g(\nu_{\mathbb{Y}}(z))| < 2^{-n}$$

for all $g \in \mathcal{A}$, $f \in \Sigma^{**}$. The main point of this representation is that we can evaluate functions. For domains there are several different representations. We will use the following representation based on the distance to the boundary.

The representation $\nu^{<}_{\subseteq\mathbb{C}} :\subseteq \Sigma^{**} \rightrightarrows (2^{\mathbb{C}}\backslash\{\mathbb{C}\})$ is defined via a modified distance function. For $f : \mathrm{dom}(\nu_{\mathbb{N}}) \times \mathrm{dom}(\nu_{\mathbb{Y}[\hat{\iota}]}) \to \mathrm{dom}(\nu_{\mathbb{R}})$ let $\nu^{<}_{\subseteq\mathbb{C}}(f) = A \subseteq \mathbb{C}$ iff $3/4 \cdot d(\nu_{\mathbb{Y}[\hat{\iota}]}(z)) - 2^{-\nu_{\mathbb{N}}(n)} < |\nu_{\mathbb{R}}(f(n,z))| < d(\nu_{\mathbb{Y}[\hat{\iota}]}(z))$ for all $n \in \mathrm{dom}(\nu_{\mathbb{N}})$ and all $z \in \nu^{-1}_{\mathbb{Y}[\hat{\iota}]}(A)$, where $d(z) := \inf_{z' \in \partial G} |z - z'|$.

Let A be a represented set. Then we say that a function $f :\subseteq A \to \mathbb{R}^+$ belongs to $\sharp\mathrm{P}_*$, iff there exists a polynomial p and a polynomial time computable function $g :\subseteq A \times \Sigma^* \to \mathbb{R}^+$, so that $f(a) = \sum_{w \in \Sigma^{p(n)}} g(a, w)$ for all $a \in \mathrm{dom}(f)$ (as usually n denotes the length of the input of finite length).

3 Riemann mappings

In this section we will summarize some central results on Riemann mappings.

Theorem 1. *Let G be a bounded simply connected domain. Then for every $z \in G$ and $\phi \in [0; 2\pi]$ there exists a unique conformal mapping $f_G^{z,\phi} : G \to \mathbb{D}$ so that $f_G^{z,\phi}(z) = 0$ and the argument of $(f_G^{z,\phi})'(z)$ is ϕ.*

We will denote these Riemann mappings usually in the way of the above theorem where we omit ϕ and/or z if $\phi = 0$ and/or $z = 0$, or if these parameters are uniquely determined by the context. If G is a Jordan domain, the Riemann mapping continues topologically onto the boundary (see [Pom92] for details). If the boundary γ of G is even analytic, the Riemann mapping continues even holomorphically, which can be easily seen by the reflection principle.

To simplify things, we will restrict ourselves in the sequel to the class of simply connected domains which are contained in the disk $\mathbb{D}_{4/5}$ and contain the disk $\mathbb{D}_{3/5}$. This class of simply connected domains will be denoted by $\mathbb{G}$ in the sequel. For more general classes of simply connected domains the ideas given below can be easily adapted as long as the domains are bounded. This can for example be achieved by the usual square root transformation (see e.g. [Hen86]). Alternatively, the osculation method can be used to reduce the domain. This method converges fast as long as the domain is far away from the unit disc (with respect to the Hausdorff distance). Furthermore we will compute the Riemann mapping on a fixed compact subset of its domain. We can then get the full Riemann mapping by continuation (see e.g. [Ret08b]).

Theorem 2 ([BBY07],see also [Ret08a]). *There exists a function $F_{conf} :\subseteq \mathbb{G} \times \mathbb{D}_{1/2} \to \mathbb{D}$, $F \in FP_*^{\sharp P_*}$, mapping each simply connected domain $G \in \mathbb{G}$ and point $z \in \mathbb{D}_{1/2}$ to $f_G(z)$.*

If we restrict the above function to boundaries, which can be computed in time bounded by a fixed polynomial, then the Riemann mapping can be computed by polynomially time bounded machines with access to a $\sharp P$-oracle.

The proof of the above theorem shows that slight changes in the shape of the domain G will only slightly change the Riemann mapping. We will use this fact e.g. to give a polynomial upper bound for the Riemann mapping for analytic boundaries in Section 4 below.

Corollary 1. *There exists a polynomial p so that for all $G, G' \in \mathbb{G}$ we have: The Riemann mappings f and g of G and G', respectively, determined by $f(0) = g(0) = 0$ and $f'(0) > 0$, $g'(0) > 0$ differ by at most 2^{-n} on $z \in \mathbb{D}_{1/2}$, i.e. $|f(z) - g(z)| \leq 2^{-n}$, if the Hausdorff distance of G and G' is at most $2^{-p(n)}$.*

4 Analytic Boundaries

In this section we will show that for any simply connected Jordan domain G with analytic, polynomial time computable boundary, the Riemann mapping from G is always computable in polynomial time. To prove this we will use a technique based on the Bergman kernel function and orthonormal polynomials.

For given $G \in \mathbb{G}$ and $i \in \mathbb{N}$ let in the sequel p_i denote the i-th orthonormal polynomial, determined by the sequence 1, z, z^2, ... and the Gram-Schmidt algorithm, using the inner product $\langle \cdot, \cdot \rangle$ defined by

$$\langle f, g \rangle = \int\int_G f(z)\overline{g(z)}dxdy$$

for all $f, g \in L^2(G, \mathbb{C})$, where $L^2(G, \mathbb{C})$ denotes the space of square integrable complex functions on G (see e.g. [Gai87]).

Lemma 1. *Let $G \in \mathbb{G}$ be a Jordan domain with its boundary given by a polynomial time computable conformal mapping $\delta : U \to \mathbb{D}$ of an open neighborhood U of $\partial\mathbb{D}$. Then the sequence p_0, p_1 ,... of orthonormal polynomials is computable in polynomial time.*

Notice that orthonormal polynomials can be computed efficiently even in other cases, e.g. in the case of Schwarz-Christoffel mappings. However, it is not known, if the polynomials in this case can be used to compute the Riemann mapping efficiently.

Once we have these orthonormal polynomials for a domain $G \in \mathbb{G}$, we can build a fast algorithm to compute the Riemann mapping upon a well known relation of the Riemann mapping and the Bergmann kernel $K : G \times G \to \mathbb{R}$ (see e.g. [Neh52]).

Theorem 3. *Let $G \in \mathbb{G}$ be a Jordan domain with its boundary given by a polynomial time computable conformal mapping $\delta : U \to \mathbb{D}$ of an open neighborhood U of $\partial\mathbb{D}$. Then f_G is computable in polynomial time, where f_G denotes the uniquely determined Riemann mapping with $f_G(0) = 0$ and $f'_G(0) > 0$.*

Proof: We have the well know relation between the Bergmann kernel function and f_G

$$f'_G(z) = \sqrt{\frac{\pi}{K(0,0)}} \cdot K(z,0).$$

Furthermore the Bergmann kernel function can be expressed by means of the orthonormal polynomials p_0, p_1, ... of G via $K(z,0) = \sum_{j=0}^{\infty} \overline{p_j(0)} \cdot p_j(z)$, where the convergence is uniformly on any compact subset of G. Approximating the Bergmann kernel function by K_n ($n \in \mathbb{N}$), where $K_n(z,0) = \sum_{j=0}^{n-1} \overline{p_j(0)} \cdot p_j(z)$ gives us the the Bieberach polynomials q_i ($i \in \mathbb{N}$), determined by $q_i(z) = \sum_{j=0}^{n} \frac{\int_0^z K_{i-1}(z,\zeta)d\zeta}{\sqrt{K_{i-1}(0,0)}}$ for $i \in \mathbb{N}$.

By Lemma 1, the Bieberach polynomials can be computed in polynomial time. Notice, as the sequence $K_i(0,0)$ converge to $K(0,0) \neq 0$, the $K_i(0,0)$ are bounded away from 0 by a constant for all but finitely many i's. Furthermore we know that for analytic boundaries there exists $M > 0$ and $q \in (0;1)$ so that $|f_G(z) - q_i(z)| < M \cdot q^i$ for all z in say $\mathbb{D}_{1/2}$ and all $i \in \mathbb{N}$ (see [Gai87]). Thus we can compute f_G in polynomial time on $\mathbb{D}_{1/2}$. As continuation of holomorphic mappings can be done in polynomial time (see e.g. [Mül93]) and the fact that f_G can be continued to a whole neighborhood of G by the reflection principle, proves that f_G can be computed in polynomial time throughout G.

□

5 Towards lower bounds

In this section we will first give a new proof for the lower bound on Riemann mappings in the uniform case first shown in [BBY07]. Afterwards we will turn to the non-uniform case. Our proof of the following theorem will use only basic ideas of potential theory.

Theorem 4 ([BBY07]). *If $F : \mathbb{G} \to \mathbb{D}$ with $F(G) = f'_G(0)$ is computable in polynomial time, then every function in $\sharp P_*$ is computable in polynomial time. Especially we have that if F is computable in polynomial time then $\sharp P = FP$.*

Even for restrictions of F to those domains $G \in \mathbb{G}$ whose boundaries are analytic or polygons, this result holds, i.e. if this restrictions are computable in polynomial time then $\sharp P_* = FP_*$.

Proof: The second statement follows from the first one by suitable approximations of general domains by the restricted ones using Corollary 1 above.

To prove the first statement, notice that $f'_G(0) = e^{-u}$, where u is the solution of the Dirichlet problem with boundary values $z \mapsto \log_e(|z|)$ for $z \in \partial G$. We will thus code the behavior of a Turing machine M into such a boundary value problem. In contrast to the construction in [BBY07] we will use the slit map rather than the crescent map, which simplifies things further. Nevertheless, using the ideas below, even the construction of the domains in [BBY07] could be used to prove the above result with potential theoretic ideas only.

Our construction will be based on the slit map (see [Hen86], Chapter 16). Let therefore, for given $\rho \in (0;1)$, $S(\rho)$ denote the straight line from -1 to $-\rho$ and furthermore, by D_ρ the set D_ρ. Then a conformal mapping $h_\rho : D_\rho \to \mathbb{D}$ with $h_\rho(0) = 0$ is given by $h_\rho(z) = (s_\rho(z) - 1 + z)/(s_\rho(z) + 1 - z)$ for all $z \in \mathbb{D} \setminus S_\rho$, where we use the abbreviation $s_\rho(z) = \sqrt{(1+\rho(z))(1+\frac{1}{\rho}z)}$. Furthermore for $h'_\rho(0)$ we have $h'_\rho(0) = \frac{(1+\rho)^2}{4\rho}$ (see [Hen74]).

The main point of giving this map explicitly is that we can easily compute $|h'_\rho(0)|$ and thus $\log|h'_\rho(0)|$.

Claim. There exists a mapping $h : (1/2;1) \to \mathbb{R}$ with $h(\rho) = h'_\rho(0)$ for all $\rho \in (1/2;1)$, which is computable in time $O(n^2)$. Furthermore there exist constants $c_0, c_1, c_2 > 0$, so that $c_0 \cdot (1-\rho)^2 < h'_\rho(0) - 1 < c_1 \cdot (1-\rho)^2$ and $|\log_e(h'_\rho(0)| > c_2 \cdot (1-\rho)^2$ for all $\rho > 3/4$.

We assume now that F is polynomial time computable and $L \in \sharp P_* \setminus FP_*$, $L : A \to \mathbb{R}^+$ for some represented space A. In a first step we will reduce L to a problem in $\sharp P$. Let M be a polynomial time computable Turing machine and q be a polynomial, so that on every input $a \in A$, $n \in \mathbb{N}$ and $w \in \Sigma^*$ with $|w| = q(n)$, M stops in exactly $q(n)$ steps, outputs $o_M(a,n,w) \in \mathbb{Y}^+$ and fulfills

$$|L(a) - \sum_{w \in \Sigma^{q(n)}} o_M(a,n,w)| \leq 2^{-n}.$$

For given input $a \in A$ and a precision 2^{-n}, we are thus asked to compute $L(a)$ up to this precision. As we have to add up at most $2^{q(n)}$ values, we have to compute each of the elements of the above sum up to precision $2^{-(n+q(n))}$ only. This can be done by an addition of $2^{q(n)}$ integers of at most $n + q(n)$ bits each with an appropriate shift afterwards. As this shift is polynomial time computable, we can, by a standard manipulation of M, give a Turing machine N and a polynomial p with the following properties:

1. N stops on input $a \in A$, $n \in \mathbb{N}$ and every $w \in \Sigma^{p(n)}$ in at most $p(n)$ steps with output $o_N(a,n,w) \in \{0,1\}$ and
2. $L(a)$ can be computed from $\hat{L}(a,n) = \sum_{w \in \Sigma^{p(n)}} o_N(a,n,w)$ in polynomial time for every $a \in A$ and $n \in \mathbb{N}$.

Let some $a \in A$, $n \in \mathbb{N}$ with $n > 2$ be given. We construct, using the slit map above, some $G_{a,n} \in \mathbb{G}$ so that for $f_{G_{a,n}}$ with $f_{G_{a,n}}(z) = F(G_{a,n}, z)$ we have

$$\hat{L}(a,n) = \lfloor \log_e(f'_{G_{a,n}}(0)) / \log_e(h'_{1-2^{-m}}(0)) \rfloor,$$

where m is polynomially bounded in n and will be chosen later on. Thus, if F is polynomial time computable, clearly L is polynomial time computable too. We will give here a slightly more general result than necessary by introducing an additional parameter ε. We will need this general result in the proof of Theorems 5 and 6 later on. For given $v \in \Sigma^{p(n)}$ and $\varepsilon \in (0; 2\pi)$ let $\phi_v^\varepsilon := \varepsilon 2^{-(p(n))}$.

$(\nu_{\text{dual}}(1v) - 2^{p(n)})$. The values ϕ_v^ε of all such v are in the interval $[0;\varepsilon]$ and for different $u, v \in \Sigma^{p(n)}$ we have $|\phi_v - \phi_u| \geq \varepsilon \cdot 2^{-p(n)}$. Now let

$$G_{a,n}^\varepsilon = \bigcup_{\substack{v \in \Sigma^{p(n)} \\ o_N(a,n,v)=1}} e^{-i\phi_v^\varepsilon} \cdot D_{1-2^{-m_\varepsilon}}.$$

Notice that for fixed $\varepsilon \in \mathbb{Y} \cap (0;1)$, the function $H : A \times \mathbb{N} \to \mathbb{G}$ with $H(a,n) = G_{a,n}^\varepsilon$ for all $a \in A$, $n \in \mathbb{N}$, is polynomial time computable. By the above discussion it remains to show that we can compute $\hat{L}(a,n)$ efficiently from $f'_{G_{a,n}^\varepsilon}(0)$, because then L can be computed in polynomial time in contradiction to our assumption.

Claim. Let $\varepsilon \in (0;2\pi)$, $n \in \mathbb{N}$ be given and $m_\varepsilon(n) = \lceil 4 \cdot p(n) + \log(1/\varepsilon) + \log(c_2) \rceil + 15$. Then $\hat{L}(a,n) = \lfloor \log_e(f'_{G_{a,n}^\varepsilon}(0)) / \log_e(h'_{1-2^{-m_\varepsilon(n)}}(0)) \rfloor$ for all $n \in \mathbb{N}$.

As we consider a fixed $n \in \mathbb{N}$ in the sequel we will write m_ε instead of $m_\varepsilon(n)$, for short. The main work in showing the above equation is, to bound the cross terms introduced to the Riemann mapping when combining the different slit maps. To this end we will use the relation between the Brownian motion and potentials as already considered in the last section. Let therefore, for given $G \in \mathbb{G}$, $z \in G$ and $Z \subseteq \partial G$, $p_G(Z|z)$ denote the probability to end up in Z when we start in z. To be more precise, let for $z \in G$, $B_G^t(z)$ denote the Brownian motion process, which starts in z. Furthermore let T be the first time $B_G^t(z)$ hits the boundary ∂G. Then for given continuous or piecewise constant and bounded values $v(x) \in \mathbb{R}$ (for boundary points $x \in \partial G$) we know that $f : G \to \mathbb{R}$, $f(z) = E(v(B_G^T(z)))$, is the unique solution to the corresponding Dirichlet problem (where $E(X)$ denotes the expectation of the random variable X). Furthermore $p_G(Z|z)$ is the expectation $p_G(Z|z) = E(\chi_Z(B_G^T(z)))$. To simplify things we will in addition use the notation $p_G(Z|z \rightsquigarrow Z')$ meaning the probability to end up in $Z \subseteq \partial G$, starting in $z \in G$ and visiting at least once a point in $Z' \subseteq G$.

A main tool in bounding the probabilities is the Poisson formula

$$u(z) = \frac{1}{2\pi} \cdot \int_{\partial \mathbb{D}} v(y) \cdot \frac{1 - |z|^2}{|z-y|^2} dy$$

which gives an explicit solution to the Dirichlet problem if $G = \mathbb{D}$. Unfortunately, however, G_w^ε is likely to be not $\mathbb{D}$ (unless $\hat{L}(n,a) = 0$). By the following result we can nevertheless use Poissons formula, where we use the abbreviation $\partial_{m_\varepsilon} := \partial\mathbb{D} \cap \mathbb{D}_{2^{-m_\varepsilon}}(-1)$:

Claim. For all $z \in S_{1-2^{-m_\varepsilon}}$ we have

$$p_{\mathbb{D}}(\partial_{m_\varepsilon}|z) = \frac{1}{2\pi} \cdot \int_{\partial \mathbb{D}} \chi_{\partial_{m_\varepsilon}}(y) \cdot \frac{1 - |z|^2}{|z-y|^2} dy > \frac{3}{4}.$$

Let $u : G^{\varepsilon}_{a,n} \to \mathbb{R}$ be the solution of the Dirichlet problem with boundary values $v(x) = \log(|x|)$ for $x \in \partial G^{\varepsilon}_{a,n}$, especially we have $f'_{G^{\varepsilon}_{a,n}}(0) = e^{-u(0)}$. We will bound the difference of $u(0)$ and $\hat{L}(n,a) \cdot |\log_e(h'_{1-2^{-m_\varepsilon}}(0))|$ accordingly. Notice that $\hat{L}(n,a)$ is the number of slits in $G^{\varepsilon}_{a,n}$. Each slit, say at angle $\phi = \phi^{\varepsilon}_v$, taken alone, adds a value $\log_e(h'_{1-2^{-m_\varepsilon}}(0))$ to u_0. However, not every path in the Brownian motion, which ends at the slit $e^{\hat{\imath}\phi} \cdot S_{2^{-m_\varepsilon}}$ on $(e^{\hat{\imath}\phi} \cdot D_{2^{-m_\varepsilon}})$, will also end there on $G^{\varepsilon}_{a,n}$, because it might hit another slit in between. (As $\log_e(1) = 0$ only the hits of slits are counted.) To simplify things we will use the abbreviation $S_\phi := e^{\hat{\imath}\phi} \cdot S_{2^{-m_\varepsilon}}$ and $D_\phi = \mathbb{D} \setminus S_\phi$ in the sequel.

As $p_{G^{\varepsilon}_{a,n}}(Z|z \rightsquigarrow Z') \leq p_{G^{\varepsilon}_{a,n}}(Z'|z) \cdot \sup_{z' \in Z'} p_{G^{\varepsilon}_{w}}(Z|z')$ for all $z, z' \in G^{\varepsilon}_{a,n}$ and $Z, Z' \subseteq G^{\varepsilon}_{a,n}$, we can bound the difference $|u(0) - \hat{L}(n,a) \cdot \log_e(h'_{1-2^{-m_\varepsilon}}(0))|$ by the sum of the probabilities to miss a slit $S_{\phi^{\varepsilon}_v}$ in $G^{\varepsilon}_{a,n}$, because of hitting a slit $S_{\phi^{\varepsilon}_u}$ first. For given $z \in S_{\phi^{\varepsilon}_u}$ we have

$$p_{G^{\varepsilon}_{a,n}}(S_{\phi^{\varepsilon}_v}|z) \leq \frac{4}{3}\frac{1}{2\pi} \cdot \int_{\partial\mathbb{D}} \chi_{\partial_{m_\varepsilon}}(y) \cdot \frac{1-|z|^2}{|z-y|^2} dy.$$

As $1 - |z|^2 \leq 2^{-m_\varepsilon+1} - 2^{-2m_\varepsilon} \leq 2^{-m_\varepsilon+1}$ and $|z - x|^2 \geq ((1/\pi) \cdot (\varepsilon \cdot 2^{-p(n)} - 2 \cdot 2^{-m_\varepsilon}))^2 \geq 2^{-2p(n)+3}$ for all $z \in S_{\phi^{\varepsilon}_u}$ and $x \in \partial_{m_\varepsilon}$, we get

$$p_{G^{\varepsilon}_{a,n}}(S_{\phi^{\varepsilon}_v}|z) \leq \frac{4}{3} \cdot (2 \cdot \pi \cdot 2^{-m_\varepsilon+1}) \cdot (2^{-m_\varepsilon+1}/(\varepsilon \cdot 2^{-(2p(n)+3)})).$$

Furthermore we have $p_{G^{\varepsilon}_{w}}(S_{\phi_u}|0) \leq \frac{4}{3} \cdot 2\pi \cdot 2^{-m_\varepsilon+1}$ thus giving

$$|u(0) - \hat{L}(n,w) \cdot \log_e(h'_{1-2^{-m_\varepsilon}}(0))| \leq \frac{2^{2p(n)} \cdot 2^{-2(m_\varepsilon-5)} \cdot 2^{-m_\varepsilon+1}}{(\varepsilon \cdot 2^{-2(p(n)+3)})}.$$

Notice that by the above cross-terms the probability to hit a slit is decreased, i.e. $u(0) \geq \hat{L}(n,a) \cdot \log_e(h'_{1-2^{-m_\varepsilon}}(0))$. As $|\log_e(h'_{1-2^{-m_\varepsilon}}(0))| > c_2 \cdot 2^{-2 \cdot m_\varepsilon}$ we get

$$\hat{L}(n,a) + 1/2 \geq u(0)/\log_e(h'_{1-2^{-m_\varepsilon}}(0)) \geq \hat{L}(n,a)$$

which proves the theorem. □

The previous theorem states that we cannot compute the Riemann mappings for all G in a uniform way. As shown in Section 4, this does not mean that the Riemann mapping for each $G \in \mathbb{G}$ cannot be computed in polynomial time. This raises the question, wether there exists a single domain G in $\mathbb{G}$, which is polynomial time computable, but the Riemann mapping f_G of G is not polynomial time computable under reasonable assumptions. We restrict ourselves to computing this map on a small neighborhood of 0 and we will answer this question affirmative under the following conditions:

1. if $UP_1 \neq_{sel} FP_1$ or
2. if $\sharp P_1 \neq_{sel} FP_1$ and in addition Conjecture 1 on the existence of Schwarz-Christoffel mappings holds.

Notice that any such result is involved with tally classes, i.e. classes of languages in $\{0\}^*$ rather than languages over alphabets with more symbols. This stems from the fact that we can compute the Riemann mapping on any compact subset of G, say to precision 2^{-n}, by asking a single question to a $\sharp P$ oracle. (Actually we need a polynomial number of such queries. However, these can be coded into a single query of a modified oracle.)

We will start to prove the existence of the domain G under the first condition.

Theorem 5. *If $UP_1 \neq_{sel} FP_1$ then there exists a polynomial time computable domain $G \in \mathbb{G}$, so that f_G is not polynomial time computable.*

Proof: Let L be a function in UP_1, which is selectively separated from FP_1. L is obviously a function in $\sharp P$ with values in $\{0,1\}$. Thus we can use all the notations of the proof of Theorem 4 also here. Notice, however that we start already with some function in $\sharp P$ and thus we do not have to reduce to such a function first. So we use L instead of $\hat{L}$ here and furthermore the parameter a used in the proof of Theorem 4 does not appear here. Especially, let N be a Turing machine and p be a polynomial with the following properties:

1. N stops on input w and every $v \in \Sigma^{p(|w|)}$ in at most $p(|w|)$ steps with output $o_N(w,v) \in \{0,1\}$ and
2. $L(w) = \sum_{v \in \Sigma^{p(|w|)}} o_N(w,v)$

for every $w \in \Sigma^*$.

The main idea of the proof is as follows. Using the techniques of the proof of Theorem 4 above, we construct a domain $G \in \mathbb{G}$ in steps i, where we determine in each step a domain G_i, a conformal mapping $f_i : G_i \to \mathbb{D}$ and a natural number n_i so that (1) we can compute $L(0^{n_i})$ from $f'_{G_i}(0)$ in polynomial time and (2) G_i differs from G_{i+1} (in the Hausdorff metric) by at most 2^{-n_i+1}.

Thus, by defining the n_i large enough, we can ensure that there exists a $G \in \mathbb{G}$ with $d_H(G_i, G) \leq 2^{-n_i+2}$ and the difference of $f'_G(0)$ and $f'_{G_i}(0)$ is small enough, so that we can still compute $L(0^{n_i})$ from $f'_G(0)$ in polynomial time. To this end we have simply to ensure that $n_{i+1} > q(n_i)$, where the polynomial is given by Corollary 1. Once we have constructed G_i, f_i and n_i with this property we proceed in step $i+1$ as follows: First we find some n'_{i+1} so that we can compute f_i in polynomial time for all inputs of length at least n'_{i+1}. Then we choose n_{i+1} to be the maximum of $q(n_i)$ and n'_{i+1}. Following the idea of the proof of Theorem 4 we can compute a domain $G^1_{n_{i+1}}$ so that $L(0^{n_{i+1}})$ can be computed from $f'_{G^1_{n_{i+1}}}(0)$ in polynomial time. If we finally fix G_{i+1} to be $G_{i+1} = f_i^{-1}(G^1_{n_{i+1}})$ and $f_{i+1} = f_{G^1_{n_{i+1}}} \circ f_i$, we can still compute the value $L(0^{n_{i+1}}$ from $f'_{i+1}(0)$ in polynomial time: Simply divide $f'_{i+1}(0)$ by $f'_i(0)$ to get $f'_{G^1_{n_{i+1}}}(0)$. As $f'_i(0)$ can be computed in polynomial time by choice of n'_{i+1}, we are done. Notice that we can define n'_{i+1} because f_i is a composition of Riemann mappings $f_{G'}$ for slit-maps G' as $L(0^{n_i}) \in \{0,1\}$ for all i.

□

Using the ideas of the previous proof, we can also show the existence of G in the second case. Before giving this result we need to specify the conjecture on the existence of efficient algorithms for Schwarz-Christoffel mappings.

We will consider polygons given by the list of their vertices, which we assume to be complex dyadics. Furthermore we restrict ourselves to polygons which are the boundary of some domain in $\mathbb{G}$. Let Polygon be the set of the polygons restricted in such a way. Furthermore we introduce a standard representation $\nu_{Polygon} :\subseteq \Sigma^* \to$ Polygon by simply taking $\langle d_1, ..., d_n\rangle$ to be a $\nu_{Polygon}$-name for a polygon γ, iff the d_i's are names of the complex dyadic vertices of γ in counter clockwise order. Finally we will not distinguish between polygons γ and the corresponding domains with boundary γ, which we denote by $I(\gamma)$.

By the well known Schwarz-Christoffel formula (see e.g. [DT02]), f_γ is determined by

$$f_\gamma^{-1}(z) = C \cdot \int_0^z (1 - x/z_k)^{\alpha_k - 1} dx$$

where z_k are the images $f_\gamma(w_i)$ of the vertices w_i of γ, $\alpha_k\pi$ are the interior angles of γ and C is a positive real number. We can compute the integral above quite efficiently once we know C and z_1,...,z_k. The determination of these parameters is called the parameter problem of the Schwarz-Christoffel mapping. The usual way to solve this problem in numerics is to consider the non-linear system composed of the side-length conditions and a transformation to get an unconstrained system, i.e. to get rid of the condition on the ordering of the vertices and images of the vertices. Then this system of equations is solved by well known methods. There exist however examples, where this leads to local solutions which are not solutions for the Schwarz-Christoffel parameter problem. We do not know wether these methods are applicable to our problem. Notice however that in contrast to the general parameter problem, the polygons used in the proof below, can be chosen up to some degree, thus probably simplifying the problem.

There are other methods to solve the parameter problem, for example by deriving conditions on the so called cross ratios (see [DV98]). This seemingly leads to equations, which might be solvable efficiently in general.

Unfortunately, however, there does not exist an analysis of these methods, which can be translated to the rigorous definition of complexity we need. Thus we will give here the result we need, and which is claimed in a much stronger sense in numerical analysis, as a conjecture.

Conjecture 1. There exists a polynomial p and a computable function F_{SC} : Polygon $\times \mathbb{D} \to \mathbb{N} \times \mathbb{D}$ so that for each $\gamma \in$ Polygon there exist n_γ so that $F_{SC}(\gamma, z) = (n_\gamma, f_\gamma^{-1}(z))$ for all $z \in \mathbb{D}$, and F_{SC} is computable in time $O(n_\gamma \cdot p)$.

Here f_γ denotes the Riemann mapping with $f_\gamma(0) = 0$ and $f'_\gamma(0) > 0$.

Using a similar proof technique as in the first case we can show the existence of the domain G also in the second case:

Theorem 6. *There exists a polynomial time computable (Jordan) domain $G \in \mathbb{G}$, so that f_G is not polynomial time computable if $\sharp P_1 \neq_{sel} FP_1$ and Conjecture 1 holds.*

6 Remarks

We have proved lower bounds on the complexity of Riemann mappings even in the computational case. As shown, a proof that the parameter problem of Schwarz Christoffel mappings is polynomial time computable, which is undoubtedly interesting on its own, would improve upon the bound we have given. Another interesting question is, wether it is possible to prove such a result for more general separation assumptions than the selective separation we have used.

Finally, a more general connection between orhtonormal polynomials and f_G for domains with non-analytic boundaries would be interesting. (Such results exist, but the corresponding speed of convergence for the Bieberach polynomials is too slow to be reasonable applicable, see e.g. [Gai87].)

For domains with analytic boundaries, a polynomial time algorithm for the Riemann mapping can be also deduced differently, by different relation of the Riemann mapping and orthonormal polynomials via a theorem by Carlemann (see [Gai87] for more details).

References

[BBY07] I. Binder, M. Braverman, and M. Yampolsky, *On computational complexity of Riemann mapping*, Arkiv for Matematik (2007), to appear.

[DK00] D.-Z. Du and K.-I Ko, *Theory of computational complexity*, Wiley-Interscience Series in Discrete Mathematics and Optimization, 2000.

[DT02] T. A. Driscoll and L. N. Trefethen, *Schwarz–Christoffel mapping*, Cambridge Monographs on Applied and Computational Mathematics, vol. 8, Cambridge University Press, Cambridge, UK, 2002.

[DV98] T. A. Driscoll and S. A. Vavasis, *Numerical conformal mapping using cross-ratios and Delaunay triangulation*, SIAM Journal on Scientific and Statistical Computing **19** (1998).

[Gai87] D. Gaier, *Lectures on complex approximation*, Birkhäuser, 1987.

[Hen74] P. Henrici, *Applied and complex analysis. Vol. 1*, Pure and Applied Mathematics, John Wiley & Sons, New York, 1974.

[Hen86] ———, *Applied and computational complex analysis. Vol. 3*, Pure and Applied Mathematics, John Wiley & Sons, New York, 1986.

[Her99] P. Hertling, *An effective Riemann Mapping Theorem*, Theoretical Computer Science **219** (1999), 225–265.

[Mül93] N. Th. Müller, *Polynomial time computation of Taylor series*, JAIIO - Panel, Part 2, 1993, Buenos Aires, 1993, pp. 259–281.

[Neh52] Z. Nehari, *Conformal mapping*, Dover Publications, New York, 1952.

[Pom92] Ch. Pommerenke, *Boundary behaviour of conformal maps*, Springer-Verlag, 1992.

[Ret08a] R. Rettinger, *Computability and complexity aspects of univariate complex analysis – habilitation thesis*, 2008.

[Ret08b] ———, *On the continuation of holomorphic functions*, CCA, 2008.

[Sip97] M. Sipser, *Introduction to the theory of computation*, PWS Publishing, 1997.

On the Computability of Rectifiable Simple Curve ⋆ (Extended Abstract)

Robert Rettinger[1] and Xizhong Zheng[2,3]⋆⋆

[1] Lehrgebiet Algorithmen und Komplexität
FernUniversität Hagen, 58084 Hagen, Germany
[2] Department of Computer Science and Mathematics
Arcadia University, Glenside, PA 19038, USA
[3] Departments of Mathematics and Computer Science
Jiangsu University, Zhenjiang 212013, China

Abstract. In mathematics curves are defined as the images of continuous real functions defined on closed intervals and these continuous functions are called parameterizations of the corresponding curves. If only simple curves of finite lengths are considered, then parameterizations can be restricted to the injective continuous functions or even to the continuous length-normalized parameterizations. In addition, a plane curve can also be considered as a connected one-dimensional compact subset of points. By corresponding effectivizations, we will introduce in this paper four versions of computable curves and show that they are all different. More interestingly, we show also that four classes of computable curves cover even different sets of points.

Keywords: Computable Curve, Simple Curve, Rectifiable Curve, Point Separability

1 Introduction

In computable analysis, we are mainly interested in the computability over various continuous structures. One realistic approach to this kind of computability is the Turing-machine-based bit model (see [7, 11, 2]). In this model, real numbers are represented by effectively convergent sequences of rational numbers and these sequences are called *names* of the real numbers. Here a sequence (x_n) converges effectively means that $|x_n - x_{n+1}| \leq 2^{-n}$ for all n. A real number x is computable if it has a computable name. Furthermore, a real function f is computable if there is a Turing machine which transfers each name of a real number x in the domain of f into a name of $f(x)$. By the same principle, computability of other mathematical objects can be defined by introducing proper "naming

⋆ This work is supported by DFG (446 CHV 113/266/0-1) and NSFC (10420130638)
⋆⋆ Corresponding author. email: ZhengX@Arcadia.edu

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 221-232
http://drops.dagstuhl.de/opus/volltexte/2009/2273

systems". For example, the computability of subsets of the Euclidean space [1], of semi-continuous functions [12], of functional spaces [13] are all defined in this way. All these computability of mathematical objects are achieved by a kind of "effectivization" of the classic mathematic definitions.

Particularly, we can introduce the computability of curves in this way too. We consider the plane curves in this paper only. The curves of higher dimensions can be discussed in essentially the same way. Notice that, there are different mathematical approaches to define curves. For example, a curve can be defined as a connected and one-dimensional compact subset. Based on this approach we can define the computable curves by means of the computability of compact subsets of Euclidean space ([1]). Physically, a curve records the trace of a particle motion. If the particle moves according to some algorithmically definable laws, its trace should be regarded as computable. In mathematical terms, a curve is the range of a continuous function defined on a closed interval and this function is called a parametrization of the curve. If a curve has a computable parametrization, then it should be naturally considered as a computable curve (see e.g., [4, 5]).

However, the parametrization of a curve may have various extra properties, particulary if we consider the curves which do not intersect itself and have finite length. Normally, a parameterization of a plane curve C is just a continuous function $f : [0, 1] \to \mathbb{R}^2$. This paramaterization possibly traces some segment of the curve several times. That is, the parameterization f retraces the curve, or it is retraceable. If a curve does not intersect itself, then, by a classic theorem in analysis, it has always an injective parameterization (with possibly exemption at the endpoints of the interval). In addition, if C has a finite length, then it has even an arc-length normalized parameterization. Here a parametrization f is called *arc-length normalized*, if the curve-segment $f([0, t])$ has a length proportional to the parameter t, for any $t \in [0, 1]$.

In this paper we will introduce four versions of computable curves by effectivizing above four mathematical approaches to the curves. We will see that these four versions of computability about curves are all different. The difference of the computability of curves introduced by computable parameterizations and computable injective parameterizations was already shown by Gu, Lutz and Mayordomo in a recent paper [5]. The separations of four versions of computable curves shown in this paper hold actually in a more stronger sense. Namely, the point sets covered by four classes of computable curves are also different. In other words, different versions of computable curves can be separated by points and then they are "point-separable" (see definition in Section 4).

Our paper is organized as follows. In Section 2 we will briefly recall some basic notions related to curves, give the precise definition of computable curves and then show some basic properties of computable curves. In Section 3, we show a technical lemma which will be used in the proof of the main theorem. In Section 4 we prove our main results that four classes of computable curves in different sense are point-separable.

2 Computable Curves

In mathematics, a *plane curve* is defined as a subset $C \subseteq \mathbb{R}^2$ which is the range of a continuous function $f : [0;1] \to \mathbb{R}^2$, i.e., $C = \text{range}(f)$. This continuous function f is then called a *parametrization* of C. Here we use w.l.o.g. the unit interval $[0,1]$ instead of more general closed intervals of the form $[a,b]$. Obviously, any curve has infinitely many parameterizations. Geometrically, a curve records the path of a particle movement on the plane. If the particle never visit one position more than once, in other words, if the curve does not intersect itself (with possible exemption of end points), then the curve is called *simple*. A classical mathematical theorem asserts that, any simple curve has a parameterization $f : [0;1] \to \mathbb{R}^2$ which is injective on $[0;1)$. If a curve C has an injective parameterization f (meaning injective on the interval $[0;1)$) and fulfills in addition $f(0) = f(1)$, then the curve C is called *closed*.

For the simple curves, their lengths can be defined by approximation of the lengths of polygons which converges to the curves according to Jordan [6]. More precisely, Let C be a simple curve and let $f : [0;1] \to \mathbb{R}^2$ be an injective continuous parameterization of C. The *length* L of the curve C is then defined by

$$L := \sup \sum_{i=0}^{n} |f(a_i) - f(a_{i+1})|.$$

where $|f(a_i) - f(a_{i+1})|$ is the length of the straight line connecting the points $f(a_i)$ and $f(a_{i+1})$ and the supremum is taken over all possible partitions $0 = a_0 < a_1 < ... < a_n = 1$. The length of a curve C is denoted by $l(C) := L$. A curve of a finite length is traditionally called *rectifiable*. Not every curve has a finite length. Some curves can even fill whole space like Peano curves (see e.g. [3]). In this paper we are mainly interested in the simple rectifiable curves.

It is well known in analysis that every simple, rectifiable curve has also a length-normalized parameterization. Here a length-normalized (or simply normalized) parameterization of a curve C is an injective continuous function $f : [0,1] \to \mathbb{R}^2$ such that the curve segment $f([0,t])$ has the length $t \cdot l(C)$ for all $t \in [0,1]$. Thus, a simple rectifiable curve can have three different kind of parameterizations—continuous, injective continuous and normalized. In addition, a curve can also be defined as a connected one-dimensional compact point set. By effectivizing these approaches to curves, we can introduce four different versions of computable curves.

Remember that a real function $f : [0;1] \to \mathbb{R}$ is computable if there is a Turing machine M which transfers any name of $x \in [0,1]$ to a name of $f(x)$. Equivalently, f is computable iff there is a computable sequence $(p_n)_{n\in\mathbb{N}}$ of computable rational polygon functions which converges uniformly and effectively to f (see [10]). Naturally, a function $f : [0;1] \to \mathbb{R}^n$ is computable if all of its component functions are computable, or equivalently, if there is a Turing machine M which transfers any name of $x \in [0,1]$ into a tuple $(\alpha_1, \cdots, \alpha_n)$ of names of $f_1(x), \cdots, f_n(x)$ respectively, where $f(x) = (f_1(x), \cdots, f_n(x))$. In this case, we simply say that M computes the function f.

Now we call define the computable curves as follows.

Definition 1. *Let C be a simple plane curve.*

1. *C is called* K-computable *if there is a computable sequence (Q_n) of finite sets of rational neighborhoods such that*

$$C \subseteq \bigcup Q_n \text{ and } d_H\left(\bigcup Q_n,\ C\right) < 2^{-n} \tag{1}$$

   for all $n \in \mathbb{N}$, where d_H denotes the Hausdorff distance.
2. *C is called* R-computable *if there is a computable function $f : [0;1] \to \mathbb{R}^2$ such that* $\mathrm{range}(f) = C$.
3. *C is called* M-computable *if there is a computable function $f : [0;1] \to \mathbb{R}^2$ which is injective on $[0;1)$ such that* $\mathrm{range}(f) = C$.
4. *C is called* N-computable *if C has a computable parameterization $f : [0;1] \to \mathbb{R}^2$ such that the length of the curve segment $f([0,t])$ is equal to $t \cdot l(C)$ for all $t \in [0,1]$.*

In the item 1 of the definition, the finite sets Q_n of rational neighborhoods are also called compact covers of the curve C. The second part of the condition (1) means that the maximal distance from C to bordering of the compact cover Q_n is bounded by 2^{-n}. In this paper, an ε-neighborhood $V_\varepsilon(a,b)$ of a point with Cartesian coordinates (a,b) means the rectangle bounded by the lines $x = a \pm \varepsilon$ and $y = b \pm \varepsilon$. A neighborhood $V_\varepsilon(a,b)$ is called rational if a, b and ε are all rational numbers. The letter K of the K-computability comes from the German word *K*ompakt (compact) due to the compact coverings.

In the item 2, the letter R stands for *R*etracable because the parametrization f of a R-computable curve C can retrace the curve C. Namely, there could be some disjoint subintervals $I_1, I_2 \subset [0,1]$ such that $f(I_1) = f(I_2)$. In this case, f traces some pieces of C more than once, or f is retraceable.

If the paramaterization of a curve C is injective, then C records the movement of a particle with a monotone direction. The letter M in M-computability stands for *M*onotonically directed movement. Notice that, in this paper, we call a parameterization $f : [0,1] \to \mathbb{R}^2$ injective even if it is only injective on $[0;1)$ and does not exclude the possible case $f(0) = f(1)$. This should not cause essential confusions.

Finally, if a parameterization $f : [0,1] \to \mathbb{R}^2$ satisfies the condition that the length of the curve segment $f([0,t])$ is proportional to t, then it is called arc-length normalized. Thus, N-computability stands for *N*ormalized parameterization.

It is well know that not every curve has a finite length. For example, the famous Peano curve can even fill the two-dimensional plan (see e.g., Peano [9]) and has an infinite length. From the definition 1, an N-computable curve has always a finite length. However, the next theorem shows that an M-computable curve does not necessarily have an finite length any more. This distinguishes the N-computability from other three versions of computability immediately.

Theorem 1. *There is an M-computable curve C which has an infinite length.*

Proof. (Sketch) We can construct firstly a computable sequence (p_n) of rational polygons such that distance between p_n and p_{n+1} is bounded by 2^{-n} and p_{n+1} has doubled length of p_n by introducing many small zigzags, for all n. Then, the limit $p := \lim p_n$ is a curve of infinite length. Corresponding to each polygon p_n we can define a computable injective function $f_n : [0,1] \to \mathbb{R}^2$ as a parameterization of p_n, and in addition, we can require that $|f_n(t) - f_{n+1}(t)| \leq 2^{-n}$ is satisfied for all $n \in \mathbb{N}$ and $t \in [0,1]$. Therefore, the limit function $f := \lim f_n$ is an injective computable parameterization of the curve p and hence p is an M-computable curve with an infinite length.

Although a computable curve may have an infinite length, computable rectifiable curves seem more interesting and more important. In this paper we will mainly focus only on the computable curves of finite length and we denote by $\mathbb{C}_K, \mathbb{C}_R, \mathbb{C}_M$ and $\mathbb{C}_N$ the classes of all K-, R-, M- and N-computable rectifiable curves, respectively. By definition, it is straightforward that we have the following relationship between these four versions of computable curves.

Theorem 2. $\mathbb{C}_N \subseteq \mathbb{C}_M \subseteq \mathbb{C}_R \subseteq \mathbb{C}_K$.

Actually we will see that all these four versions of computability of curves are different and hence all the subset relations above are proper.

In the paper [5], Gu, Lutz and Mayordomo have shown that any rectifiable R-computable curve has a left computable length, where a real number x is left computable or computably enumerable (c.e. for short) if there is an increasing computable sequence (x_n) of rational numbers which converges to x. This can be strengthen further to the K-computable curves as follows.

Theorem 3. *Any rectifiable K-computable curve has a left computable length.*

Proof. (Sketch) If C is a rectifiable K-computable curve, then there is a computable sequence (Q_n) of rational compact covers of C such that $d_H\left(\bigcup Q_n,\ C\right) < 2^{-n}$ and Q_n consists of rational neighborhoods. In each cover $\bigcup Q_n$ we can find the shortest polygon which straight through the whole area. This polygon is called a "diameter polygon" of the cover Q_n. The length l_n of this polygon is a lower bound of the length of C (possible with the error $\leq 2^{-n+1}$ because of the endpoints). Since C has a finite length l, the limit $l = \lim l_n$ is left computable because $l_n - 2^{-n+1} \leq l$ for all n.

By Theorems 2 and 3, any rectifiable R-, M- and N-computable curve has left computable length. Ko [8] constructed "monster curve" which is M-computable (even in polynomial time) with a non-computable length. The fact that the length of an M-computable curve is not necessarily computable follows also from the next result.

Theorem 4. *If C is a K-computable curve with a computable length, then C must be N-computable.*

Proof. Suppose that C is K-computable whose length l is a computable real number. Then there is a computable sequence (Q_n) of rational compact covers of C and a computable sequence (l_n) of rational numbers which converges to l effectively. Let q_n be the length of the "diameter polygon" of the area $\bigcup Q_n$.

For each $n \in \mathbb{N}$, we can find a sufficiently large index s_n such that $|q_{s_n} - l_{s_n}| \leq 2^{-n}$. Such an index s_n exists because both sequences (q_s) and (l_s) converge to the same limit $l(C)$. Suppose that p_n is a rational "diameter polygon" of the area $\bigcup Q_{s_n}$ and let f_n be the length-normalized parameterization of p_n. Then (f_n) is a computable sequence of computable functions which converges effectively to a computable function f. This limit function f is a length normalized parameterization of C. Therefore, the curve C is N-computable.

Notice that, if we consider only the curves of computable length, then the K-, R-, M- and N-computability of curves are equivalent. Now let C be an M-computable rectifiable curve which is not N-computable (by Theorem 8). This curve C is of course K-computable (Theorem 2). By the Theorem 4, C does not have a computable length. In fact, by a direct construction, we can show that even an N-computable curve may have a non-computable length.

Theorem 5. *There is an N-computable curve with a non-computable length.*

Proof. (Sketch) Let l be a left computable but not computable real number. There is an increasing computable sequence (l_n) of rational numbers which converges to l. Construct a computable sequence (p_n) of rational polygons such that the distance between p_n and p_{n+1} is bounded by $2^{(n+1)}$ and $l_n = l(p_n)$ for all n. Then we can choose a normalize computable parameterization f_n of p_n such that $|f_n(t) - f_{n+1}(t)| \leq 2^{-n}$ for each n. Therefore the limit curve $p := \lim p_n$ has a computable normalized parameterization $f := \lim f_n$ and hence is N-computable. The length of the N-computable curve p is l which is not computable.

3 A Technical Lemma

In this section we will show a technical lemma which will be used for the proofs of our main results in section 4. Remember that our goal is to separate the classes of curves by points covered by the curves. That is, we are interested in the points which are covered by curves from one class of curves but cannot be covered by any curves from another class of curves.

The next lemma shows a simple fact related to two curves which separates a curve from another one by a small neighborhood as long as the first curve is not a part of the second.

Lemma 1. *Let C and C' be two rectifiable, non-closed simple curves and let $g : [0;1] \to \mathbb{R}^2$ be a parametrization of C'. If we have $C' \cap U_z \neq \emptyset$ for all points $z \in C$ and all open neighborhoods U_z of z, then there exists an interval $[a;b] \subseteq [0;1]$ such that $g([a;b]) = C$.*

Proof. Suppose that C, C' are rectifiable, non-closed simple curves. If $C' \cap U_z \neq \emptyset$ for any point $z \in C$ and any open neighborhood U_z of z, then C must be a part of C', i.e., $C \subseteq C'$. Otherwise, by the compactness of C', we can find a point z in $C \backslash C'$ which has a positive distance from C' and hence some open neighborhood of z is disjointed from C which contradicts the hypothesis.

Because C' is a rectifiable simple curve, there exists an one-to-one parameterization $f : [0;1] \to C'$. This parameterization f must be injective since C' is non-closed. Therefore the inverse function f^{-1} exists which is also continuous and maps particularly two end points of C to $u, v \in [0;1]$. Suppose w.l.o.g. that $u < v$. Then we have $f([u;v]) = C$.

Let $h : [0;1] \to [0;1]$ be a continuous function defined by $h := f^{-1} \circ g$. Since $f([0;1]) = C \subseteq C' = g([0;1])$, we have $[u;v] \subseteq h([0;1])$. By the continuity of h, there exist $a \in h^{-1}(u)$ and $b \in h^{-1}(v)$ such that $h([a;b]) = [u;v]$ (we suppose w.l.o.g that $a < b$). This implies immediately that $g([a;b]) = C$.

By Lemma 1, if a curve C is not contained completely in another curve C', then there exist a point z in C and a small neighborhood U_z around z such that U is totally disjoint from the curve C'. Particularly, if C is longer than C', then C cannot be completely contained in C'. If in addition C is a rational polygon and C' is a computable curve, then such a point z and the corresponding neighborhood U_z can be effectively found. That is, we have the following lemma.

Lemma 2. *Let C be a rational polygon and let C' be a computable curve. If the curve C is not contained completely in the curve C', then we can effectively find a rational point z on C and a rational neighborhood U_z of z such that $C' \cap U_z = \emptyset$.*

4 Point-Separability

This section will prove our main results that the four versions of computable curves introduced in the Definition 1 are different. More interestingly, we will see that four classes of computable curves cover even different point sets in the plane.

The difference between the R-computable curve and M-computable curve follows from a recent result of Gu, Lutz and Mayordomo [5]. They actually show that there is a polynomial time computable curve $\mathbf{\Gamma}$ which does not have any injective computable parametrization. In other words, any computable parametrization f of $\mathbf{\Gamma}$ must be retraced in the sense that $f(I_1) = f(I_2)$ for some disjoint subintervals $I_1, I_2 \subseteq [0;1]$. Thus, $\mathbf{\Gamma}$ is R-computable but not M-computable.

Our main theorem shows actually even more. Namely, the four classes $\mathbb{C}_K$, $\mathbb{C}_R$, $\mathbb{C}_M$ and $\mathbb{C}_N$ of computable curves are not only different, they cover also different sets of points in the plane. More precisely, they are all "point-separable" in the following sense.

Definition 2. *Let $\mathbb{C}$ and $\mathbb{C}_1$ be classes of curves.*

1. *A point x is called a* $\mathbb{C}$-point *if it is a point of some curve C in the class $\mathbb{C}$.*

2. *The classes* $\mathbb{C}$ *and* $\mathbb{C}_1$ *are called* point-separable *if the sets of* $\mathbb{C}$*-points and* $\mathbb{C}_1$*-points are different.*

Remember that a function $f : [0,1] \to \mathbb{R}^2$ is computable if there is a Turing machine which computes f. Let (M_n) be an effective enumeration of all Turing machines M_n which compute the (possibly partial) functions $\varphi_n : [0,1] \to \mathbb{R}^2$. Then (φ_n) is an effective enumeration of functions including all total computable functions from $[0,1]$ to $\mathbb{R}^2$.

Theorem 6. *There exists a* K*-computable curve* C *and a point* z *on* C *such that* z *does not belong to any* R*-computable curve* C'*. In other words, the classes* $\mathbb{C}_K$ *and* $\mathbb{C}_R$ *are point-separable.*

Proof. (Sketch) We are going to construct a K-computable curve C and a point z which satisfy the condition mention in the theorem. By Definition 1, the K-computability of the curve C requires a computable sequence of finite sets (compact covers) of rational neighborhoods which approximates the curve C effectively. Such kind of compact covers can be easily constructed from rational polygons. Therefore, we need only to construct a computable sequence (p_n) of rational polygons which converges effectively to the curve C.

If C' is an R-computable curve, then C' has a computable parameterization $\varphi_i : [0,1] \to \mathbb{R}^2$, for some i, which is computed by the Turing machine M_i. Denote this curve simply by C_i. For the technical simplicity, let C_i be an empty set (curve) if M_i does not compute a total computable function. Therefore (C_i) is an effective enumeration of all R-computable curves. Thus, it suffices to construct the K-computable curve C and a point z on C which satisfy the following requirements:

$$R_i : \text{If } C_i \text{ has a finite length, then point } z \text{ does not belong to } C_i$$

To satisfy a single requirement R_i, we choose a straight line segment of the constructed polygon C. For simplicity, consider just the line segment J which connects the points $(0,0)$ and $(1,0)$. Simulate the computation of M_i to sufficient precision. If M_i computes a parameterization of the curve C_i which is not very close to J, then, by Lemma 1, we can find a point z on J and a neighborhood V of z such that $C_i \cap V = \emptyset$. If, on the other hand, C_i looks very close to J, then we have to look at more closely how the function φ_i possibly traces the segment J.

For any $q \in [0,1]$ and $\epsilon < l(J)/2$, we say that φ_i has a (q,ϵ)-sweep if the function φ_i approximately traces from $(q,0)$ to $(q+\epsilon,0)$, back to $(q,0)$ and finally passes $(q+\epsilon,0)$ forwardly again. As a parameterization of the curve C_i, φ_i can retrace some segment of C_i several times. However, it is impossible, for a fixed ϵ, that it has (q,ϵ)-sweep for all $q \in [0,1]$. If at some stage we find that φ_i cannot have a (q,ϵ)-sweep, then replace the linear segment from $(q,0)$ to $(q+2\epsilon,0)$ by the polygon which connects the points $(q,0)$, $(q+\epsilon,0)$, (q,δ) and $(q+2\epsilon,0)$ in the given order. Where $\delta > 0$ is a rational number which should be small enough to guarantee the K-computability of the constructed curve. After this change,

the constructed new polygon C is different enough from C_i so that we can apply the Lemma 1 again to find a point z on C and a neighborhood V such that $C_i \cap V = \emptyset$.

In both cases, we have a neighborhood V such that every point in this neighborhood and in C satisfies the requirement R_i. Then, we can consider the segment of C in the neighborhood V to satisfy other requirements R_j for $j > i$. Formally we need a finite injury priority construction

Theorem 6 separates the K-computability from R-computability. In [5] it is shown that the R-computability and M-computability are different too, that is, there is an R-computable curve which does not have any injective computable parameterization at all. This can also be followed from our next more strong result.

Theorem 7. *There exists an R-computable curve C and a point z on C such that z does not belong to any M-computable curves C'. That is, the classes $\mathbb{C}_R$ and $\mathbb{C}_M$ are point-separable.*

Proof. (Sketch) We are going to construct an R-computable curve C and a point z on C which satisfy all the requirements

$$R_i : \text{If } \varphi_i \text{ is an injective parameterization of } C_i, \text{ then } z \text{ is not on } C_i.$$

where (φ_i) is a computable enumeration of all (possibly partial) computable functions $\varphi_i : [0,1] \to \mathbb{R}^2$. The construction uses again the finite injury priority method. We explain the rough idea how to satisfy a single requirement R_i only.

Take a linear segment of the constructed polygon C. For simplicity, consider just the line segment J from the point $(0,0)$ to $(1,0)$ with a parameterization φ which sweeps between these points. That is, φ goes from $(0,0)$ to $(1,0)$ first, then back to $(0,0)$ and finally goes through $(1,0)$ again. This is allowed because we want to construct an R-computable curve C.

Simulate the computation of M_i which computes the function φ_i to sufficient precision. If φ_i is an injective parameterization of C_i, then consider the following cases:

Case 1. C_i is not close to J at all, then we are done by the Lemma 1.

Case 2. C_i closely passes the segment J only once. In this case, alter the segment J by a Z-sweep of height δ which is a polygon connecting the points $(0,0)$, $(1,\delta)$, $(0,-\delta)$ and $(1,0)$ in the given order. Where $\delta > 0$ is a sufficiently small rational number. Then the Lemma 1 can be applied.

Case 3. C_i is close to J and also has Z-sweeps near J. Suppose that the minimal height of all these Z-sweeps is $\epsilon > 0$. Then replace the segment J by a Z-sweep of a height δ such that $\delta < \epsilon/2$. After that we can apply the Lemma 1.

In all three cases, according to Lemma 1, we can fine a z on C and a neighborhood V of z such that $C_i \cap V = \emptyset$. Thus, the segment of C in the neighborhood V can be used to satisfy other requirements R_j for $j > i$. The priority technique guarantees that all requirements can be satisfied simultaneously.

Finally, we want show the difference between M- and N-computability of curves.

Theorem 8. *There exists an M-computable curve C and a point z on C such that z does not belong to any N-computable curves C'. That is, the classes $\mathbb{C}_M$ and $\mathbb{C}_N$ are point-separable.*

Proof. (Sketch) We use priority technique again to construct an M-computable curve C and a point z on C such that the following requirements are satisfied

R_i : If φ_i is a length-normalized parameterization of C_i, then z is not on C_i.

Suppose that C_i is an N-computable curve and φ_i is a length-normalized parameterization of C_i. Choose a linear segment J of already constructed curve C. For simplicity, let J be the line segment connecting the points $(0, 0)$ and $(1, 0)$. Compute φ_i to sufficient precision. If C_i is not close to the segment J, then we can apply the Lemma 1 directly. Otherwise, suppose that C_i is very close to the segment J. That is, there are $t_1, t_2 \in [0, 1]$ such that the segment $\varphi_i([t_1, t_2])$ almost coincides with J. Then compute the middle point $\varphi_i((t_1 + t_2)/2)$ of the segment $\varphi_i([t_1, t_2])$ and check if it is close to the middle point of J. If it is not the case, then φ_i is not length-normalized and we are done. Otherwise, double the length of the first half of the segment J (i.e. the part from $(0, 0)$ to $(1/2, 0)$) by introducing small zigzags. This makes the new segment different enough from the curve C_i and hence we can apply the Lemma 1 to find a point on C and a neighborhood V of z such that $V \cap C_i = \emptyset$. Therefore, the standard priority construction works.

Notice that an N-computable curve has a computable parameterization which traces the curve in one direction and with a constant speed. Thus, Theorem 8 shows that some curve describes the computable particle motion in one direction but the speed of the motion cannot be constant.

Remark: In the proofs of above three theorems, we always choose a linear segment J which connects the points $(0, 0)$ and $(1, 0)$. This choice may help reader to understand how a new polygon should be constructed. However, there is a drawback for this choice of J that we cannot see how to guarantee that the constructed curve has a finite length. So in more formal constructions, we should choose the segment J with much short length so that the new curve increases the length only in a very small portion. This guarantees that the constructed curve is rectifiable.

References

1. V. Brattka and K. Weihrauch. Computability on subsets of Euclidean space I: Closed and compact subsets. *Theoretical Computer Science*, 219:65–93, 1999.
2. M. Braverman and S. Cook. Computing over real numbers: Foundation for scientific computing. *Notics of AMS*, 53(3):318–329, 2006.
3. G. A. Edgar. *Measure, Topology, and Fractal Geometry.* Undergraduate Texts in Mathematics. Springer, 1995.
4. X. Gu, J. H. Lutz, and E. Mayordomo. Points on computable curves. In *Proceedings of FOCS 2006*, pages 469–474. IEEE Computer Society Press, 2006.
5. X. Gu, J. H. Lutz, and E. Mayordomo. Curves that must be retraced, (Preprint) 2008.
6. C. Jordan. *Cours d'analyse de l'Ecole Polytechnique.* Publications mathématiques d'Orsay, 1882.
7. K.-I. Ko. *Complexity Theory of Real Functions.* Progress in Theoretical Computer Science. Birkhäuser, Boston, MA, 1991.
8. K.-I. Ko. A polynomial-time computable curve whose interior has a nonrecursive measure. *Theoretical Computer Science*, 145:241–270, 1995.
9. G. Peano. Sur une vourbe, qui remplit toute une aire plane. *Mathematische Annalen*, 36(1):157–160, 1890.
10. M. B. Pour-El and J. I. Richards. *Computability in Analysis and Physics.* Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1989.
11. K. Weihrauch. *Computable Analysis, An Introduction.* Springer, Berlin Heidelberg, 2000.
12. K. Weihrauch and X. Zheng. Computability on continuous, lower semi-continuous and upper semi-continuous real functions. In T. Jiang and D. Lee, editors, *Computing and Combinatorics*, volume 1276 of *Lecture Notes in Computer Science*, pages 166–175, Berlin, 1997. Springer. Third Annual Conference, COCOON'97, Shanghai, China, August 1997.
13. N. Zhong and K. Weihrauch. Computability theory of generalized functions. *J. ACM*, 50(4):469–505, 2003.

A Note on Closed Subsets in Quasi-zero-dimensional Qcb-spaces (Extended Abstract)

Matthias Schröder

Fakulät für Informatik
Universität der Bundeswehr München, Neubiberg, Germany
matthias.schroeder@unibw.de

Abstract. We introduce the notion of quasi-zero-dimensionality as a substitute for the notion of zero-dimensionality, motivated by the fact that the latter behaves badly in the realm of qcb-spaces. We prove that the category QZ of quasi-zero-dimensional qcb_0-spaces is cartesian closed. Prominent examples of spaces in QZ are the spaces in the sequential hierarchy of the Kleene-Kreisel continuous functionals. Moreover, we characterise some types of closed subsets of QZ-spaces in terms of their ability to allow extendability of continuous functions. These results are related to an open problem in Computable Analysis.

Keywords: Computable Analysis, Qcb-spaces, Extendability

1 Introduction

The category QCB of quotients of countably based spaces [15] has excellent closure properties. For example, it is cartesian closed, in contrast to the category Top of all topological spaces (see [1, 12]). This means that QCB allows us to form products and functions space with the usual transposing properties. Qcb-spaces are known to form exactly the class of topological spaces which can be handled appropriately by the representation based approach to Computable Analysis, the Type Two Model of Effectivity, TTE ([16]).

Unfortunately, exponentiation in QCB behaves badly in terms of preservation of classical topological notions like regularity, normality and zero-dimensionality. For example, the function space $\mathbb{N}^{(\mathbb{N}^{\mathbb{N}})}$ formed in QCB is neither zero-dimensional nor normal (see [13]) despite the fact that both the exponent $\mathbb{N}^{\mathbb{N}}$ and the basis $\mathbb{N}$ are even zero-dimensional Polish spaces. In [14] the notion of a quasi-normal qcb-space is introduced as a substitute for normality in the realm of qcb-spaces (see Section 2.7). This notion has the advantage of being preserved by exponentiation in QCB. Moreover, quasi-normal qcb-spaces admit a Tietze-Urysohn Extension Theorem for continuous real-valued functions defined on functionally closed subspaces.

In an analogous way, we introduce the notion of a quasi-zero-dimensional qcb-space (see Section 3). The category QZ of quasi-zero-dimensional qcb-spaces

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 233-244
http://drops.dagstuhl.de/opus/volltexte/2009/2274

turns out to be an exponential ideal of QCB. In Section 4 we investigate extendability of continuous functions that have as codomain either a quasi-zero-dimensional qcb-space or the real numbers. We prove that a subspace X of a QZ-space Y admits continuous extendability of all continuous functions from X to $\mathbb{N}$ if, and only if, X is in closed in the zero-dimensional reflection of Y. Analogously, we characterise functionally closed subspaces of a quasi-normal qcb-space as those subspaces that admit continuous extendability of all continuous real-valued functions defined on them.

In Section 5 we discuss the relationship of our results with an open problem in Computable Analysis. The problem is whether two hierarchies of functionals over the reals coincide (see [2]).

Since this is an extended abstract, most proofs are omitted.

2 Preliminaries

We repeat some notions and basic facts about sequential spaces, qcb-spaces, pseudobases, and quasi-normal spaces. Moreover, we remind the reader of the definition of the completely regular reflection and of the zero-dimensional reflection of a sequential space.

2.1 Notations

We use sans-serif letters like X, Y etc. to denote topological spaces. We write $\mathcal{O}(\mathsf{X})$ for the topology of a topological space X and $\mathcal{A}(\mathsf{X})$ for the family of closed sets of X. In abuse of notation, we will denote the carrier set of a space X by the same symbol X.

We use the following symbols for relevant topological spaces: $\mathbb{R}$ for the space of real numbers endowed with the Euclidean topology, $\mathbb{I}$ for the unit interval $[0,1]$ endowed with the Euclidean subspace topology, $\mathbb{N}$ for the discrete topological space of natural numbers $\{0,1,2,\dots\}$, $\mathbb{J}$ for the one-point compactification of $\mathbb{N}$ with carrier set $\mathbb{N}\cup\{\infty\}$, and the the sans-serif letter $\mathsf{2}$ for the two-point discrete space with points 0 and 1.

2.2 Sequential spaces, sequential coreflections

A subset A of a topological space X is called *sequentially closed*, if A contains any limit of any convergent sequence of points in A. Complements of sequentially closed sets are called *sequentially open*. For a given topology τ, we denote the topology of sequentially open sets by $\mathit{seq}(\tau)$. Spaces such that every sequentially open set is open are called *sequential*. The sequential coreflection (or sequentialisation) $\mathit{seq}(\mathsf{X})$ of X is the topological space that carries the topology $\mathit{seq}(\mathcal{O}(\mathsf{X}))$ consisting of all sequentially open sets of X. The operator seq is idempotent. Importantly, a function between two sequential spaces is topologically continuous if, and only if, it is sequentially continuous.

For more details about the theory of sequential spaces we refer to [3, 17].

2.3 Qcb-spaces

A qcb-space [15] is a topological *q*uotient of a *c*ountably-*b*ased topological space. Qcb_0-spaces, i.e. qcb-spaces that satisfy the T_0-property, are well-established to be exactly the class of sequential spaces which can be handled by the Type Two Model of Effectivity.

Qcb-spaces are hereditarily Lindelöf (i.e. any open cover of any subset has a countable subcover) and sequential. The category QCB of qcb-spaces as objects and of continuous functions as morphisms is cartesian closed. Moreover, QCB has all countable limits and all countable colimits. For two qcb-spaces A and B we denote by $\mathsf{A} \times \mathsf{B}$ their product, by $\mathsf{A} + \mathsf{B}$ their coproduct, and by B^{A} their function space formed in QCB.

More information can be found in [1, 11, 12, 15].

2.4 Pseudobases and pseudo-open decompositions

Given a topological space X, we say that a family $\mathcal{A}$ of subsets of X is a *pseudo-open decomposition* of a subset M, if $M = \bigcup \mathcal{A}$ holds and for every sequence $(x_n)_n$ that converges to some element $x_\infty \in M$ there is some set $B \in \mathcal{A}$ and some $n_0 \in \mathbb{N}$ such that $\{x_n, x_\infty \,|\, n \geq n_0\} \subseteq B \subseteq M$ holds. Clearly, a set has a pseudo-open decomposition if, and only if, it is sequentially open.

A *(sequential) pseudobase* for X is a family $\mathcal{B}$ of subsets such that every open set has a pseudo-open decomposition into members of $\mathcal{B}$. Any base of topological space is a pseudobase, but not vice versa. Pseudobases characterise qcb-spaces: a sequential space is a qcb-space if, and only if, it has a countable pseudobase. Any countably pseudobased space is hereditarily Lindelöf and its sequential coreflection is a qcb-space. In this paper we will only deal with spaces having a countable pseudobase. More information can be found in [4, 12, 15].

2.5 Completely regular reflections, functionally open sets

Let X be a sequential space. The *completely regular reflection* of X is defined to carry the topology that is induced by the base

$$\mathcal{B} := \left\{ h^{-1}(0,1] \,\middle|\, h\colon \mathsf{X} \to \mathbb{I} \text{ is continuous} \right\}.$$

We denote this topological space by $\mathcal{R}(\mathsf{X})$. It has the property that every real-valued function f on X is continuous w.r.t. the original topology $\mathcal{O}(\mathsf{X})$ if, and only if, f is continuous w.r.t. the topology $\mathcal{O}(\mathcal{R}(\mathsf{X}))$. If $\mathcal{R}(\mathsf{X})$ is a T_0-space then $\mathcal{R}(\mathsf{X})$ is a Tychonoff space.

A subset U of X is called *functionally open*, if there is a continuous function h from X to the unit interval $\mathbb{I} = [0,1]$ such that $h^{-1}\{0\} = \mathsf{X} \setminus A$. Complements of functionally open sets are called *functionally closed*. A common term for "functionally closed set" is *zero-set*, and for "functionally open set" is *cozero-set*. We denote the family of functionally open sets of X by $\mathcal{FO}(\mathsf{X})$ and the family of functionally closed sets by $\mathcal{FA}(\mathsf{X})$. If X is a hereditarily Lindelöf space then $\mathcal{FO}(\mathsf{X})$

forms the topology of the completely regular reflection $\mathcal{R}(\mathsf{X})$. Otherwise $\mathcal{FO}(\mathsf{X})$ need not be a topology.

Regularity, normality and perfect normality[1] are equivalent for hereditarily Lindelöf spaces, thus for countably pseudobased spaces and for qcb-spaces.

2.6 Zero-dimensional spaces, zero-dimensional reflections

A *zero-dimensional space* is a topological space that has a base consisting of *clopen* (= closed and open) sets. Any zero-dimensional T_0-space is regular. Zero-dimensional hereditarily Lindelöf spaces X are even *strongly zero-dimensional*, meaning that any pair of disjoint closed sets A, B can be separated by a clopen set C (i.e. $A \subseteq C \subseteq \mathsf{X} \backslash B$). Strongly zero-dimensional T_0-spaces are zero-dimensional and normal.

Let X be a sequential space. The *zero-dimensional reflection* of X is defined to be the space that carries the topology induced by the base

$$\mathcal{B} := \left\{ h^{-1}\{1\} \,\middle|\, h\colon \mathsf{X} \to \mathsf{2} \text{ is continuous} \right\}.$$

We denote this space by $\mathcal{Z}(\mathsf{X})$. Clearly, $\mathcal{Z}(\mathsf{X})$ is zero-dimensional. If X is hereditarily Lindelöf then the zero-dimensional reflection $\mathcal{Z}(\mathsf{X})$ is hereditarily Lindelöf as well and thus strongly zero-dimensional (see [3]).

2.7 Quasi-normal spaces and the category QN

A *quasi-normal* space is defined to be the sequential coreflection of some normal space [14]. The category of quasi-normal qcb-spaces, which is denoted by QN, is cartesian closed and inherits finite products and exponentials from its supercategory QCB. This is not the case for the category of normal qcb-spaces. Any continuous function $f\colon \mathsf{X} \to \mathbb{R}$ from a functionally closed subspace X of a space $\mathsf{Y} \in \mathsf{QN}$ can be extended to a continuous function $F\colon \mathsf{Y} \to \mathbb{R}$. Details can be found in [14].

3 Quasi-zero-dimensional Qcb-Spaces

In this section we introduce and investigate the notion of a quasi-zero-dimensional qcb-space.

The category QCB of qcb-spaces is known to be cartesian closed. However, forming function spaces in QCB does not preserve classical topological notions like regularity, normality and zero-dimensionality. For example, the function space $\mathbb{N}^{(\mathbb{N}^{\mathbb{N}})}$ formed in QCB is neither zero-dimensional nor normal (see [13]),

[1] A *normal* space is a T_1-space such that for a pair of disjoint closed sets (A, B) there exists a pair of disjoint open sets (U, V) such that $A \subseteq U$ and $B \subseteq V$. A perfectly normal space is a T_1-space in which every closed sets is functionally closed. Note that some authors omit the T_1-condition.

although both $\mathbb{N}$ and $\mathbb{N}^{\mathbb{N}}$ are zero-dimensional and normal. Hence the final topology of the natural TTE-representation on $\mathbb{N}^{(\mathbb{N}^{\mathbb{N}})}$, which is equal to the topology of $\mathbb{N}^{(\mathbb{N}^{\mathbb{N}})}$, is not zero-dimensional. By contrast, the compact-open topology on $\mathbb{N}^{(\mathbb{N}^{\mathbb{N}})}$ is even strongly zero-dimensional.

This fact motivates the introduction of an appropriate substitute for the property of zero-dimensionality in the realm of qcb-spaces. We use the same idea as in [14], where the notion of quasi-normality is defined as a replacement for normality. The idea behind the following definition is the fact that finite products and function spaces in the category QCB are constructed as the sequential coreflection of their counterparts in classical topology, which enjoy the property of preserving zero-dimensionality.

Definition 1. A qcb-space X is called *quasi-zero-dimensional*, if X is the sequential coreflection of a zero-dimensional T_0-space.

So a qcb-space is quasi-zero-dimensional if, and only if, its convergence relation is induced by some zero-dimensional T_0-topology. Clearly, any quasi-zero-dimensional space is hereditarily disconnected. Simple examples of quasi-zero-dimensional spaces are zero-dimensional separable metrisable spaces, because they are equal to their own sequentialisation. By QZ we denote the full subcategory of QCB that are quasi-zero-dimensional spaces.

Recall that a quasi-normal space is defined to be the sequential coreflection of a normal space (see [14]). Since zero-dimensional hereditarily Lindelöf T_0-spaces are normal, we have:

Lemma 1. *Any* QZ*-space is a* QN*-space (and thus a Hausdorff space).*

3.1 Characterisation of quasi-zero-dimensionality

We will give now several characterisations of QZ-spaces. They are analogous to characterisations of quasi-normality given in [14, Section 3.2]. We begin with the following observation.

Lemma 2. *A qcb_0-space X is quasi-zero-dimensional if, and only if, it is the sequential coreflection of its zero-dimensional reflection $\mathcal{Z}(\mathsf{X})$.*

For the second characterisation, we define two families of (respectively, closed and open) subsets of a topological space X by

$$\mathcal{ZA}(\mathsf{X}) := \big\{h^{-1}\{\infty\} \,\big|\, h\colon \mathsf{X} \to \mathbb{J} \text{ continuous}\big\},\ \mathcal{ZO}(\mathsf{X}) := \big\{\mathsf{X} \setminus A \,\big|\, A \in \mathcal{ZA}(\mathsf{X})\big\}.$$

Here $\mathbb{J}$ denotes the one-point compactification of $\mathbb{N}$. Obviously, every set in $\mathcal{ZA}(\mathsf{X})$ is closed in the zero-dimensional reflection of X. We will sometimes use the term *$\mathcal{Z}$-closed* for the members of $\mathcal{ZA}(\mathsf{X})$ and *$\mathcal{Z}$-open* for members of $\mathcal{ZO}(\mathsf{X})$. In general, $\mathcal{ZO}(\mathsf{X})$ is not a topology, unless X is hereditarily Lindelöf.

Lemma 3. *Let X be a hereditarily Lindelöf space. Then $\mathcal{ZO}(\mathsf{X})$ is the family of all open sets of $\mathcal{Z}(\mathsf{X})$. Dually, $\mathcal{ZA}(\mathsf{X})$ is the family of all closed sets of $\mathcal{Z}(\mathsf{X})$.*

Obviously, every $\mathcal{Z}$-closed in a quasi-zero-dimensional space is functionally closed, because $\mathbb{J}$ is homeomorphic to the closed subspace $\{0, 2^{-n} \mid n \in \mathbb{N}\}$ of $\mathbb{I}$. It is not known for which QZ-spaces the converse is true as well.

Definition 2. We say that a qcb-space X has *Normann's property*, if $\mathsf{X} \in \mathsf{QZ}$ and every functionally closed set of X is closed in the zero-dimensional reflection $\mathcal{Z}(\mathsf{X})$ of X (i.e. $\mathcal{FA}(\mathsf{X}) = \mathcal{ZA}(\mathsf{X})$).

Clearly, zero-dimensional spaces have Normann's property.

Lemma 3 implies the following reformulation of Lemma 2.

Corollary 1. *A qcb_0-space X is quasi-zero-dimensional if, and only if, its convergence relation is induced by the topology $\mathcal{ZO}(\mathsf{X})$.*

We now work towards a characterisation of quasi-zero-dimensionality in terms of pseudobases. Recall that qcb-spaces are known to be those sequential spaces that have a countable pseudobase (see Section 2.4). We start with the following separation lemma for disjoint $\mathcal{Z}$-closed subsets.

Lemma 4. *Let X be a hereditarily Lindelöf space, and let A, B be disjoint closed subsets of $\mathcal{Z}(\mathsf{X})$. Then there is a continuous function $h\colon \mathsf{X} \to \mathbb{J}$ with $h^{-1}\{\infty\} = A$ and $B \subseteq h^{-1}\{0\}$.*

This lemma is instrumental in proving the following lemma about sequentially open sets that are $\mathcal{G}_\delta$-sets in the zero-dimensional reflection of a QZ-space.

Lemma 5. *Let X be a qcb-space equipped with a countable pseudobase consisting of sets in $\mathcal{ZA}(\mathsf{X})$. Then every sequentially open set $V \in \mathcal{O}(\mathsf{X})$ that is a $\mathcal{G}_\delta$-set in $\mathcal{Z}(\mathsf{X})$ is open in $\mathcal{Z}(\mathsf{X})$. Dually, every sequentially closed set $A \in \mathcal{A}(\mathsf{X})$ that is an $\mathcal{F}_\sigma$-set in $\mathcal{Z}(\mathsf{X})$ is closed in $\mathcal{Z}(\mathsf{X})$.*

Now we are ready to characterise quasi-zero-dimensional qcb-spaces in terms of properties of pseudobases.

Proposition 1. *A qcb_0-space X is quasi-zero-dimensional if, and only if, it has a countable pseudobase consisting of sets in $\mathcal{ZA}(\mathsf{X})$.*

Note that quasi-normal qcb-spaces are characterised via countable pseudobases consisting of functionally closed sets (see [14, Proposition 4]).

A continuous function $e\colon \mathsf{X} \to \mathsf{Y}$ between sequential spaces X, Y is said to *reflect convergent sequences*, if, for any sequence $(x_n)_n$ in X and any point $x_\infty \in \mathsf{X}$, $(x_n)_n$ converges to x_∞ in X whenever $(e(x_n))_n$ converges to $e(x_\infty)$ in Y.

Proposition 2. *A qcb-space X is quasi-zero-dimensional if, and only if, there are a qcb-space Z and a continuous injection $e\colon \mathsf{X} \to 2^{\mathsf{Z}}$ that reflects convergent sequences.*

3.2 Constructing quasi-zero-dimensional spaces

The category QZ of quasi-zero-dimensional qcb-spaces enjoys excellent closure properties. Like quasi-normality, quasi-zero-dimensionality is preserved by forming (i) countable products, (ii) subspaces, (iii) countable coproducts, and (iv) function spaces in the category QCB of qcb-spaces. So QZ inherits the cartesian-closed structure of QCB. In fact QZ is an exponential ideal of QCB.

Theorem 1. *The category* QZ *of quasi-zero-dimensional qcb-spaces is cartesian closed. Moreover, it has all countable limits, all countable colimits and is an exponential ideal of* QCB*.*

Proof. Similar to the proof of Theorem 6 in [14]. Alternatively, one can apply Proposition 2.

Obviously, all zero-dimensional metric spaces are in QZ. Theorem 1 implies that all Kleene-Kreisel spaces [5] of the form $\mathbb{N}^{\mathsf{Z}}$ belong to QZ. Furthermore, for all $k \in \mathbb{N}$ the space $\mathsf{N}\langle k\rangle$ of Kleene-Kreisel continuous functional of order k (see [7, 8, 10]) is a quasi-zero-dimensional space. The hierarchy $(\mathsf{N}\langle k\rangle)_k$ is recursively defined by the formula $\mathsf{N}\langle 0\rangle := \mathbb{N}$ and $\mathsf{N}\langle k+1\rangle := \mathbb{N}^{\mathsf{N}\langle k\rangle}$. On the other hand, the Euclidean space $\mathbb{R}$ is not quasi-zero-dimensional by being connected.

Remark 1. One can show that there is a cartesian closed embedding of QZ into the cartesian closed category $\mathsf{k}_2\mathsf{0dim}$ considered by G. Lukács in [9]. This category is itself equivalent to a full reflective sub-ccc of the category of Hausdorff k-spaces.

4 Extendability of continuous functions

In this section we investigate extendability of continuous functions defined on subspaces of quasi-zero-dimensional spaces. Moreover, we classify subspaces in terms of their ability to admit extendability of continuous functions.

4.1 A transitivity property for $\mathcal{Z}$-closed sets

It is well-known that the subspace operator on topological spaces has the following transitivity property: Any closed subset of a closed subspace is closed in the original space, whereas the analogous statement for functionally closed sets is false in general (see [3, 2.1.B]).

In [14], it is shown that functionally closed sets in quasi-normal qcb-spaces do have the transitivity property. Recall that functionally closed sets of a QN-space Y are exactly the closed sets of the completely regular reflection of Y.

In Proposition 1 and Lemma 3 we have seen that $\mathcal{Z}$-closed sets play a similar role for QZ-spaces as functionally closed sets do for QN-spaces. Validity of the transitivity property for $\mathcal{Z}$-closed sets is related to extendability of continuous functions with zero-dimensional codomains as follows: Let X be a $\mathcal{Z}$-closed subspace of a sequential space Y. If any continuous function from X to $\mathbb{J}$ (the

one-point compactification of $\mathbb{N}$) is extendable onto Y, then any closed subset A of $\mathcal{Z}(\mathsf{X})$ is also closed in $\mathcal{Z}(\mathsf{Y})$: Choose continuous functions $f\colon \mathsf{X} \to \mathbb{J}$ and $g\colon \mathsf{Y} \to \mathbb{J}$ with $f^{-1}\{\infty\} = A$, $g^{-1}\{\infty\} = \mathsf{X}$ and extend f to a continuous function $F\colon \mathsf{Y} \to \mathbb{J}$. Then the function $h\colon \mathsf{Y} \to \mathbb{J}$ defined by $h(y) := \min\{F(y), g(y)\}$ is a continuous function witnessing that A is closed in $\mathcal{ZA}(\mathsf{Y})$.

Fortunately, the transitivity property for $\mathcal{Z}$-closed sets is valid in the realm of QZ-spaces (see Proposition 3). So the zero-dimensional reflection of any $\mathcal{Z}$-closed subspace is a subspace of the zero-dimensional reflection of its QZ-superspace.

Proposition 3. *Let $\mathsf{Y} \in \mathsf{QZ}$. Let X be a subspace of Y with $\mathsf{X} \in \mathcal{ZA}(\mathsf{Y})$. Then every set that is closed in $\mathcal{Z}(\mathsf{X})$ is closed in $\mathcal{Z}(\mathsf{Y})$. Moreover, $\mathcal{Z}(\mathsf{X})$ is a topological subspace of $\mathcal{Z}(\mathsf{Y})$.*

4.2 Extendability of continuous functions into QZ-spaces

In this section we work towards showing that, for any zero-dimensional Polish space B, any continuous B-valued function defined on a $\mathcal{Z}$-closed subspace of a QZ-space is continuously extendable. We start by showing that clopens of $\mathcal{Z}$-closed subspaces extend to clopens of the whole space, provided that the latter is in QZ.

Lemma 6. *Let $\mathsf{Y} \in \mathsf{QZ}$, and let X be a subspace of Y with $\mathsf{X} \in \mathcal{ZA}(\mathsf{Y})$. Then for every set D that is clopen in X there is a clopen C in Y with $D = C \cap \mathsf{X}$.*

Proof. By Proposition 3, both D and $\mathsf{X} \setminus D$ are closed sets in $\mathcal{Z}(\mathsf{Y})$. By strong zero-dimensionality of $\mathcal{Z}(\mathsf{Y})$, there is a clopen set C in Y with $D \subseteq C \subseteq \mathsf{X} \setminus D$. Clearly, $C \cap \mathsf{X} = D$. □

Lemma 6 can be reformulated by stating that any continuous function from a $\mathcal{Z}$-closed subset into the two-point discrete space $\mathsf{2}$ has a continuous extension.

We now investigate the full subcategory ZEXT of QCB consisting of those quasi-zero-dimenional qcb-spaces $\mathsf{Z} \in \mathsf{QZ}$ that have the following property: For all spaces $\mathsf{Y} \in \mathsf{QZ}$, for all $\mathcal{Z}$-closed subspaces X of Y and for all continuous functions $f\colon \mathsf{X} \to \mathsf{Z}$ there exists a continuous function $F\colon \mathsf{Y} \to \mathsf{Z}$ extending f. Lemma 6 states that $\mathsf{2}$ is an object of ZEXT.

Given two qcb-spaces Y, B, we say that a subspace X of Y *admits a continuous extension operator for* B, if there exists a continuous function $E\colon \mathsf{B}^{\mathsf{X}} \to \mathsf{B}^{\mathsf{Y}}$ satisfying $E(f)(x) = f(x)$ for all $x \in \mathsf{X}$ and all continuous functions $f\colon \mathsf{X} \to \mathsf{B}$. Cartesian closedness of QZ (see Theorem 1) yields the following characterisation of the objects in ZEXT.

Proposition 4. *A space $\mathsf{Z} \in \mathsf{QZ}$ is an object of ZEXT if, and only if, any $\mathcal{Z}$-closed subspace X of any space $\mathsf{Y} \in \mathsf{QZ}$ admits a continuous extension operator $E\colon \mathsf{Z}^{\mathsf{X}} \to \mathsf{Z}^{\mathsf{Y}}$ for Z.*

The category ZEXT enjoys excellent closure properties.

Proposition 5.
1. *If* $\mathsf{A}, \mathsf{B} \in \mathsf{ZEXT}$*, then* $\mathsf{A} \times \mathsf{B} \in \mathsf{ZEXT}$.
2. *If* $\mathsf{B} \in \mathsf{ZEXT}$ *and* $\mathsf{A} \in \mathsf{QCB}$*, then* $\mathsf{B}^{\mathsf{A}} \in \mathsf{ZEXT}$.
3. *If* $\mathsf{A}, \mathsf{B} \in \mathsf{ZEXT}$*, then* $\mathsf{A} + \mathsf{B} \in \mathsf{ZEXT}$.
4. *If* $\mathsf{B} \in \mathsf{ZEXT}$ *and* A *is a* QCB*-retract of* B*, then* $\mathsf{A} \in \mathsf{ZEXT}$.
5. *If* $\mathsf{B} \in \mathsf{ZEXT}$ *and* A *is a* $\mathcal{Z}$*-open subspace of* B*, then* $\mathsf{A} \in \mathsf{ZEXT}$.

In the category of sequential spaces and hence in QCB the discrete space $\mathbb{N}$ is homeomorphic to the function space $2^{2^{\mathbb{N}}}$ by [2, Proposition 3]. Moreover, by [6, Theorem 7.8] every zero-dimensional Polish space is homeomorphic to a closed subset of the Baire space $\mathbb{N}^{\mathbb{N}}$. In turn, this closed subspace is a retract of $\mathbb{N}^{\mathbb{N}}$ by [2, Proposition 2]. We obtain by Proposition 5 and Lemma 6:

Example 1. The following spaces are objects of ZEXT:
(a) the discrete space $\mathbb{N}$,
(b) the Baire space $\mathbb{N}^{\mathbb{N}}$,
(c) any zero-dimensional Polish space,
(d) the one-point compactification $\mathbb{J}$ of $\mathbb{N}$,
(e) for any $k \in \mathbb{N}$ the space $\mathsf{N}\langle k \rangle$ of all Kleene-Kreisel continuous functionals of order k (see Section 3.2).

4.3 Subspaces that admit continuous extendability

Now we study under which conditions a subspace admits continuous extendability of continuous functions. We start with the following simple observation.

Lemma 7. *Let* $\mathsf{Y} \in \mathsf{QZ}$ *and let* X *be a subspace of* Y *such that for every subset* $D \subseteq \mathsf{X}$ *that is clopen in* X *there exists a clopen* C *in* Y *with* $D = C \cap \mathsf{X}$*. Then* X *is sequentially closed.*

We have already seen that the property of X being closed in $\mathcal{ZA}(\mathsf{Y})$ is sufficient to guarantee extendability of all continuous $\mathbb{N}$-valued functions defined on X. We show that this condition is also necessary.

Lemma 8. *Let* X *be a subspace of a* $\mathsf{Y} \in \mathsf{QZ}$*. If every continuous function* $h\colon \mathsf{X} \to \mathbb{N}$ *can be extended to a continuous function* $H\colon \mathsf{Y} \to \mathbb{N}$*, then* $\mathsf{X} \in \mathcal{ZA}(\mathsf{Y})$.

We obtain as an easy consequence:

Corollary 2. *Let* A *be a retract of a space* $\mathsf{Y} \in \mathsf{QZ}$*. Then* A *is homeomorphic to a* $\mathcal{Z}$*-closed subspace of* Y.

Lemma 8 generalises to all non-compact QZ-spaces replacing $\mathbb{N}$ as codomain space.

Proposition 6. *Let* $\mathsf{Z} \in \mathsf{QZ}$ *such that* Z *is not compact. Let* X *be a subspace of a space* $\mathsf{Y} \in \mathsf{QZ}$ *such that every continuous function* $f\colon \mathsf{X} \to \mathsf{Z}$ *can be extended to a continuous function* $F\colon \mathsf{Y} \to \mathsf{Z}$*. Then* $\mathsf{X} \in \mathcal{ZA}(\mathsf{Y})$.

Proposition 6 is a consequence of Lemma 8 and the following equivalence.

Lemma 9. *A space* $\mathsf{X} \in \mathsf{QZ}$ *is not compact if, and only if,* $\mathbb{N}$ *is a retract of* X.

We do not know whether Lemma 8 is valid for the two-point discrete space $\mathsf{2}$ replacing $\mathbb{N}$. However, the (possibly) stronger condition on a subspace X to admit a continuous extension operator for the continuous functions with codomain $\mathsf{2}$ is enough to ensure that X is $\mathcal{Z}$-closed.

Proposition 7. *Let* $\mathsf{Y} \in \mathsf{QZ}$. *Let* X *be a subspace of* Y *that admits a continuous extension operator* $E\colon \mathsf{2}^{\mathsf{X}} \to \mathsf{2}^{\mathsf{Y}}$. *Then* $\mathsf{X} \in \mathcal{ZA}(\mathsf{Y})$.

With the help of Propositions 2, 5 and 7 one can prove:

Proposition 8. *A qcb-space* X *is an object of* ZEXT *if, and only if, there is a qcb-space* Z *such that* X *is a retract of* $\mathsf{2}^{\mathsf{Z}}$.

We summerise some of the above results in a characterisation theorem for sets that are closed in the zero-dimensional reflection.

Theorem 2 (Characterisation of $\mathcal{Z}$-closed subsets). *Let* Y *be a quasi-zero-dimensional qcb-space, and let* X *be a subspace of* Y. *Then the following statements are equivalent:*

(a) The set X *is closed in the zero-dimensional reflection* $\mathcal{Z}(\mathsf{Y})$ *of* Y.
(b) The set is $\mathcal{Z}$*-closed (i.e.* $\mathsf{X} \in \mathcal{ZA}(\mathsf{Y})$*).*
(c) The subspace X *admits a continuous extension operator* $E\colon \mathsf{2}^{\mathsf{X}} \to \mathsf{2}^{\mathsf{Y}}$.
(d) The subspace X *admits a continuous extension operator* $E\colon \mathbb{N}^{\mathsf{X}} \to \mathbb{N}^{\mathsf{Y}}$.
(e) Any continuous function $f\colon \mathsf{X} \to \mathbb{N}$ *can be extended to a continuous function* $F\colon \mathsf{Y} \to \mathbb{N}$.
(f) There is a non-compact quasi-zero-dimensional qcb-space Z *such that any continuous function* $f\colon \mathsf{X} \to \mathsf{Z}$ *can be extended to a continuous function* $F\colon \mathsf{Y} \to \mathsf{Z}$.

4.4 Characterisation of functionally closed subsets

In this section we present a characterisation of all functionally closed subsets of quasi-normal spaces that is similar to Theorem 2.

In [14] it is shown that real-valued functions defined on a functionally closed subspace can be extended to the whole space, provided the latter is a quasi-normal qcb-space. We remark that cartesian closedness of QN implies the following uniform versions of this extendability result.

Proposition 9. *Let* X *be a functionally closed subspace of a space* $\mathsf{Y} \in \mathsf{QN}$. *Then* X *admits continuous extension operators* $E_{\mathbb{I}}\colon \mathbb{I}^{\mathsf{X}} \to \mathbb{I}^{\mathsf{Y}}$ *and* $E_{\mathbb{R}}\colon \mathbb{R}^{\mathsf{X}} \to \mathbb{R}^{\mathsf{Y}}$.

Now we investigate under which condition a subspace admits continuous extendability of continuous real-valued functions. We begin with the following simple observation which is analogous to Lemma 7.

Lemma 10. *Let* $\mathsf{Y} \in \mathsf{QN}$*. Let* X *be a* QCB*-subspace of* Y *such that every continuous function* $f\colon \mathsf{X} \to \mathbb{I}$ *can be extended to a continuous function* $F\colon \mathsf{Y} \to \mathbb{I}$*. Then* X *is sequentially closed.*

The fact that every qcb-space is hereditarily Lindelöf implies the following observation.

Lemma 11. *Let* $\mathsf{Y} \in \mathsf{QN}$*. Let* X *be a* QCB*-subspace of* Y *such that every continuous function* $f\colon \mathsf{X} \to \mathbb{R}$ *can be extended to a continuous function* $F\colon \mathsf{Y} \to \mathbb{R}$*. Then* $\mathsf{X} \in \mathcal{FA}(\mathsf{Y})$*.*

We obtain the following corollary which parallels Corollary 2.

Corollary 3. *Let* A *be a retract of a space* $\mathsf{Y} \in \mathsf{QN}$*. Then* A *is homeomorphic to a functionally closed subspace of* Y*.*

We do not know whether non-uniform extendability of all continuous functions on X into the unit interval $\mathbb{I} = [0, 1]$ implies that X is functionally closed. However, if X admits a continuous extension operator for $\mathbb{I}$ as codomain, then X must be functionally closed (cf. Proposition 7).

Proposition 10. *Let* Y *be a quasi-normal qcb-space. Let* X *be a* QCB*-subspace of* Y *that admits a continuous extension operator* $E\colon \mathbb{I}^{\mathsf{X}} \to \mathbb{I}^{\mathsf{Y}}$*. Then* $\mathsf{X} \in \mathcal{FA}(\mathsf{Y})$*.*

We summerise the above results in a characterisation theorem for subsets of quasi-normal qcb-spaces that are closed in the completely regular reflection.

Theorem 3 (Characterisation of functionally closed subsets). *Let* Y *be a quasi-normal qcb-space, and let* X *be a* QCB*-subspace of* Y*. Then the following statements are equivalent:*

(a) The set X *is functionally closed in* Y *(i.e.* $\mathsf{X} \in \mathcal{FA}(\mathsf{Y})$*).*
(b) The set X *is closed the completely regular reflection* $\mathcal{R}(\mathsf{Y})$ *of* Y*.*
(c) The subspace X *admits a continuous extension operator* $E\colon \mathbb{I}^{\mathsf{X}} \to \mathbb{I}^{\mathsf{Y}}$*.*
(d) The subspace X *admits a continuous extension operator* $E\colon \mathbb{R}^{\mathsf{X}} \to \mathbb{R}^{\mathsf{Y}}$*.*
(e) Any continuous function $f\colon \mathsf{X} \to \mathbb{R}$ *can be extended to a continuous function* $F\colon \mathsf{Y} \to \mathbb{R}$*.*

5 Discussion

We have seen that the category QZ of quasi-zero-dimensional qcb-spaces and the category QN of quasi-normal qcb-spaces enjoy several similarities, for example they are exponential ideals of QCB. Both classes of topological spaces possess a distinguished family of closed subsets ($\mathcal{Z}$-closed subsets in the case of QZ and functionally closed subsets in the case of QN) with following property: either class is characterised by the existence of a countable pseudobase consisting of sets in the respective family of closed subsets. Functionally closed subspaces of QN-spaces are characterised as those subspaces that admit continuous extendability

of real-valued functions, while $\mathcal{Z}$-closed subsets are exactly the class of sets which allow to extend continuous functions that have a Kleene-Kreisel space of the form $\mathbb{N}^{\mathsf{Z}}$ as codomain.

It is not known whether every QZ-space Y has *Normann's property* (i.e. every functionally closed set in Y is $\mathcal{Z}$-closed). This question is related to an open problem in Computable Analysis, namely whether or not two natural hierarchies of continuous functionals over the reals (called the intensional hierarchy and the extensional hierarchy, see [2]) coincide. D. Normann [10] proved that the two hierarchies agree if, and only if, for all $k \geq 2$ the space $\mathsf{N}\langle k\rangle$ of Kleene-Kreisel continuous functionals of type k (see Section 3.2) has Normann's property.

Acknowledgement

I would like to thank the anonymous referees for their valuable suggestions.

References

1. Battenfeld, I., Schröder, M., and Simpson, A.: A Convenient Category of Domains. Electronic Notes in Theoretical Computer Science **172** (2007) 69–99
2. Bauer, A., Escardó, M., and Simpson, A.: Comparing Functional Paradigms for Exact Real-number Computation. in *Proc. ICALP '02*, Lecture Notes in Computer Science **2380** (2002) 488–500
3. Engelking, R.: General Topology. Heldermann, Berlin (1989)
4. Escardó, M., Lawson, J., Simpson, A.: Comparing Cartesian-closed Categories of Core Compactly Generated Spaces. Topology and its Applications **143** (2004) 105–145
5. Escardó, M.: Exhausible Sets in Higher-type Computation, Logical Methods in Computer Science **4(3:3)** (2009) 1–37
6. Kelchris, A.: Classical Descriptive Set Theory. Springer, Berlin (1995)
7. Kleene, S.E.: Countable Functionals. In *Constructivity in Mathematics*, North-Holland (1959) 81–100
8. Kreisel, G.: Interpretation of Analysis by means of Functionals of Finite Type. In *Constructivity in Mathematics*, North-Holland (1959) 101–128
9. Lukács, G.: A Cartesian Closed Category of Zero-dimensional Spaces. Unpublished Note
10. Normann, D.: Comparing Hierarchies of Total Functionals. Logical Methods in Computer Science **1(2:4)** (2005) 1–28
11. Schröder, M.: Extended Admissibility. Theoretical Computer Science **284** (2002) 519–538
12. Schröder, M.: Admissible Representations for Continuous Computations. PhD Thesis, Fachbereich Informatik, FernUniversität Hagen (2002)
13. Schröder, M.: The sequential topology on $\mathbb{N}^{\mathbb{N}^{\mathbb{N}}}$ is not regular. Accepted for publication in Mathematical Structures in Computer Science (2009)
14. Schröder, M.: An Effective Tietze-Urysohn Theorem for QCB-Spaces. Journal of Universal Computer Science **15(6)** (2009) 1317–1336
15. Simpson, A.: Towards a Convenient Category of Topological Domains. In: Proceedings of thirteenth ALGI Workshop, RIMS, Kyoto University (2003)
16. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)
17. Willard, S.: General Topology. Addison-Wesley, Reading (1970)

Random Iteration Algorithm for Graph-Directed Sets

Yoshiki Tsujii[1], Takakazu Mori[1], Mariko Yasugi[2], and Hideki Tsuiki[3]

[1] Faculty of Science, Kyoto Sangyo University
tsujiiy,morita@cc.kyoto-su.ac.jp
[2] Kyoto Sangyo University
yasugi@cc.kyoto-su.ac.jp
[3] Graduate School of Human and Environmental Studies, Kyoto University
tsuiki@i.h.kyoto-u.ac.jp

Abstract. A random iteration algorithm for graph-directed sets is defined and discussed. Similarly to the Barnsley-Elton's theorem, it is shown that almost all sequences obtained by this algorithm reflect a probability measure which is invariant with respect to the system of contractions with probabilities.

1 Introduction

The motif of this article is the random iteration algorithm for a family of graph-directed sets. According to Barnsley [1], the random iteration algorithm can be used to picture a fractal defined by a finite number of contractions. Our interest is to extend this idea to graph-directed sets (cf. [7], [8], [9], [10]).

Our present interest was originally motivated by the work of Brattka [4], in which Brattka presented an example of a "Fine-computable" function which is not "locally uniformly Fine-computable." The graph of Brattka's function can be characterized as a graph-directed set, and in [10] we have depicted graphs of some graph-directed sets by using a deterministic algorithm.

The random iteration algorithm is an alternative for picturing some invariant sets. Let us briefly explain this algorithm according to Barnsley and Elton (cf. [1], [2], [6]).

Let $\{S_1, S_2, \ldots, S_K\}$ be a family of contractions on $\mathbf{R}^d$. Let $(p_1, p_2, \ldots, p_K)$ be a system of probabilities assigned to $\{S_1, S_2, \ldots, S_K\}$, where $p_i > 0$ $(i = 1, \ldots, K)$ and $\sum_{i=1}^{K} p_i = 1$. Choose $x(0) \in \mathbf{R}^d$ and choose randomly, recursively and independently $x(t) \in \{S_1(x(t-1)), S_2(x(t-1)), \ldots, S_K(x(t-1))\}$, where the probability for the event $x(t) = S_i(x(t-1))$ is p_i. The sequence $\{x(0), x(1), \ldots, x(n), \ldots\}$ "converges to" the invariant set with respect to $\{S_1, S_2, \ldots, S_K\}$. Moreover, the density of points in this sequence reflects a measure which is invariant with respect to $\{S_1, S_2, \ldots, S_K\}$ and $(p_1, p_2, \ldots, p_K)$ in the sense of Theorem 2 (Barnsley and Elton). Let us give an example.

Example 1 (Koch Curve). The Koch curve is invariant for S_1, S_2, S_3, S_4, where S_i maps the whole triangle to a smaller triangle for $i = 1, 2, 3, 4$ (cf. Fig. 1).

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 245-256
http://drops.dagstuhl.de/opus/volltexte/2009/2275

Let $(3/7, 1/7, 2/7, 1/7)$ be a system of probabilities assigned to $\{S_1, S_2, S_3, S_4\}$. Starting with $x(0) = (0, 0)$, we obtained the figure after 4000 times loop.

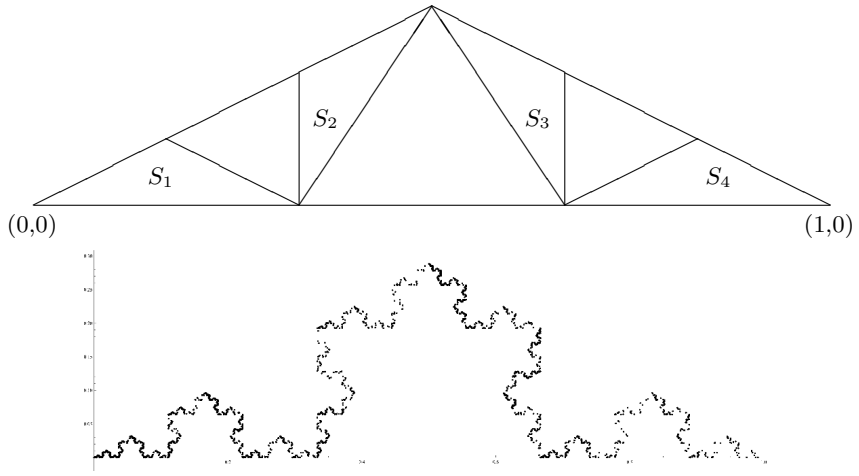


Fig. 1. Koch curve drawn with the random iteration algorithm.

In Section 2, we review the theory of graph-directed sets, and then explain the random iteration algorithm for graph-directed sets. In Section 3, we prove the Barnsley-Elton theorem for graph-directed sets (Theorems 3-5 and Corollary 1). At the end, another random iteration algorithm is proposed and some results thereof are previewed; details will be developed later.

We might note that I. Werner has investigated a random iteration algorithm for a family of graph-directed sets in a different approach in [11].

2 Random iteration algorithm for graph-directed sets

Graph-directed sets are defined as follows ([3], [5] and [9]). Let $K \geq 2$. Let $V = \{1, \ldots, K\}$ be a set of vertices, and let $E_{k,l}$ be a set of edges from vertex l to vertex k. Put $E = \{E_{k,l}\}_{k,l \in V}$. Assume that $\cup_{l=1}^{K} E_{k,l} \neq \emptyset$ for each k, although some of $E_{k,l}$'s may be empty. Let $E_{i,j}^{k}$ be the set of sequences of k edges $(e_1, e_2, \ldots, e_k)$ which is a directed path from vertex j to vertex i. We say that the graph is transitive if, for any i, j, there is a positive integer p such that $E_{i,j}^{p}$ is non-empty.

Definition 1 (Graph-directed sets). *Let (V, E) be a transitive directed graph. For each $e \in E_{k,l}$, let S_e be a contraction on a compact space. A K-tuple of non-empty compact sets $(F_1, F_2, \ldots, F_K)$ is called a family of graph-directed sets if it*

satisfies

$$F_k = \bigcup_{l=1}^{K} \bigcup_{e \in E_{k,l}} S_e(F_l) \quad (k = 1, \ldots, K).$$

If we put

$$\{S_e : e \in E_{k,l}\} = \{S_i^{kl} : i = 1, \ldots, n_{kl}\} \quad (k, l = 1, \ldots, K),$$

the definition above can be stated in the following form.

Definition 2. *Put*

$$\mathcal{S} = \begin{pmatrix} \{S_i^{11}\}_{i=1}^{n_{11}} & \{S_i^{12}\}_{i=1}^{n_{12}} & \cdots & \{S_i^{1K}\}_{i=1}^{n_{1K}} \\ \cdots & \cdots & \cdots & \cdots \\ \{S_i^{K1}\}_{i=1}^{n_{K1}} & \{S_i^{K2}\}_{i=1}^{n_{K2}} & \cdots & \{S_i^{KK}\}_{i=1}^{n_{KK}} \end{pmatrix},$$

where each S_i^{kl} is a contraction on a compact space, $n_{kl} \geq 0$ and $\sum_{l=1}^{K} n_{kl} > 0$ $(k = 1, \ldots, K)$. Assume that the matrix $\{n_{kl}\}_{k,l=1,\ldots,K}$ is irreducible. A K-tuple of sets $(F_1, \ldots, F_K)$ is called a family of graph-directed sets for $\mathcal{S}$ if

$$F_k = \textstyle\bigcup_{i=1}^{n_{k1}} S_i^{k1}(F_1) \cup \cdots \cup \bigcup_{i=1}^{n_{kK}} S_i^{kK}(F_K) \quad (k = 1, \ldots, K).$$

We have the following theorem.

Theorem 1. ([3], [5], [7], [8], [9]) *Let $K \geq 2$ and let $\mathcal{S}$ be as above. Then there is a unique K-tuple of non-empty compact graph-directed sets $(F_1, \ldots, F_K)$.*

We explain the random iteration algorithm with an example.

Example 2. Let T_i $(i = 1, 2, 3, 4)$ be a contraction, which is the similarity (dilation) that maps the whole square $\mathbf{X} = [0, 1] \times [0, 1]$ to the corresponding square in Fig. 2. Consider a pair of graph-directed sets (A, B) defined by

$$A = S_1^{11}(A) \cup S_1^{12}(B) \cup S_2^{12}(B),$$
$$B = S_1^{21}(A) \cup S_2^{21}(A) \cup S_1^{22}(B).$$

Here, each S_i^{kl} is defined as $S_1^{11} = T_2, S_1^{12} = T_1, S_2^{12} = T_4, S_1^{21} = T_1, S_2^{21} = T_4$ and $S_1^{22} = T_3$.

Let $x_1(0)$ and $x_2(0)$ be arbitrary points in $\mathbf{X}$ and choose randomly, recursively and independently

$$x_1(t+1) \in \{S_1^{11}(x_1(t)), S_1^{12}(x_2(t)), S_2^{12}(x_2(t))\},$$
$$x_2(t+1) \in \{S_1^{21}(x_1(t)), S_2^{21}(x_1(t)), S_1^{22}(x_2(t))\}.$$

The probabilities for selecting $\{S_1^{11}(x_1(t)), S_1^{12}(x_2(t)), S_2^{12}(x_2(t))\}$ as $x_1(t+1)$ and $\{S_1^{21}(x_1(t)), S_2^{21}(x_1(t)), S_1^{22}(x_2(t))\}$ as $x_2(t+1)$ are $(p_1^{11}, p_1^{12}, p_2^{12}) = (1/2, 1/4, 1/4)$ and $(p_1^{21}, p_2^{21}, p_1^{22}) = (1/4, 1/2, 1/4)$, respectively. Starting with $x_1(0) = (0, 0)$ and $x_2(0) = (0, 0)$, we obtained the pair of figures (A', B') in Fig. 2 after 10000 times loop.

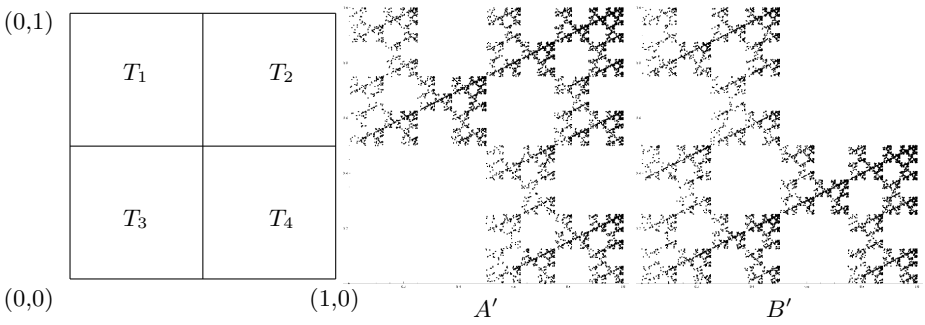


Fig. 2. An example of random iteration algorithm for graph-directed sets.

We will subsequently show that there is a unique pair of probability measures (μ_1, μ_2) on the pair of graph-directed sets (A, B) in Example 2 which satisfies

$$\mu_1 = p_1^{11}\ \mu_1 \circ (S_1^{11})^{-1} + \sum_{i=1}^{2} p_i^{12}\ \mu_2 \circ (S_i^{12})^{-1},$$

$$\mu_2 = \sum_{i=1}^{2} p_i^{21}\ \mu_1 \circ (S_i^{21})^{-1} + p_1^{22}\ \mu_2 \circ (S_1^{22})^{-1}.$$

For μ_1 and μ_2, it holds that for all $(x_1(0), x_2(0)) \in \mathbf{X} \times \mathbf{X}$,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=0}^{n-1} f(x_1(t)) = \int_{\mathbf{X}} f(x) d\mu_1(x),$$

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=0}^{n-1} f(x_2(t)) = \int_{\mathbf{X}} f(x) d\mu_2(x),$$

for almost all sequences $\{(x_1(t), x_2(t)) : t = 0, 1, \dots\}$, and for any continuous real function f on $\mathbf{X}$. In fact, for a unique probability measure $\tilde{\mu}$ on $\mathbf{X} \times \mathbf{X}$, it holds that for any $(x_1(0), x_2(0)) \in \mathbf{X} \times \mathbf{X}$,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=0}^{n-1} f(x_1(t), x_2(t)) = \int_{\mathbf{X}\times\mathbf{X}} f(x_1, x_2) d\tilde{\mu}(x_1, x_2) \quad \text{a.e.}$$

for any continuous real function f on $\mathbf{X} \times \mathbf{X}$. The measures μ_1 and μ_2 are the marginal distributions of the measure $\tilde{\mu}$ on $\mathbf{X} \times \mathbf{X}$.

Now, we state our *random iteration algorithm for a family of graph-directed sets*. Let $\mathbf{X}$ be a non-empty compact set in $\mathbf{R}^d$ such that $S_i^{kl}(\mathbf{X}) \subset \mathbf{X}$, for $k, l = 1, \dots, K, i = 1, \dots, n_{kl}$. A closed sphere $B(0, r)$ in $\mathbf{R}^d$ with a sufficiently large $r > 0$ such that $S_i^{kl}(B(0, r)) \subset B(0, r)$ for any k, l, i is an example of $\mathbf{X}$. For $k = 1, \dots, K$, let $(p_1^{k1}, \dots, p_{n_{k1}}^{k1}, \dots, p_1^{kK}, \dots, p_{n_{kK}}^{kK})$ be a system of probabilities

assigned to $\{S_1^{k1}, \ldots, S_{n_{k1}}^{k1} \ldots, S_1^{kK}, \ldots, S_{n_{kK}}^{kK}\}$, where $p_i^{kl} \geq 0$ $(i = 1, \ldots, n_{k1})$ for $l = 1, \ldots, K$ and $\sum_{l=1}^{K} \sum_{i=1}^{n_{kl}} p_i^{kl} = 1$.

Choose $(x_1(0), \ldots, x_K(0)) \in \mathbf{X}^K$, and choose randomly, recursively and independently

$$x_k(t+1) \in \{S_i^{kl}(x_l(t)) : l = 1, \ldots, K \text{ for which } n_{kl} > 0 \text{ and } i = 1, \ldots, n_{kl}\},$$

for $k = 1, \ldots, K$. The probability for the event $x_k(t+1) = S_i^{kl}(x_l(t))$ is p_i^{kl}. This produces a sequence of K-tuples of points $\{(x_1(t), \ldots, x_K(t)) : t = 0, 1, \ldots\}$.

3 Invariant probability measure

Barnsley and Elton have shown the following.

Theorem 2. (Barnsley and Elton: [1], [2], [6]) *Let Y be a complete metric space. Let $\{T_1, \ldots, T_N\}$ be a family of Lipschitz maps on Y. Let $(p_1, \ldots, p_N)$ be a system of probabilities assigned to $\{T_1, \ldots, T_N\}$, where $p_i > 0$ $(i = 1, \ldots, N)$ and $\sum_{i=1}^{N} p_i = 1$. Suppose there exists $0 < r < 1$ such that*

$$\prod_{i=1}^{N} d(T_i(y), T_i(z))^{p_i} \leq r \; d(y, z)$$

for $y, z \in Y$.

Choose $y(0) \in Y$ and choose randomly, recursively and independently, $y(t) \in \{T_1(y(t-1)), \ldots, T_N(y(t-1))\}$, where the probability for the event $\{y(t) = T_i(y(t-1))\}$ is p_i. Then the following hold.

(1) There is a unique invariant probability measure μ associated with transition probability $p(y, B) = \sum_{i=1}^{N} p_i 1_B(T_i(y))$, that is, $\mu(B) = \int p(y, B) d\mu(y)$ for all Borel set B.

(2) Let P be a probability $\prod_{i=1}^{\infty} P_i$ on $\prod_{i=1}^{\infty} J_i$, where $P_i = (p_1, \ldots, p_N)$ and $J_i = \{1, \ldots, N\}$. It holds that for any $y(0) \in Y$,

$$\lim_{n \to \infty} \frac{1}{n} \sum_{t=0}^{n-1} f(y(t)) = \int_Y f(y) d\mu(y) \; P\text{−a.e.}$$

for all continuous function $f : Y \to \mathbf{R}$.

Let us note that μ is an invariant probability measure if and only if $\mu = M(\mu)$ for the Markov operator

$$M(\nu) = \sum_{i=1}^{N} p_i \, \nu \circ T_i^{-1}.$$

By applying Barnsley and Elton's theorem, we show the uniqueness of an invariant probability measure of a random iteration algorithm for a family of

graph-directed sets. Recall that $\mathbf{X}$ is a non-empty compact set in $\mathbf{R}^d$ such that $S_i^{kl}(\mathbf{X}) \subset \mathbf{X}$ for $k, l = 1, \ldots, K, i = 1, \ldots, n_{kl}$. Put $\mathbf{X}_k = \mathbf{X}$ for $k = 1, \ldots, K$, and define $\mathbf{X}^K = \mathbf{X}_1 \times \cdots \mathbf{X}_K$. Define a metric d on $\mathbf{X}^K$ by

$$d((x_1, \ldots, x_K), (y_1, \ldots, y_K)) = \mathrm{Max}\{|x_k - y_k| : k \ = 1, \ldots, K\},$$

where $|x_k - y_k|$ denotes the d-dimensional Euclidean metric.

Put $I_k = \{(l_k, i_k) : n_{kl_k} > 0,\ 1 \leq i_k \leq n_{kl_k}\} \subset \{1, \ldots, K\} \times \mathbf{N}$ for $k = 1, \ldots, K$. Put further $I = I_1 \times \cdots \times I_K$. For $S_i^{kl} : \mathbf{X} \to \mathbf{X}$, where $k = 1, \ldots, K$ and $(l, i) \in I_k$, let $\tilde{S}_i^{kl} : \mathbf{X}^K \to \mathbf{X}_k$ be defined by $\tilde{S}_i^{kl}(x_1, \ldots, x_K) = S_i^{kl}(x_l)$.

For $((l_1, i_1), \ldots, (l_K, i_K)) \in I$, a transformation $T_{((l_1,i_1),\ldots,(l_K,i_K))} : \mathbf{X}^K \to \mathbf{X}^K$ is defined by

$$\begin{aligned} T_{((l_1,i_1),\ldots,(l_K,i_K))}(x_1, \ldots, x_K) &:= (\tilde{S}_{i_1}^{1l_1}(x_1, \ldots, x_K), \ldots, \tilde{S}_{i_K}^{Kl_K}(x_1, \ldots, x_K)) \\ &= (S_{i_1}^{1l_1}(x_{l_1}), \ldots, S_{i_K}^{Kl_K}(x_{l_K})) \end{aligned}$$

with the associated probability

$$p_{((l_1,i_1),\ldots,(l_K,i_K))} = p_{i_1}^{1l_1} \cdots p_{i_K}^{Kl_K}.$$

We apply Barnsley and Elton's theorem to $Y = \mathbf{X}^K$ and

$$\mathcal{T} = \{T_{((l_1,i_1),\ldots,(l_K,i_K))} : \ ((l_1, i_1), \ldots, (l_K, i_K)) \in I\}$$

with probabilities $p_{i_1}^{1l_1} \cdots p_{i_K}^{Kl_K}$. Let L be the set of functions as defined below.

$$\begin{aligned} L = \{f : \mathbf{X}^K \to \mathbf{R} :& \\ &|f(x_1, \ldots, x_K) - f(y_1, \ldots, y_K)| \leq \mathrm{Max}\{|x_k - y_k| : k \ = 1, \ldots, K\}\}, \end{aligned}$$

where $|x_k - y_k|$ denotes the d-dimensional Euclidean metric.

Let $\mathbf{P}(\mathbf{X}^K)$ be the space of normalized Borel measures on $\mathbf{X}^K$. The Hutchinson metric d_H of $\mathbf{P}(\mathbf{X}^K)$ is defined by

$$d_H(\mu, \nu) = \mathrm{Sup}\Big\{ \int f d\mu - \int f d\nu : f \in L \Big\}.$$

It is well known that $(\mathbf{P}(\mathbf{X}^K), d_H)$ is a compact space. (See Barnsley [1].)

Let us define a Markov operator $M : \mathbf{P}(\mathbf{X}^K) \to \mathbf{P}(\mathbf{X}^K)$, and prove a theorem which claims the existence of a certain measure.

Definition 3. *The Markov operator associated with*

$$\mathcal{T} = \{T_{((l_1,i_1),\ldots,(l_1,i_1))} : ((l_1, i_1), \ldots, (l_K, i_K)) \in I\}$$

is a transformation $M : \mathbf{P}(\mathbf{X}^K) \to \mathbf{P}(\mathbf{X}^K)$ *defined by*

$$M(\nu) = \sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I} \prod_{k=1}^{K} p_{i_k}^{kl_k} \nu \circ (T_{((l_1,i_1),\ldots,(l_K,i_K))})^{-1}.$$

Theorem 3. *There exists a unique probability measure $\tilde{\mu}$ on $\mathbf{X}^K$ such that $\tilde{\mu} = M(\tilde{\mu})$.*

Proof (Proof1: Application of Barnsley and Elton's criterion). Recall that, for $((l_1, i_1), \ldots, (l_K, i_K)) \in I$,

$$T_{((l_1,i_1),\ldots,(l_K,i_K))}(x_1, \ldots, x_K) = (S_{i_1}^{1l_1}(x_{l_1}), \ldots, S_{i_K}^{Kl_K}(x_{l_K})).$$

Let s be the maximum of the contraction ratios of $\{S_i^{kl}\}$. Note that $s < 1$. Recall that $d((x_1, \ldots, x_K), (y_1, \ldots, y_K)) = \mathrm{Max}\{|x_k - y_k| : k = 1, \ldots, K\}$, where $|x_k - y_k|$ denotes the d-dimensional Euclidean metric. Then it holds that

$$\begin{aligned}
&d(T_{((l_1,i_1),\ldots,(l_K,i_K))}(x_1, \ldots, x_K)), T_{((l_1,i_1),\ldots,(l_K,i_K))}(y_1, \ldots, y_K))) \\
&\quad = d((S_{i_1}^{1l_1}(x_{l_1}), \ldots, S_{i_K}^{Kl_K}(x_{l_K})), (S_{i_1}^{1l_1}(y_{l_1}), \ldots, S_{i_K}^{Kl_K}(y_{l_K}))) \\
&\quad = \mathrm{Max}\{|S_{i_1}^{1l_1}(x_{l_1}) - S_{i_1}^{1l_1}(y_{l_1})|, \ldots, |S_{i_K}^{Kl_K}(x_{l_K}) - S_{i_K}^{Kl_K}(y_{l_K})|\} \\
&\quad \le s\mathrm{Max}\{|x_{l_1} - y_{l_1}|, \ldots, |x_{l_K} - y_{l_K}|\} \\
&\quad \le s\mathrm{Max}\{|x_1 - y_1|, \ldots, |x_K - y_K|\}. \qquad (1)
\end{aligned}$$

The Barnsley and Elton's condition holds if $d(T_i(x), T_i(y)) \le sd(x, y)$ for an $s < 1$. From (1) above this criterion is satisfied, and so we can apply the Barnsley and Elton's theorem and obtain the desired measure. □

Proof (Proof2: Direct proof). Notice that, for $f \in L$,

$$\begin{aligned}
&\left|f(T_{((l_1,i_1),\ldots,(l_K,i_K))}(x_1, \ldots, x_K)) - f(T_{((l_1,i_1),\ldots,(l_K,i_K))}(y_1, \ldots, y_K))\right| \\
&\quad = \left|f(S_{i_1}^{1l_1}(x_{l_1}), \ldots, S_{i_K}^{Kl_K}(x_{l_K})) - f(S_{i_1}^{1l_1}(y_{l_1}), \ldots, S_{i_K}^{Kl_K}(y_{l_K}))\right| \\
&\quad \le \mathrm{Max}\{|S_{i_1}^{1l_1}(x_{l_1}) - S_{i_1}^{1l_1}(y_{l_1})|, \ldots, |S_{i_K}^{Kl_K}(x_{l_K}) - S_{i_K}^{Kl_K}(y_{l_K})|\} \\
&\quad \le s\mathrm{Max}\{|x_{l_1} - y_{l_1}|, \ldots, |x_{l_K} - y_{l_K}|\} \\
&\quad \le s\mathrm{Max}\{|x_1 - y_1|, \ldots, |x_K - y_K|\}.
\end{aligned}$$

Define

$$\hat{f}(x_1, \ldots, x_K) = s^{-1} \sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I} \prod_{k=1}^{K} p_{i_k}^{kl_k} f(T_{((l_1,i_1),\ldots,(l_K,i_K))}(x_1, \ldots, x_K)).$$

Then

$$\begin{aligned}
&\left|\hat{f}(x_1, \ldots, x_K) - \hat{f}(y_1, \ldots, y_K)\right| \\
&\quad \le s^{-1} \sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I} \prod_{k=1}^{K} p_{i_k}^{kl_k} s\ \mathrm{Max}\{|x_1 - y_1|, \ldots, |x_K - y_K|\} \\
&\quad \le \mathrm{Max}\{|x_1 - y_1|, \ldots, |x_K - y_K|\},
\end{aligned}$$

since $\sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I}\prod_{k=1}^{K} p_{i_k}^{kl_k} = 1$. It therefore follows that $\hat{f} \in L$. If we put $\hat{L} = \{\hat{f}(x_1,\ldots,x_K) : f \in L\}$, then $\hat{L} \subset L$ holds.

By the definition,

$$\begin{aligned}
d_H(M(\mu),M(\nu)) &= \mathrm{Sup}\Big\{ \int f dM(\mu) - \int f dM(\nu) : f \in L \Big\} \\
&= \mathrm{Sup}\Big\{ \int \sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I} \prod_{k=1}^{K} p_{i_k}^{kl_k} \\
&\qquad f(T_{((l_1,i_1),\ldots,(l_K,i_K))}(x_1,\ldots,x_K))d\mu(x_1,\ldots,x_K) \\
&\quad - \int \sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I} \prod_{k=1}^{K} p_{i_k}^{kl_k} \\
&\qquad f(T_{((l_1,i_1),\ldots,(l_K,i_K))}(x_1,\ldots,x_K)d\nu(x_1,\ldots,x_K) : f \in L \Big\} \\
&= \mathrm{Sup}\Big\{ s\Big(\int \hat{f}(x_1,\ldots,x_K)d\mu(x_1,\ldots,x_K) \\
&\qquad - \int \hat{f}(x_1,\ldots,x_K)d\nu(x_1,\ldots,x_K)\Big) : \hat{f} \in \hat{L} \Big\} \\
&\le \mathrm{Sup}\Big\{ s\Big(\int f(x_1,\ldots,x_K)d\mu(x_1,\ldots,x_K) \\
&\qquad - \int f(x_1,\ldots,x_K)d\nu(x_1,\ldots,x_K)\Big) : f \in L \Big\} \\
&= s\ d_H(\mu,\nu).
\end{aligned}$$

Therefore the Markov operator M is a contraction map on $\mathbf{P}(\mathbf{X}^K)$. This implies that there is a unique invariant probability measure $\tilde{\mu}$ in $\mathbf{P}(\mathbf{X}^K)$. □

Barnsley and Elton's theorem for random iterated algorithms can be extended to a family of graph-directed sets.

Theorem 4. *Let $\tilde{\mu}$ be the unique invariant probability measure claimed in Theorem 3. Then for any $(x_1(0),\ldots,x_K(0)) \in \mathbf{X}^K$,*

$$\lim_{n\to\infty} \frac{1}{n}\sum_{t=0}^{n-1} f(x_1(t),\ldots,x_K(t)) = \int_{\mathbf{X}^K} f(x_1,\ldots,x_K)d\tilde{\mu}(x_1,\ldots,x_K) \quad \text{a.e.}$$

for all continuous function $f : \mathbf{X}^K \to \mathbf{R}$.

Proof. We apply (2) of Barnsley and Elton's theorem to $T_{((l_1,i_1),\ldots,(l_K,i_K))}$ on $\mathbf{X}^K$ with probabilities $\prod_{k=1}^{K} p_{i_k}^{kl_k}$. □

Corollary 1. *(1) For the marginal distributions $\tilde{\mu}_1,\ldots,\tilde{\mu}_K$, it holds that*

$$\tilde{\mu}_k = \sum_{l=1}^{K}\sum_{i=1}^{n_{kl}} p_i^{kl}\tilde{\mu}_l \circ (S_i^{kl})^{-1}$$

for $k = 1,\ldots,K$.

(2) For any $(x_1(0), \ldots, x_K(0)) \in \mathbf{X}^K$,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=0}^{n-1} g(x_k(t)) = \int_{\mathbf{X}} g(x) d\tilde{\mu}_k(x) \quad \text{a.e.}$$

for all continuous function $g : \mathbf{X} \to \mathbf{R}$ *and for* $k = 1, \ldots, K$.

Proof. Proof of (1). Note that for a family of Borel sets $A_1, \ldots, A_K$ in $\mathbf{X}$, it holds that

$$\begin{aligned}
&(T_{((l_1,i_1),\ldots,(l_K,i_K))})^{-1}(A_1 \times \cdots \times A_K) \\
&\qquad = \{(x_1, \ldots, x_K) \ : \ \tilde{S}^{kl_k}_{i_k}(x_1, \ldots, x_K) \in A_k, k = 1, \ldots, K\} \\
&\qquad = \bigcap_{k=1}^{K} (\tilde{S}^{kl_k}_{i_k})^{-1}(A_k).
\end{aligned}$$

So we have

$$(T_{((l_1,i_1),\ldots,(l_K,i_K))})^{-1}(\mathbf{X}_1 \times \cdots \times \mathbf{X}_{k-1} \times A_k \times \mathbf{X}_{k+1} \cdots \times \mathbf{X}_K) = (\tilde{S}^{kl_k}_{i_k})^{-1}(A_k),$$

because $(\tilde{S}^{jl_j}_{i_j})^{-1}(\mathbf{X}_j) = \mathbf{X}^K$. Recall that $\mathbf{X}_l = \mathbf{X}$ for all l. Note that $\tilde{\mu} = M(\tilde{\mu})$. Then it holds that

$$\begin{aligned}
\tilde{\mu}_k(A) &= \tilde{\mu}(\mathbf{X}_1 \times \cdots \times \mathbf{X}_{k-1} \times A \times \mathbf{X}_{k+1} \cdots \times \mathbf{X}_K) \\
&= M(\tilde{\mu})(\mathbf{X}_1 \times \cdots \times \mathbf{X}_{k-1} \times A \times \mathbf{X}_{k+1} \cdots \times \mathbf{X}_K) \\
&= \sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I} \prod_{j=1}^{K} p^{jl_j}_{i_j} \\
&\qquad \tilde{\mu}((T_{((l_1,i_1),\ldots,(l_K,i_K))})^{-1}(\mathbf{X}_1 \times \cdots \times \mathbf{X}_{k-1} \times A \times \mathbf{X}_{k+1} \cdots \times \mathbf{X}_K)) \\
&= \sum_{((l_1,i_1),\ldots,(l_K,i_K))\in I} \prod_{j=1}^{K} p^{jl_j}_{i_j} \tilde{\mu}((\tilde{S}^{kl_k}_{i_k})^{-1}(A)) \\
&= \sum_{(l_k,i_k)\in I_k} p^{kl_k}_{i_k} \tilde{\mu}((\tilde{S}^{kl_k}_{i_k})^{-1}(A)) \prod_{j\neq k} \sum_{(l_j,i_j)\in I_j} p^{jl_j}_{i_j} \\
&= \sum_{(l_k,i_k)\in I_k} p^{kl_k}_{i_k} \tilde{\mu}((\tilde{S}^{kl_k}_{i_k})^{-1}(A)) \\
&= \sum_{(l_k,i_k)\in I_k} p^{kl_k}_{i_k} \tilde{\mu}_{l_k}((\tilde{S}^{kl_k}_{i_k})^{-1}(A)).
\end{aligned}$$

This proves the assertion (1).

Proof of (2). Define $f(x_1, \ldots, x_K) = g(x_k)$. Then by virtue of Theorem 4, it holds that

$$\lim_{n\to\infty}\frac{1}{n}\sum_{t=0}^{n-1} g(x_k(t)) = \lim_{n\to\infty}\frac{1}{n}\sum_{t=0}^{n-1} f(x_1(t),\ldots,x_K(t))$$
$$= \int_{\mathbf{X}^K} f(x_1,\ldots,x_K)d\tilde{\mu}(x_1,\ldots,x_K) \quad \text{a.e.}$$
$$= \int_{\mathbf{X}} g(x)d\tilde{\mu}_k(x).$$

We thus have

$$\lim_{n\to\infty}\tfrac{1}{n}\textstyle\sum_{t=0}^{n-1} g(x_k(t)) = \int_{\mathbf{X}} g(x)d\tilde{\mu}_k(x) \quad \text{a.e.}$$

for all continuous function $g : \mathbf{X} \to \mathbf{R}$ and k=1, ..., K.

This proves the assertion (2).

□

Theorem 5. *Let $\tilde{\mu}$ be the unique probability measure in Theorem 3, and let $\tilde{\mu}_1,\ldots,\tilde{\mu}_K$ be the marginal distributions of $\tilde{\mu}$. Then for $m = 1,\ldots,K$, the support of $\tilde{\mu}_m$ is F_m, where $(F_1,\ldots,F_K)$ is the family of graph-directed sets in Theorem 1.*

Proof. The proof is analogous to that of Theorem 2 in Section 9.6 of [1].

Let A denote the support of $\tilde{\mu}$. Notice that

$$T_{((l_1,i_1),\ldots,(l_K,i_K))}(F_1\times\cdots\times F_K) \subset F_1\times\cdots\times F_K$$

for any $((l_1,i_1),\ldots,(l_K,i_K)) \in I$. It follows that $\{T_{((l_1,i_1),\ldots,(l_K,i_K))}\}$ restricted on $F_1\times\cdots\times F_K$ defines a random iteration algorithm with the probabilities $\prod_{k=1}^{K} p_{i_k}^{kl_k}$. Let $\tilde{\nu}$ be an invariant probability measure for the restricted random iteration algorithm, and this $\tilde{\nu}$ is an invariant probability measure for the random iteration algorithm on $\mathbf{X}^K$. Since $\tilde{\mu}$ is unique, $\tilde{\mu} = \tilde{\nu}$. It follows that $A \subset F_1 \times \cdots \times F_K$, and so the support of $\tilde{\mu}_m$ is included in F_m.

For $m = 1,\ldots,K$, let Σ_m be the set of sequences $\{(l_1,i_1;\ldots;l_n,i_n;\ldots) : n_{l_{n-1}\ l_n} > 0,\ 1 \le i_n \le n_{l_{n-1}\ l_n} \ \text{ for } \ n = 1,\ldots\}$, where $l_0 = m$.

For each point $a \in F_m$, there is a (not necessarily unique) sequence in Σ_m such that

$$a \in S_{i_1}^{ml_1} \circ S_{i_2}^{l_1l_2} \circ \cdots \circ S_{i_n}^{l_{n-1}l_n}(\mathbf{X}_{l_n})$$

holds for all n. Let O be an open set in $\mathbf{X}$ which contains a. By the fact that S_i^{kl} is a contraction, there is a positive integer n such that

$$S_{i_1}^{ml_1} \circ S_{i_2}^{l_1l_2} \circ \cdots \circ S_{i_n}^{l_{n-1}l_n}(\mathbf{X}_{l_n})) \subset O.$$

Note that $\tilde{\mu}_m(S_{i_1}^{ml_1} \circ S_{i_2}^{l_1l_2} \circ \cdots \circ S_{i_n}^{l_{n-1}l_n}(\mathbf{X}_{l_n})) \ge \prod_{j=1}^{n} p_{i_j}^{l_{j-1}l_j} > 0$. It holds that $\tilde{\mu}_m(O) > 0$, and so F_m is included in the support of $\tilde{\mu}_m$. □

Remark 1. In the above proofs we have not used the independence of choosing $\{S_{i_1}^{1l_1}, \ldots, S_{i_K}^{Kl_K}\}$, or the productivity of the probabilities $\prod_{k=1}^{K} p_{i_k}^{kl_k}$. So we can formulate the random iteration algorithm so that the probability of choosing $\{S_{i_1}^{1l_1}, \ldots, S_{i_K}^{Kl_K}\}$ can be expressed as $p_{(l_1,i_1;\ldots,l_K,i_K)}$, which is not restricted to the independent case of $p_{i_1}^{1l_1} \ldots p_{i_K}^{Kl_K}$. Theorems 3, 4 and 5 hold for thus modified random iteration algorithm.

Remark 2. We propose a variation of this algorithm which changes only one coordinate X_k on each step. Let $\{q_1, \ldots, q_K\}$ be a probability, that is, $q_k > 0$ for $k = 1, \ldots, K$ and $\sum_{k=1}^{K} q_k = 1$. For $k = 1, \ldots, K$, let $(p_1^{k1}, \ldots, p_{n_{k1}}^{k1}, \ldots, p_1^{kK}, \ldots, p_{n_{kK}}^{kK})$ be a system of probabilities defined in Section 2.

Choose $(x_1(0), \ldots, x_K(0)) \in \mathbf{X}^K$. Next choose randomly $k(1) \in \{1, \ldots, K\}$, with probability $q_{k(1)}$, and then choose randomly $S_i^{k(1)l}(x_l(0))$ for $l = 1, \ldots, K$ with $n_{k(1)l} > 0$ and $1 \leq i \leq n_{k(1)l}$, with probability $p_i^{k(1)l}$. Let $x_{k(1)}(1) = S_i^{k(1)l}(x_l(0))$ and $x_j(1) = x_j(0)$ for $j \neq k(1)$. Continue this procedure recursively and independently.

So we have

$$x_{k(t+1)}(t+1) = S_i^{k(t+1)l}(x_l(t)),$$
$$x_j(t+1) = x_j(t) \text{ for } j \neq k(t+1),$$

with probability $q_{k(t+1)} p_i^{k(t+1)l}$, where $k(t+1) = 1, \ldots, K$, $l = 1, \ldots, K$ with $n_{k(t+1)l} > 0$ and $1 \leq i \leq n_{k(t+1)l}$.

This produces a sequence of K-tuples of points $\{(x_1(t), \ldots, x_K(t)) : t = 0, 1, \ldots\}$. We then have the following results.

(1) There exists a unique probability measure $\hat{\mu}$ on $\mathbf{X}^K$ such that $\hat{\mu} = \hat{M}(\hat{\mu})$, where $\hat{M}$ is the associated Markov operator.

(2) Let $\hat{\mu}_1, \ldots, \hat{\mu}_K$ be the marginal distributions of $\hat{\mu}$. Then for $m = 1, \ldots, K$, the support of $\hat{\mu}_m$ is F_m, where $(F_1, \ldots, F_K)$ is the family of graph-directed sets in Theorem 1.

(3) For any $(x_1(0), \ldots, x_K(0)) \in \mathbf{X}^K$,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=0}^{n-1} f(x_1(t), \ldots, x_K(t)) = \int_{\mathbf{X}^K} f(x_1, \ldots, x_K) d\hat{\mu}(x_1, \ldots, x_K) \quad \text{a.e.}$$

for all continuous function $f : \mathbf{X}^K \to \mathbf{R}$.

(4) (i) For the marginal distributions $\hat{\mu}_1, \ldots, \hat{\mu}_K$, it holds that

$$\hat{\mu}_k = \sum_{l=1}^{K} \sum_{i=1}^{n_{kl}} p_i^{kl} \hat{\mu}_l \circ (S_i^{kl})^{-1}$$

for $k = 1, \ldots, K$.

(ii) For any $(x_1(0), \ldots, x_K(0)) \in \mathbf{X}^K$,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=0}^{n-1} g(x_k(t)) = \int_{\mathbf{X}} g(x) d\hat{\mu}_k(x) \quad \text{a.e.}$$

for all continuous function $g : \mathbf{X} \to \mathbf{R}$ and for $k = 1, \ldots, K$.

Acknowledgement This work has been supported in part by Research Grant from KSU(2008, 282), Research Grant from KSU(2008, 339), JSPS Grant-in-Aid No. 20540143, and JSPS Grant-in-Aid No. 18500013, respectively.

References

1. M. Barnsley, Fractals everywhere. *Academic Press, INC*, Boston, 2000.
2. M. Barnsley and J. Elton, A new class of Markov processes for image encoding. *Adv. Appl. Prob.*, 20, 14-32, 1988.
3. T. Bedford, Dimension and dynamics of fractal recurrent sets. *J. London Math. Soc. (2)*, 33:89-100, 1986.
4. V. Brattka, Some Notes on Fine Computability. *Journal of Universal Computer Science*, 8:382-395, 2002.
5. G.A. Edgar, Measure, Topology and Fractal Geometry. *Springer-Verlag*, 1990.
6. J.H. Elton, An ergodic theorem for iterated maps. *Ergodic Theory and Dynamical Systems*, 7:481-488, 1987.
7. K. J. Falconer, Fractal Geometry. *John Wiley & Sons*, Chichester, 1990.
8. J. E. Hutchinson, Fractals and self similarity. *Indiana Univ. Math. J.*,30:713-747, 1981.
9. R.D. Mauldin and S.C. Williams, Hausdorff dimension in graph directed constructions. *Trans. Amer. math. Soc*,309:811-829, 1988.
10. Y. Tsujii, T.Mori, Y.Yasugi and H.Tsuiki, Fine-Continuous Functions and Fractals Defined by Infinite Systems of Contractions. Proceeding Volume ILC2007 (to appear).
11. I.Werner, Contractive Markov Systems. J. London Math. Soc. (2) 71:236-258, 2005.

Computable Separation in Topology, from T_0 to T_3

Klaus Weihrauch

University of Hagen, Universitätsstr. 1, 58097 Hagen, Germany,
klaus.weihrauch@fernuni-hagen.de

Abstract. This article continues the study of computable elementary topology started in [7]. We introduce a number of computable versions of the topological T_0 to T_3 separation axioms and solve their logical relation completely. In particular, it turns out that computable T_1 is equivalent to computable T_2. The strongest axiom SCT_3 is used in [2] to construct a computable metric.

1 Preliminaries

We use the representation approach to computable analysis [6] as the basis for our investigation. In particular, we use the terminology and concepts introduced in [7] (which can be considered as a revision and extension of parts from [6]).

Let Σ^* and Σ^ω be the sets of the finite and infinite sequences, respectively, of symbols from a finite alphabet Σ. A function mapping finite or infinite sequences of symbols from Σ is computable, if it can be computed by a Type-2 machine, that is, a Turing machine with finite or infinite input and output tapes. On Σ^* and Σ^ω we use canonical tupling functions $\langle\cdot\rangle$ that are computable and have computable inverses. Computability on finite or infinite sequences of symbols is transferred to other sets by representations, where elements of Σ^* or Σ^ω are used as "concrete names" of abstract objects. For representations $\gamma_i :\subseteq Y_i \to M_i$ we consider the product representation defined by $[\gamma_1,\gamma_2]\langle p,q\rangle := (\gamma_1(p_1),\gamma_2(p_2))$. Let $Y = Y_1 \times \ldots \times Y_n$, $M = M_1 \times \ldots \times M_n$ and $\gamma : Y \to M$, $\gamma(y_1,\ldots,y_n) = \gamma_1(y_1)\times\ldots\times\gamma_n(y_n)$. A partial function $h :\subseteq Y \to Y_0$ realizes the multi-function $f : M \rightrightarrows M_0$ if $\gamma_0 \circ h(y) \in f(x)$ whenever $x = \gamma(y)$ and $f(x)$ exists. This means that $h(y)$ is a name of some $z \in f(x)$ if y is a name of $x \in \mathrm{dom}(f)$. The function f is (γ,γ_0)-computable, if it has a computable realization.

We will consider computable topological spaces as defined in [7]. Various similar definitions have been used, see, for example, [4, 3, 5] and the references in [7]. In particular, the definition in [6] is slightly different. A computable topological space is a 4-tuple $\mathbf{X} = (X,\tau,\beta,\nu)$ such that (X,τ) is a topological T_0-space, $\nu :\subseteq \Sigma^* \to \beta$ is a notation of a base β of τ, $\mathrm{dom}(\nu)$ is recursive and $\nu(u)\cap\nu(v) = \bigcup\{\nu(w) \mid (u,v,w) \in S\}$ for all $u,v \in \mathrm{dom}(\nu)$ for some r.e. set $S\subseteq(\mathrm{dom}(\nu))^3$.

For the points, the open sets and the closed sets we use the representations δ, θ and ψ^- that are defined as follows. For $p \in \Sigma^\omega$ and $x \in X$, $\delta(p) = x$ iff p is

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 257-268
http://drops.dagstuhl.de/opus/volltexte/2009/2276

a list of all $u \in \mathrm{dom}(\nu)$ such that $x \in \nu(u)$, $\theta(p)$ is the union of all $\nu(u)$ where u is listed by p, and $\psi^-(p) := X \setminus \theta(p)$.

2 Axioms of Computable Separation

For a topological space $\mathbf{X} = (X, \tau)$ with set $\mathcal{A}$ of closed sets we consider the following classical separation properties:

Definition 1 (separation axioms).

$$\mathrm{T}_0 : (\forall x, y \in X,\ x \neq y)(\exists W \in \tau)((x \in W \wedge y \notin W) \vee (x \notin W \wedge y \in W))),$$
$$\mathrm{T}_1 : (\forall x, y \in X,\ x \neq y)(\exists W \in \tau)(x \in W \wedge y \notin W),$$
$$\mathrm{T}_2 : (\forall x, y \in X,\ x \neq y)(\exists U, V \in \tau)(U \cap V = \emptyset \wedge x \in U \wedge y \in V),$$
$$\mathrm{T}_3 : (\forall x \in X, \forall A \in \mathcal{A}, x \notin A)(\exists U, V \in \tau)(U \cap V = \emptyset \wedge x \in U \wedge A \subseteq V),$$
$$\mathrm{T}_4 : (\forall A, B \in \mathcal{A}, A \cap B = \emptyset)(\exists U, V \in \tau)(U \cap V = \emptyset \wedge A \subseteq U \wedge B \subseteq V).$$

For $i = 0, 1, 2, 3$, we call $\mathbf{X} = (X, \tau)$ a T_i-space iff $\mathrm{T_i}$ *is true.*

For the four axioms, $\mathrm{T}_2 \Longrightarrow \mathrm{T}_1 \Longrightarrow \mathrm{T}_0$ and $\mathrm{T}_0 + \mathrm{T}_3 \Longrightarrow \mathrm{T}_2$, where all the implications are proper [1]. T_2-spaces are called *Hausdorff spaces* and T_3-spaces are called *regular*. (Many authors, for example [1], call a space T_3-space or regular iff $\mathrm{T}_1 + \mathrm{T}_3$.) We mention that (X, τ) is a T_1-space, iff all sets $\{x\}$ $(x \in X)$ are closed [1]. For computable topological spaces $\mathbf{X} = (X, \tau, \beta, \nu)$, which are countably based T_0-spaces (also called *second countable*), $T_3 \Longrightarrow T_2$.

We introduce computable versions $\mathrm{CT_i}$ of the conditions $\mathrm{T_i}$ by requiring that the existing open neighborhoods can be computed. For the points we compute basic neighborhoods.

Definition 2 (axioms of computable separation). *For $i \in \{0, 1, 2, 3\}$ define conditions* $\mathrm{CT_i}$ *as follows.*

CT_0 : *The multi-function t_0 is (δ, δ, ν)-computable where t_0 maps each $(x, y) \in X^2$ such that $x \neq y$ to some $U \in \beta$ such that*

$$(x \in U \text{ and } y \notin U) \text{ or } (x \notin U \text{ and } y \in U). \tag{1}$$

CT_1 : *The multi-function t_1 is (δ, δ, ν)-computable, where t_1 maps each $(x, y) \in X^2$ such that $x \neq y$ to some $U \in \beta$ such that $x \in U$ and $y \notin U$.*

CT_2 : *The multi-function t_2 is $(\delta, \delta, [\nu, \nu])$-computable, where t_2 maps each $(x, y) \in X^2$ such that $x \neq y$ to some $(U, V) \in \beta^2$ such that $U \cap V = \emptyset$, $x \in U$ and $y \in V$.*

CT_3 : *The multi-function t_3 is $(\delta, \psi^-, [\nu, \theta])$-computable, where t_3 maps each (x, A) such that $x \in X$, $A \subseteq X$ closed, and $x \notin A$ to some $(U, V) \in \beta \times \tau$ such that $U \cap V = \emptyset$, $x \in U$ and $A \subseteq V$.*

Obviously, $\mathrm{CT_i}$ implies $\mathrm{T_i}$. We introduce some further computable T_i-conditions.

Definition 3 (further axioms of computable separation).

WCT_0 : *There is an r.e. set* $H \subseteq \mathrm{dom}(\nu) \times \mathrm{dom}(\nu)$ *such that*

$$(\forall x, y,\ x \neq y)(\exists (u,v) \in H)(x \in \nu(u)\ \wedge\ y \in \nu(v)) \quad \textit{and} \tag{2}$$

$$(\forall (u,v) \in H) \begin{cases} \nu(u) \cap \nu(v) = \emptyset \\ \vee\ (\exists x)\, \nu(u) = \{x\} \subseteq \nu(v) \\ \vee\ (\exists y)\, \nu(v) = \{y\} \subseteq \nu(u)\,. \end{cases} \tag{3}$$

SCT_0 : *The multi-function* t_0^s *is* $(\delta, \delta, [\nu_{\mathbb{N}}, \nu])$*-computable where* t_0^s *maps each* $(x,y) \in X^2$ *such that* $x \neq y$ *to some* $(k, U) \in \mathbb{N} \times \beta$ *such that* $(k = 1,\ x \in U$ *and* $y \notin U)$ *or* $(k = 2,\ x \notin U$ *and* $y \in U)$*.*

CT_0' : *There is an r.e. set* $H \subseteq \mathrm{dom}(\nu_{\mathbb{N}}) \times \mathrm{dom}(\nu) \times \mathrm{dom}(\nu)$ *such that*

$$(\forall x, y,\ x \neq y)(\exists (w,u,v) \in H)(x \in \nu(u)\ \wedge\ y \in \nu(v)) \quad \textit{and} \tag{4}$$

$$(\forall (w,u,v) \in H) \begin{cases} \nu(u) \cap \nu(v) = \emptyset \\ \vee\ \nu_{\mathbb{N}}(w) = 1 \wedge (\exists x)\, \nu(u) = \{x\} \subseteq \nu(v) \\ \vee\ \nu_{\mathbb{N}}(w) = 2 \wedge (\exists y)\, \nu(v) = \{y\} \subseteq \nu(u)\,. \end{cases} \tag{5}$$

CT_1' : *There is an r.e. set* $H \in \Sigma^* \times \Sigma^*$ *such that*

$$(\forall x, y,\ x \neq y)(\exists (u,v) \in H)(x \in \nu(u) \wedge y \in \nu(v)) \quad \textit{and} \tag{6}$$

$$(\forall (u,v) \in H) \begin{cases} \nu(u) \cap \nu(v) = \emptyset \\ \vee\ (\exists x)\, \nu(u) = \{x\} \subseteq \nu(v)\,. \end{cases} \tag{7}$$

CT_2' : *There is an r.e. set* $H \in \Sigma^* \times \Sigma^*$ *such that*

$$(\forall x, y,\ x \neq y)(\exists (u,v) \in H)(x \in \nu(u) \wedge y \in \nu(v)) \quad \textit{and} \tag{8}$$

$$(\forall (u,v) \in H) \begin{cases} \nu(u) \cap \nu(v) = \emptyset \\ \vee\ (\exists x)\, \nu(u) = \{x\} = \nu(v)\,. \end{cases} \tag{9}$$

SCT_2 : *There is an r.e. set* $H \in \Sigma^* \times \Sigma^*$ *such that*

$$(\forall x, y,\ x \neq y)(\exists (u,v) \in H)(x \in \nu(u) \wedge y \in \nu(v)) \quad \textit{and} \tag{10}$$

$$(\forall (u,v) \in H)\, \nu(u) \cap \nu(v) = \emptyset\,. \tag{11}$$

CT_3' : *The multi-function* t_3' *is* $(\delta, \nu, [\nu, \psi^-])$*-computable where* t_3' *maps each* $(x, W) \in X \times \beta$ *such that* $x \in W$ *to some* (U, B) *such that* $U \in \beta$, $B \subseteq X$ *is closed and* $x \in U \subseteq B \subseteq W$*.*

WCT_3 : *The multi-function* t_3^w *is* (δ, ν, ν)*-computable where* t_3^w *maps each* $(x, W) \in X \times \beta$ *such that* $x \in W$ *to some* U *such that* $U \in \beta$ *and* $x \in U \subseteq \overline{U} \subseteq W$*.*

SCT_3: *There are an r.e. set* $R \subseteq \mathrm{dom}(\nu) \times \mathrm{dom}(\nu)$ *and a computable function* $r :\subseteq \Sigma^* \times \Sigma^* \to \Sigma^\omega$ *such that for all* $u, w \in \mathrm{dom}(\nu)$*,*

$$\nu(w) = \bigcup \{\nu(u) \mid (u,w) \in R\}\,, \tag{12}$$

$$(u,w) \in R \Longrightarrow\ \nu(u) \subseteq \psi^- \circ r(u,w) \subseteq \nu(w)\,. \tag{13}$$

CT_0', CT_1' and CT_2' are versions of CT_0, CT_1 and CT_2, respectively, where base sets are used instead of points (see Theorem 1 below). Similarly, SCT_3 is a pointless version of CT_3'. In contrast to CT_0, in SCT_0 the separating function gives immediate information about the direction of the separation. Also in CT_0' some information about the direction of the separation is included while no such information is given in its weak version WCT_0. The strong version SCT_2 results from CT_2' by excluding the case $(\exists x)\,\nu(u) = \{x\} = \nu(v)$. Notice that SCT_2 results also from WCT_0, CT_0' and CT_1' by excluding the corresponding cases. The following examples illustrate the definitions. Further examples can be found in Section 4.

Example 1. 1. Consider the computable real line $\mathbf{R} := (\mathbb{R}, \tau_{\mathbb{R}}, \beta, \nu)$ such that $\tau_{\mathbb{R}}$ is the real line topology and ν is a canonical notation of the set of all open intervals with rational endpoints. $\mathbf{R}$ is SCT_3 (easy proof).
2. (T_0 but not WCT_0) Consider the computable lower real line $\mathbf{R}_< := (\mathbb{R}, \tau_<, \beta_<, \nu_<)$, defined by $\nu_<(w) := (\nu_{\mathbb{Q}}; \infty)$, which is T_0 but not T_1. Suppose $\mathbf{R}_<$ is WCT_0. Since for any two base elements U, V, U is not a singleton and $U \cap V \neq \emptyset$, $H = \emptyset$ by (3). But $H \neq \emptyset$ by (2).
3. (T_1 but not T_2 or WCT_0) Let $\mathbf{X} = (\mathbb{N}, \tau, \beta, \nu)$ such that $\tau = \beta$ is the set of cofinite subsets of $\mathbb{N}$ and ν is a canonical notation of ν. Then $\mathbf{X}$ is a computable topological space. It is T_1 since singletons $\{x\}$ are closed. Suppose $\mathbf{X}$ is WCT_0. Since the intersection of base elements cannot be empty and singletons are not open the set H in (3) must be empty. But then (2) cannot be true. The space is not T_2 since the intersection of any two non-empty open set is not empty.

By the next lemma the above computable separation axioms are robust, that is, they do not depend on the notation ν of the base explicitly but only on the computability concept on the points induced by it. Call the computable topological spaces $\mathbf{X} = (X, \tau, \beta, \nu)$ and $\widetilde{\mathbf{X}} = (X, \tau, \widetilde{\beta}, \widetilde{\nu})$ equivalent, iff $\delta \equiv \widetilde{\delta}$ [7, Definition 21 and Theorem 22].

Lemma 1. *1. For $i \in \{0,1,2,3\}$ let $\overline{\mathrm{CT}}_\mathrm{i}$ be the condition obtained from CT_i and let $\overline{\mathrm{SCT}}_0$ be the condition obtained from SCT_0 by replacing β and ν by τ and θ, respectively. Then $\overline{\mathrm{CT}}_\mathrm{i} \iff \mathrm{CT}_\mathrm{i}$ and $\overline{\mathrm{SCT}}_0 \iff \mathrm{SCT}_0$.*
2. Let $\widetilde{\mathbf{X}} = (X, \tau, \widetilde{\beta}, \widetilde{\nu})$ be a computable topological space equivalent to $\mathbf{X} = (X, \tau, \beta, \nu)$. Then each separation axiom from Definitions 2 and 3 for $\mathbf{X}$ is equivalent to the corresponding axiom for $\widetilde{\mathbf{X}}$.

The proofs are straightforward. In particular, apply [7, Theorem 22] by which "equivalence" is equivalent to ($\nu \leq \widetilde{\theta}$ and $\widetilde{\nu} \leq \theta$).

3 Implications

In this section we prove the implications between the separation properties, in the next section we give counterexamples for the proper ones.

Theorem 1.

1. $\mathrm{SCT}_3 \Longrightarrow \mathrm{CT}_3 \Longrightarrow \mathrm{SCT}_2 \Longrightarrow \mathrm{CT}_2 \Longrightarrow \mathrm{CT}_0 \Longrightarrow \mathrm{WCT}_0$,
2. $CT_3 \iff CT_3' \Longrightarrow \mathrm{WCT}_3$,
3. $CT_2 \iff CT_2' \iff CT_1 \iff CT_1'$,
4. $CT_0 \iff SCT_0 \iff CT_0'$,

The proofs of $SCT_0 \Longrightarrow CT_0'$ and $\mathrm{CT}_3' \Longrightarrow \mathrm{SCT}_2$ need some care. They are based on the observation that a realizing machine needs only finitely many steps for finding an appropriate base element for the result. We omit the details (approximately 2 pages).

Surprisingly, computable T_1-spaces are exactly computable T_2. We add some further interesting results. Let "D" be the axiom stating that the topological space is discrete.

Theorem 2. *For computable topological spaces,*

1. *if* $\{x\}$ *is not open for all* $x \in X$ *then* $\mathrm{WCT}_0 \Longrightarrow \mathrm{SCT}_2$,
2. SCT_2 *if* T_2 *and* $\{(u,v) \mid \nu(u) \cap \nu(v) = \emptyset\}$ *is r.e.*,
3. $\mathrm{SCT}_2 \iff (x \neq y$ *is* (δ,δ)*-r.e.*$)$,
4. $\mathrm{CT}_3 \Longrightarrow \mathrm{SCT}_3$ *if the set* $\{w \in \Sigma^* \mid \nu(w) \neq \emptyset\}$ *is r.e.*
5. $\mathrm{D} \Longrightarrow \mathrm{WCT}_3$

We include only the proof of 4. For the terminology see [7].

Proof: Since finite intersection is computable, there is a computable function g such that $\bigcap \nu^{\mathrm{fs}}(w) = \theta \circ g(w)$. Therefore, the set $\{w \in \Sigma^* \mid \bigcap \nu^{\mathrm{fs}}(w) \neq \emptyset\}$ is r.e. There is a machine M such that f_M realizes the multi-function t_3'. If $x = \delta(p) \in \nu(w)$ then for some $u_1 \in \mathrm{dom}(\nu)$ and $q \in \mathrm{dom}(\psi^-)$, $f_M(p,w) = \langle u_1, q\rangle = \iota(u_1)q$ such that

$$x \in \nu(u_1) \subseteq \psi^-(q) \subseteq \nu(w)\,. \tag{14}$$

For computing $\iota(u_1)$ some prefix $u_0 \in \mathrm{dom}(\nu^{\mathrm{fs}}) \cap \Sigma^*11$ of p suffices. Since $\delta(p) \in \nu(w)$ we may assume $w \ll u_0$. Since $x \in \delta[u_0 11 \Sigma^\omega] = \bigcap \nu^{\mathrm{fs}}(u_0)$, $\bigcap \nu^{\mathrm{fs}}(u_0) \neq \emptyset$. We will compute $\bigcap \nu^{\mathrm{fs}}(u_0) \cap \nu(u_1)$ as a union $\bigcup\{\nu(u) \mid u \in L\}$ of base sets and add all these (u,w) to R.

There is a machine N that works on input (u,w) as follows:
(S1) If $u, w \in \mathrm{dom}(\nu)$, $\nu(u) \neq \emptyset$ and $\nu(w) \neq \emptyset$ then
(S2) N searches for words $u_0 \in \mathrm{dom}(\nu^{\mathrm{fs}}) \cap \Sigma^*11$ and $u_1 \in \mathrm{dom}(\nu)$ such that $w \ll u_0$, M on input $(u_0 1^\omega, w)$ writes $\iota(u_1)$ in at most $|u_0|$ steps and $u \ll g(u_0\iota(u_1))$,
(S3) and then writes all words $\iota(v)$ for which there are words u_2, u_3 such that $u_0u_2 \in \mathrm{dom}(\nu^{\mathrm{fs}})$, $\bigcap \nu^{\mathrm{fs}}(u_0u_2) \neq \emptyset$, the machine M on input $(u_0u_2 1^\omega, w)$ writes $\iota(u_1)u_3$ in at most $|u_0u_2|$ steps and $v \ll 11u_3$. (In order to guarantee an infinite output, N writes 11 from time to time.)
(S4) If (1) is false or the search in (2) is not successful then N computes forever without writing. Let $r := f_N$ and $R := \mathrm{dom}(f_N)$. Then $R \subseteq \mathrm{dom}(\nu) \times \mathrm{dom}(\nu)$ and R is r.e. We must prove correctness.

We show (12): Suppose $x = \delta(p) \in \nu(w)$. Then for some u_1, q, $f_M(p, w) = \iota(u_1)q$, hence for some prefix $u_0 \sqsubseteq p$ such that $w \ll u_0$ and $u_0 \in \Sigma^*11$ (since we my assume that p has the subword 11 infinitely often), M on input $(u_0 1^\omega, w)$ writes $\iota(u_1)$ in at most $|u_0|$ steps. Since $x \in \bigcap \nu^{\rm fs}(u_0)$ and $x \in \nu(u_1)$ by (14), $x \in \theta \circ g(u_0\iota(u_1))$, hence $x \in \nu(u)$ for some $u \ll g(u_0\iota(u_1))$. Therefore, there is some u such that $x \in \nu(u)$ and the machine N on input (u, w) will find some words such that (S2) is true. Therefore $x \in \nu(u)$ for some $(u, w) \in R$, hence "$\supseteq$" is true in (12).

On the other hand, suppose $(u, w) \in R$ and $x \in \nu(u)$ for some x. Then on input (u, w) the machine N finds words u_0, u_1 such that the conditions in (S2) above are true. Since $u \ll g(u_0\iota(u_1))$ and $w \ll u_0$, $x \in \nu(u) \subseteq \bigcap \nu^{\rm fs}(u_0) \subseteq \nu(w)$. Therefore, "$\subseteq$" is true in (12).

For showing (13) suppose $(u, w) \in R$ and $x \in \nu(u)$ for some x again. Then on input (u, w) the machine N finds words u_0, u_1 such that the conditions in (S2) above are true. Since $x \in \bigcap \nu^{\rm fs}(u_0)$, $x = \delta(u_0 p')$ for some $p' \in \Sigma^\omega$. Since $x \in \nu(w)$, $f_M(u_0 p', w) = \langle u_1, q\rangle = \iota(u_1)q$ for some $q \in \Sigma^\omega$ such that (14). Suppose $v \ll q$. Then for some u_2, u_3 such that $u_0u_2 \in {\rm dom}(\bigcap \nu^{\rm fs})$, the machine M on input $(u_0u_2 1^\omega, w)$ writes $\iota(u_1)u_3$ in at most $|u_0u_2|$ steps and $v \ll \iota(u_1)u_3$, therefore, $v \ll r(u, w)$. By (14),

$\nu(w)^c \subseteq \theta(q) = \bigcup\{\nu(v) \mid v \ll q\} \subseteq \bigcup\{\nu(v) \mid v \in r(u, w)\} = \theta \circ r(u, w)$.
This proves $\psi^- \circ r(u, w) \subseteq \nu(w)$ in(13).

Finally let v be some word such that $\iota(v)$ is listed by the machine N on input (u, w), that is, $v \ll r(u, w)$. Then there are words u_2, u_3 such that $\bigcap \nu^{\rm fs}(u_0u_2) \neq \emptyset$, the machine M on input $(u_0u_2 1^\omega, w)$ writes $\iota(u_1)u_3$ in at most $|u_0u_2|$ steps and $v \ll 11u_3$. Since $\bigcap \nu^{\rm fs}(u_0u_2) \neq \emptyset$ and $w \ll u_0$, there is some p' such that $\delta(u_0u_2p') \in \nu(w)$ and $f_M(u_0u_2p', w) = \iota(u_1)u_3q'$ for some q'. By (14) $\nu(u_1) \cap \theta(u_3q') = \emptyset$. Since $\nu(u) \subseteq \nu(u_1)$ (by $u \ll g(u_0\iota(u_1))$ in (S2)) and $\nu(v) \subseteq \theta(u_3q')$ (since $v \ll u_3$), $\nu(u) \cap \nu(v) = \emptyset$.

Since this is true for all $v \ll r(u, w)$, $\nu(u) \cap \theta \circ r(u, w) = \emptyset$, hence $\nu(u) \subseteq \psi^- \circ r(u, w)$.

Therefore, we have also proved (13). □

4 Counterexamples

A topological space is discrete iff every singleton $\{x\}$ is open iff every subset $B \subseteq X$ is open. A discrete space is T_i for $i = 0, \ldots, 4$. Let "D" be the axiom stating that the topological space is discrete. Counterexamples show that the implications in Theorem 1.1 are proper. Since this is an extended abstract we include only two of them.

Theorem 3. *For computable topological spaces,*

$\mathrm{T}_0 \not\Longrightarrow \mathrm{WCT}_0$	*by Example 1.2;*
$\mathrm{T}_1 \not\Longrightarrow \mathrm{WCT}_0$	*by Example 1.3;*
$\mathrm{D} \not\Longrightarrow \mathrm{WCT}_0$	*by Example 2;*
$\mathrm{D} + \mathrm{WCT}_0 \not\Longrightarrow \mathrm{CT}_0$	*by Example 3;*
$\mathrm{D} + \mathrm{CT}_0 \not\Longrightarrow \mathrm{CT}_1$	*by Example 4;*
$\mathrm{D} + \mathrm{CT}_2 \not\Longrightarrow \mathrm{SCT}_2$	*by Example 5;*
$\mathrm{WCT}_3 + \mathrm{CT}_2 \not\Longrightarrow \mathrm{SCT}_2$	*by Example 5;*
$\mathrm{T}_4 + \mathrm{SCT}_2 \not\Longrightarrow \mathrm{WCT}_3$	*by Example 7;*
$\mathrm{SCT}_2 \not\Longrightarrow \mathrm{T}_3$	*by Example 6;*
$\mathrm{CT}_3 \not\Longrightarrow \mathrm{SCT}_3$	*by Example 8.*

In the following examples let $(a_i)_{i\in\mathbb{N}}$, $(b_i)_{i\in\mathbb{N}}$, ..., $(e_i)_{i\in\mathbb{N}}$ be injective families with pairwise disjoint ranges and let $\{0,1,\ldots,7\}\subseteq\Sigma$.

Example 2. (D but not WCT_0) Omitted. □

Example 3. ($D + WCT_0$ but not CT_0) Let $A\subseteq\mathbb{N}$ be some non-r.e. set. Let $X := \{a_i, b_i \mid i \in \mathbb{N}\}$ and let τ be the discrete topology on X. Below we will define sets $B, C, D\subseteq\mathbb{N}$ such that $\{A, B, C, D\}$ is a partition of $\mathbb{N}$. Define a notation ν of a basis β of the topology as follows.

	0^i1	0^i2	0^i3	0^i12	0^i13	0^i23
$i \in A \cup D$	$\{a_i\}$	$\{b_i\}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$i \in B$	$\{a_i\}$	$\{a_i, b_i\}$	$\{b_i\}$	$\{a_i\}$	$\emptyset$	$\{b_i\}$
$i \in C$	$\{a_i, b_i\}$	$\{b_i\}$	$\{a_i\}$	$\{b_i\}$	$\{a_i\}$	$\emptyset$

Since $\nu(0^ik)\cap\nu(0^im) = \nu(0^ikm)$, $\nu(u)\cap\nu(v) = \nu\circ g(u,v)$ for some computable function g. Therefore $\mathbf{X} := (X, \tau, \beta, \nu)$ is a computable topological space. Let $H := \{(0^ik, 0^jl) \mid i, j \in \mathbb{N};\ k, l \in \{1, 2\};\ (i \neq j \vee k \neq l\}$. Then H satisfies (2) and (3) for the space $\mathbf{X}$. Therefore, $\mathbf{X}$ is a WCT_0-space.

We show that $\mathbf{X}$ is not SCT_0.

Let $l, r \in \Sigma^*$ such that $\nu_{\mathbb{N}}(l) = 1$ and $\nu_{\mathbb{N}}(r) = 2$. We assume w.l.o.g. that $\nu_{\mathbb{N}}$ is injective. For $i \in \mathbb{N}$ let

$$S_i := \{\langle l, 0^i1\rangle,\ \langle r, 0^i3\rangle,\ \langle l, 0^i12\rangle,\ \langle r, 0^i23\rangle\},$$
$$T_i := \{\langle r, 0^i2\rangle,\ \langle l, 0^i3\rangle,\ \langle r, 0^i12\rangle,\ \langle l, 0^i13\rangle\}.$$

Suppose, the function $f :\subseteq \Sigma^\omega \times \Sigma^\omega \to \Sigma^*$ realizes the separation function t_0^s for $\mathbf{X}$. If $\delta(p) = a_i$ and $\delta(q) = b_i$ then

$$f(p,q) \in \begin{cases} S_i \text{ if } & i \in B \\ T_i \text{ if } & i \in C \end{cases} \tag{15}$$

since $\nu(u)$ must be either $\{a_i\}$ or $\{b_i\}$ if $f(p,q) = \langle w, u\rangle$. Notice that $S_i \cap T_i = \emptyset$.

For all $i \in \mathbb{N}$ define $p_i, q_i \in \Sigma^\omega$ by $p_i := \iota(0^i1)\iota(0^i1)\iota(0^i1)\dots$ and $q_i := \iota(0^i2)\iota(0^i2)\iota(0^i2)\dots$. Let F be the set of all computable functions $f :\subseteq \Sigma^\omega \times \Sigma^\omega \to \Sigma^*$ such that $f(p_i, q_i)$ exists for all $i \in A$. Consider $f \in F$. Then $f' : i \mapsto f(p_i, q_i)$ is computable such that $A \subseteq \mathrm{dom}(f')$. Since A is not r.e. and $\mathrm{dom}(f')$ is r.e., $\mathrm{dom}(f') \setminus A$ is infinite. Since F is countable, there is a bijective function $g : E \to F$ for some $E \subseteq \mathbb{N}$ such that $i \in \mathrm{dom}(g'_i) \setminus A$ for all $i \in E$ ($g_i := g(i)$). Then $A \cap E = \emptyset$.

For each $i \in E$ we put i to B or C in such a way that g_i does not realize the separating function t^s_0 for SCT_0.

$$B := \{i \in E \mid g_i(p_i, q_i) \notin S_i\},$$
$$C := \{i \in E \mid g_i(p_i, q_i) \in S_i\},$$

and $D := \mathbb{N} \setminus (A \cup B \cup C)$. Since $A \cap E = \emptyset$, $E = B \cup C$ and $B \cap C = \emptyset$, $\{A, B, C, D\}$ is a partition of $\mathbb{N}$.

Suppose some computable function f realizes t^s_0. Since for $i \in A, \delta(p_i) = a_i$ and $\delta(q_i) = b_i$, $f(p_i, q_i)$ exists for all $i \in A$, hence $f = g_i$ for some $i \in E$.

If $i \in B$ then $g_i(p_i, q_i) \notin S_i$, hence by (15) the function g_i does not realize t^s_0. If $i \in C$ then $g_i(p_i, q_i) \in S_i$, hence not in T_i since $S_i \cap T_i = \emptyset$. By (15) the function g_i does not realize t^s_0.

From this contradiction we conclude that $\mathbf{X}$ is not SCT_0. By Theorem 1 $\mathbf{X}$ is not CT_0. □

Example 4. (D and CT_0 but not CT_1) Omitted. □

Example 5. (D and CT_2 but not SCT_2) Let $A \subseteq \mathbb{N}$ be an r.e. set with non-r.e. complement. Define a notation ν by
$\nu(0^i1) := \{a_i\}, \nu(0^i2) := \{a_i\}$ for $i \in A$,
$\nu(0^i1) := \{a_i\}, \nu(0^i2) := \{b_i\}$ for $i \notin A$
for all $i \in \mathbb{N}$. Then ν is a notation of a base β of a topology (the discrete topology) τ on a subset $X \subseteq \mathbb{N}$ such that $\mathbf{X} = (X, \tau, \beta, \nu)$ is a computable topological space.

The space $\mathbf{X}$ is T_i for $i = 0, \dots, 4$ since it is discrete. It is CT_2 but not SCT_2: The set $H := \{(0^ik, 0^jl) \mid i, j \in \mathbb{N},\ k, l \in \{1, 2\}\}$ satisfies CT'_2. Therefore, the space is CT_2. Suppose SCT_2. Let H be the r.e. set for SCT_2. By (10), $i \notin A \implies (0^i1, 0^i2) \in H$ and by (11), $i \in A \implies (0^i1, 0^i2) \notin H$. Since H is r.e., the complement of A must be r.e. (contradiction). Notice that $x \neq y$ is not (δ, δ)-r.e., see Theorem 2.3. It can be shown easily that $\mathbf{X}$ is WCT_3. □

Example 6. (SCT_2 but not T_3) Omitted. □

Example 7. (T_4 and SCT_2 but not WCT_3) Omitted. □

Example 8. (CT_3 but not SCT_3) Define a notation I of the open rational intervals by $I\langle u, v\rangle := (\nu_{\mathbb{Q}}(u); \nu_{\mathbb{Q}}(v)) \subseteq \mathbb{R}$. Let $\mathbb{R}_c \subseteq \mathbb{R}$ be the set of (ρ-) computable real numbers. There is a computable function $g : \Sigma^* \to \Sigma^*$ such that $\mathbb{R}_c \subseteq \bigcup_{i\in\mathbb{N}} I \circ g(0^i)$ and $\sum_{i\in\mathbb{N}} \mathrm{length}(I \circ g(0^i)) < 1$ [6, Theorem 4.2.8]. Let $z := \inf\{a \in \mathbb{Q} \mid [a; 1] \subseteq \bigcup_{i\in\mathbb{N}} I \circ g(0^i)\}$. Then $0 < z < 1$, z is $\rho_>$-computable and not ρ-computable, hence not $\rho_<$-computable [6]. Furthermore for all k, $z \notin I \circ g(0^k)$.

Let $X := \mathbb{R}_c \cup \{z\}$. Define a notation ν of subsets of X by $\nu(0v) := I(v) \cap X$ and $\nu(1v) := I(v) \cap (-\infty; z) \cap X$ ($v \in \mathrm{dom}(I)$).Then $\beta := \mathrm{range}(\nu)$ is a base of a topology τ such that $\mathbf{X} := (X, \tau, \beta, \nu)$ is a computable topological space. Notice that for $x < z$, $z \in \mathrm{cls}_X((x; z) \cap X)$. Let δ be the inner representation for the points of $\mathbf{X}$.

Proposition 1: The multi-function $h : x \rightrightarrows a$ mapping each $x \in X$ such that $x < z$ to some $a \in \mathbb{Q}$ such that $x < a < z$ is $(\delta, \nu_{\mathbb{Q}})$-computable.

Proof 1: If $x < z$ and $x \in I \circ g(0^k)$, then $\sup I \circ g(0^k) < z$, since $z \not\leq \inf I \circ g(0^k)$ (since $x < z$), $z \notin I \circ g(0^k)$ and $z \neq \sup I \circ g(0^k)$ (since $z \notin \mathbb{Q}$). There is a machine M that on input p searches for some $k \in \mathbb{N}$ such that $0g(0^k) \ll p$ and writes some u such that $\nu_{\mathbb{Q}}(u) = \sup I \circ g(0^k)$. Let $\delta(p) = x < z$. Since $x \in \mathbb{R}_c$, there is some k such that $x \in I \circ g(0^k)$, hence $0g(0^k) \ll p$. We obtain $\nu_{\mathbb{Q}} \circ f_M(p) < z$. Therefore, the multi-function h is $(\delta, \nu_{\mathbb{Q}})$-computable.

Proposition 2: The multi-function $f : (x, U) \rightrightarrows V$ mapping each $(x, (a; b)) \in X \times \mathrm{range}(I)$ such that $x \in (a; b)$ to some $(c; d) \in \mathrm{range}(I)$ such that $x \in (c; d) \subseteq [c; d] \subseteq (a; b)$ is (δ, I, I)-computable.

Proof 2: Every δ-name of x lists arbitrarily short rational intervals containing x. Search for a sufficiently short interval $(c; d)$.

We show that t_3' from Definition 3 is computable. Suppose $x \in W \in \beta$. If $W = \nu(0w) = I(w) \cap X$ for some w then $W' := I(w)$. If $W = \nu(1w) = I(w) \cap (-\infty; z) \cap X$ for some w then by means of h find some $e \in \mathbb{Q}$ such that $x < e < z$ and let $W' := I(w) \cap (-\infty; e)$. Then $x \in W' \cap X \subseteq W$. By means of f from x and $(a; b) := W'$ find $(c; d) \in \mathrm{range}(I)$ such that $x \in (c; d) \subseteq [c; d] \subseteq (a; b)$. Then $x \in (c; d) \cap X \subseteq [c; d] \cap x \subseteq W$.

From a, b, c and d some u and q can be computed such that $\nu(u) = (c; d) \cap X$ and $\psi^-(q) = [c; d] \cap X$. Then $x \in \nu(u) \subseteq \psi^-(q) \subseteq W$. Therefore, t_3' is $(\delta, \nu, [\nu, \psi^-])$-computable.

Suppose, $\mathbf{X}$ is SCT_3. Let R be the r.e. set for SCT_3 from Definition 3. There is some w such that $\nu(w) = (0; z) \cap X$. Suppose $(u, w) \in R$. Then $\nu(u) \subseteq \nu(w)$, hence for some $a, b \in \mathbb{Q}$ such that $a < b < z$, $\nu(u) = (a; b) \cap X$ or $\nu(u) = (a; z) \cap X$. If $\nu(u) = (a; z) \cap X$, then $z \in \mathrm{cls}_X(\nu(u))$, but $\mathrm{cls}_X(\nu(u)) \subseteq \nu(w) = (0; z)$ by SCT_3, hence $z \in \nu(w) = (0; z)$ (contradiction). Therefore, $\sup \nu(u) = (a; b)$ for some rational numbers a, b such that $a < b < z$.

The function $U \mapsto \sup U$ for all $U = (a; x) \in \beta$ such that $x < z$ is $(\nu, \nu_{\mathbb{Q}})$-computable. Since R is r.e., the number $y := \sup\{\sup \nu(u) \mid (u, w) \in R\}$ is $\rho_<$-computable such that $y \leq z$. Since $(0; z) = \nu(w) = \bigcup_{(u,w) \in R} \nu(u)$, for every $x < z$ there is some $(u, w) \in R$ such that $x < \sup \nu(u)$. Therefore, $y = z$, hence z is $\rho_<$-computable. Contradiction. Therefore, $\mathbf{X}$ is not SCT_3. Notice that $U \neq \emptyset$ is not ν-r.e. □

Further results can be obtained in combination with the positive results from Theorem 1. Figure 1 visualizes the interplay between the computable versions of T_i for $i = 0, 1, 2, 3$ from Definitions 2 and 3 we have proved. "$A \longrightarrow B$" means $A \Longrightarrow B$, "$A \not\longrightarrow B$" means that we have constructed a computable topological space for which $A \wedge \neg B$, and $A \stackrel{C}{\not\longrightarrow} B$" means that we have constructed a

computable topological space for which $(A \wedge C) \wedge \neg B$. Remember that $\mathrm{SCT}_0 \iff \mathrm{CT}_0 \iff \mathrm{CT}'_0$, $\mathrm{CT}_1 \iff \mathrm{CT}'_1 \iff \mathrm{CT}_2 \iff \mathrm{CT}'_2$ and $\mathrm{CT}_3 \iff \mathrm{CT}'_3$.

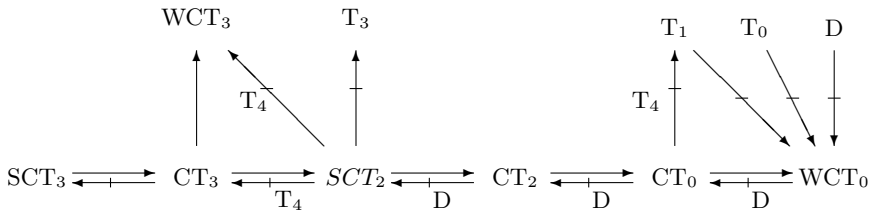


Fig. 1. The relation between computable T_0-, T_1-, T_2- and T_3-separation.

5 Further Results

For a computable topological space $\mathbf{X} = (X, \tau, \beta, \nu)$ and $B \subseteq X$ the subspace $\mathbf{X}_B = (B, \tau_B, \beta_B, \nu_B)$ of $\mathbf{X}$ to B is the computable topological space defined by $\mathrm{dom}(\nu_B) := \mathrm{dom}(\nu)$, $\nu_B(w) := \nu(w) \cap B$. The separation axioms from Definitions 2 and 3 are invariant under restriction to subspaces.

Theorem 4. *If a computable topological space satisfies some separation axiom from Definitions 2 and 3 then each subspace satisfies this axiom.*

Proof: Straightforward. □

The product of two T_i-spaces is a T_i-space for $i = 0, 1, 2, 3$. This is no longer true for some of the computable separation axioms. By definition for the product $\mathbf{X}_1 \times \mathbf{X}_2 = \overline{\mathbf{X}} = (X_1 \times X_2, \overline{\tau}, \overline{\beta}, \overline{\nu})$ of two computable topological spaces $\mathbf{X}_1 = (X_1, \tau_1, \beta_1, \nu_1)$ and $\mathbf{X}_2 = (X_2, \tau_2, \beta_2, \nu_2)$, $\overline{\nu}\langle u_1, u_2\rangle = \nu_1(u_1) \times \nu_2(u_2)$.

Example 9. The space $\mathbf{X}$ from Example 5 is CT_2 but not SCT_2. Let $\mathbf{R}$ be the computable real line from Example 1.1. We show that the product $\mathbf{X} \times \mathbf{R}$ is not WCT_0. Suppose, $\mathbf{X} \times \mathbf{R}$ is WCT_0. Since every base element of $\mathbf{X} \times \mathbf{R}$ has the form $\nu(u) \times (a; b)$ $(a, b \in \mathbb{Q},\ a < b)$ no singleton $\{(x, y)\}$ $(x \in X,\ y \in \mathbb{R})$ is open. By Theorem 2.1, $\mathbf{X} \times \mathbf{R}$ is SCT_2. By Theorem 1 the relation $(x, x') \neq (y, y')$ is $([\delta, \rho], [\delta, \rho])$-r.e. where δ is the inner representation of the points of $\mathbf{X}$. There is a machine M that halts on input $(\langle p, p'\rangle, \langle q, p'\rangle)$ for $p, q \in \mathrm{dom}(\delta)$ and $p' \in \mathrm{dom}(\rho)$ iff $\delta(p) \neq \delta(q)$. There is a computable element $p' \in \mathrm{dom}(\rho)$. Therefore, there is a machine N that halts on input (p, q) iff $\delta(p) \neq \delta(q)$, hence $x \neq y$ is (δ, δ)-r.e. By Theorem 1, $\mathbf{X}$ must be SCT_2. But $\mathbf{X}$ is not SCT_2.

Theorem 5. *1. The SCT_2-, WCT_3-, CT_3- and SCT_3-spaces are closed under product.*
2. The WCT_0-, CT_0- and CT_2-spaces are not closed under product.

Proof: 1. Suppose, $\mathbf{X}_1$ and $\mathbf{X}_2$ are SCT_2. By Theorem 1, $x_i \neq y_i$ is (δ_i, δ_i)-r.e. for $i = 1, 2$, hence $(x_1, x_2) \neq (y_1, y_2)$ is $([\delta_1, \delta_2], [\delta_1, \delta_2])$-r.e., hence again by Theorem 1, $\mathbf{X}_1 \times \mathbf{X}_2$ is SCT_2.

Suppose, $\mathbf{X}_1$ and $\mathbf{X}_2$ are WCT_3. Let $(x_1, x_2) \in W_1 \times W_2$. From x_i and W_i we can find $U_i \in \beta_i$ such that $x_i \in U_i \subseteq \overline{U}_i \subseteq W_i$ (for $i = 1, 2$). Then $(x_1, x_2) \in U_1 \times U_2 \subseteq \overline{U_1 \times U_2} = \overline{U}_1 \times \overline{U}_2 \subseteq W_1 \times W_2$.

Suppose, $\mathbf{X}_1$ and $\mathbf{X}_2$ are CT_3'. We consider computability w.r.t. ν_i, δ_i, ψ_i^-, $\overline{\nu}$, $\overline{\delta}$ and $\overline{\psi}^-$. Suppose $(x_1, x_2) \in (W_1, W_2) \in \beta_1 \times \beta_2$. From $((x_1, x_2), (W_1, W_2))$ we can compute x_1, x_2, W_1 and W_2. Using t_3' for $\mathbf{X_1}$ and $\mathbf{X_2}$ we can compute (U_i, B_i) such hat $U_i \in \beta_i$ $B_i \subseteq X_i$ is closed and $x_i \in U_i \subseteq B_i \subseteq W_i$ $(i = 1, 2)$. Observe that $(x_1, x_2) \in U_1 \times U_2 \subseteq B_1 \times B_2 \subseteq W_1 \times W_2$. Form (U_1, B_1) and (U_2, B_2) we can compute $((u_1, u_2), (B_1, B_2))$.

Suppose, $\mathbf{X}_1$ and $\mathbf{X}_2$ are SCT_3. For $\mathbf{X_i}$ $(i = 1, 2)$ let R_i be the r.e. set and let r_i be the computable function for SCT_3 from Definition 3. There is a computable function h such that $\psi_1^-(p_1) \times \psi_2^-(p_2) = \overline{\psi}^- \langle p_1, p_2 \rangle$. Let

$$\overline{R} := \{(\langle u_1, u_1 \rangle, \langle w_1, w_2 \rangle) \mid (u_1, w_1) \in R_1 \wedge (u_2, w_2) \in R_2\},$$
$$\overline{r}(\langle u_1, u_1 \rangle, \langle w_1, w_2 \rangle) := h(r_1(u_1, w_1), r_2(u_2, w_2)).$$

A straightforward calculation shows that $\overline{R}$ is the r.e. set and $\overline{r}$ be the computable function for SCT_3 from Definition 3 for the product $\mathbf{X}_1 \times \mathbf{X}_2$.

2. In Example 9, the spaces $\mathbf{X}$ and $\mathbf{R}$ are CT_2, CT_0 and WCT_0. Their product $\mathbf{X} \times \mathbf{R}$, however, is not WCT_0, hence not CT_0 and not CT_2. □

References

[1] Ryszard Engelking. *General Topology*, volume 6 of *Sigma series in pure mathematics*. Heldermann, Berlin, 1989.

[2] Tanja Grubba, Matthias Schröder, and Klaus Weihrauch. Computable metrization. *Mathematical Logic Quarterly*, 53(4–5):381–395, 2007.

[3] Margarita Korovina and Oleg Kudinov. Towards computability over effectively enumerable topological spaces. In Vasco Brattka, Ruth Dillhage, Tanja Grubba, and Angela Klutsch, editors, *CCA 2008, Fifth International Conference on Computability and Complexity in Analysis*, volume 221 of *Electronic Notes in Theoretical Computer Science*, pages 115–125. Elsevier, 2008. CCA 2008, Fifth International Conference, Hagen, Germany, August 21–24, 2008.

[4] Christoph Kreitz and Klaus Weihrauch. Theory of representations. *Theoretical Computer Science*, 38:35–53, 1985.

[5] Victor Selivanov. On the Wadge reducibility of k-partitions. In Ruth Dillhage, Tanja Grubba, Andrea Sorbi, Klaus Weihrauch, and Ning Zhong, editors, *Proceedings of the Fourth International Conference on Computability and Complexity in Analysis (CCA 2007)*, volume 202 of *Electronic Notes in Theoretical Computer Science*, pages 59–71. Elsevier, 2008. CCA 2007, Siena, Italy, June 16–18, 2007.

[6] Klaus Weihrauch. *Computable Analysis*. Springer, Berlin, 2000.

[7] Klaus Weihrauch and Tanja Grubba. Elementary computable topology. *Journal of Universal Computer Science*, 15(6).

Real Computation with Least Discrete Advice: A Complexity Theory of Nonuniform Computability

Martin Ziegler*

Technische Universität Darmstadt

Abstract. It is folklore particularly in numerical and computer sciences that, instead of solving some general problem $f : A \to B$, additional structural information about the input $x \in A$ (that is any kind of promise that x belongs to a certain subset $A' \subseteq A$) should be taken advantage of. Some examples from real number computation show that such discrete advice can even make the difference between computability and uncomputability. We turn this into a both topological and combinatorial complexity theory of information, investigating for several practical problems how much advice is necessary and sufficient to render them computable.
Specifically, finding a nontrivial solution to a homogeneous linear equation $A \cdot \boldsymbol{x} = 0$ for a given singular real $n \times n$-matrix A is possible when knowing $\mathrm{rank}(A) \in \{0, 1, \ldots, n-1\}$; and we show this to be best possible. Similarly, diagonalizing (i.e. finding a basis of eigenvectors of) a given real symmetric $n \times n$-matrix A is possible when knowing the number of distinct eigenvalues: an integer between 1 and n (the latter corresponding to the nondegenerate case). And again we show that n–fold (i.e. roughly $\log n$ bits of) additional information is indeed necessary in order to render this problem (continuous and) computable; whereas finding *some single* eigenvector of A requires and suffices with $\Theta(\log n)$–fold advice.

1 Introduction

Recursive Analysis, that is Turing's [Turi36] theory of rational approximations with prescribable error bounds, is generally considered a very realistic model of real number computation [BrCo06]. Much research has been spent in 'effectivizing' classical mathematical theorems, that is replacing mere existence claims

i) "*for all* x*, there exists some* y *such that* ..." with
ii) "*for all* computable x*, there exists some* computable y *such that* ..."

Cf. e.g. the Intermediate Value Theorem in classical analysis [Weih00, Theorem 6.3.8.1] or the Krein-Milman Theorem from convex geometry [GeNe94].

* Supported by DFG grant Zi 1009/2-1. The author wishes to thank a very carful reviewer for helpful comments. For the full version of this work see arxiv.org/abs/0811.3782

Andrej Bauer, Peter Hertling, Ker-I Ko (Eds.)
6th Int'l Conf. on Computability and Complexity in Analysis, 2009, pp. 269-280
http://drops.dagstuhl.de/opus/volltexte/2009/2277

Note that Claim ii) is non-uniform: it asserts y to *be* computable whenever x is; yet, there may be no way of *converting* a Turing machine M computing x into a machine N computing y [Weih00, Section 9.6]. In fact, computing a function $f : x \mapsto y$ is significantly limited by the sometimes so-called Main Theorem, requiring that any such f be necessarily continuous: because finite approximations to the argument x do not allow to determine the value $f(x)$ up to absolute error smaller than the 'gap' $\limsup_{t\to x} f(t) - \liminf_{t\to x} f(t)$ in case x is a point of discontinuity of f. In particular any non-constant discrete-valued function on the reals is uncomputable—for information-theoretic (as opposed to recursion-theoretic) reasons. Thus, Recursive Analysis is sometimes criticized as a purely mathematical theory, rendering uncomputable even functions as simple as Gauß' staircase [Koep01].

1.1 Motivating Examples

On the other hand many applications do provide, in addition to approximations to the continuous argument x, also certain promise or discrete 'advice'; e.g. whether x is integral or not. And such additional information does render many otherwise uncomputable problems computable:

Example 1. *The Gauß staircase is discontinuous, hence uncomputable. Restricted to integers, however, it is simply the identity, thus computable. And restricted to* non-*integers, it is computable as well; cf.* [Weih00, Exercise 4.3.2]. *Thus, one bit of additional advice (*"integer or not"*) suffices to make* $\lfloor \cdot \rfloor : \mathbb{R} \to \mathbb{Z}$ *computable.*

Also many problems in analysis involving compact (hence bounded) sets are discontinuous unless provided with some integer bound; compare e.g. [Weih00, Section 5.2]. For a more involved illustration from computational linear algebra, we report from [ZiBr04, Section 3.5] the following

Example 2. *Given a real symmetric* $d \times d$ *matrix* A *(in form of approximations* $A_n \in \mathbb{Q}^{d\times d}$ *with* $|A - A_n| \leq 2^{-n}$*), it is generally impossible, for lack of continuity and even in the multivalued sense, to compute (approximations to) any eigenvector of* A*.*
However when providing, in addition to A *itself, the number of distinct eigenvalues (i.e.* not *counting multiplicities) of* A*, finding the entire spectral resolution (i.e. an orthogonal basis of eigenvectors) becomes computable.*

1.2 Complexity Measure of Non-Uniform Computability

We are primarily interested in problems over real Euclidean spaces $\mathbb{R}^d$, $d \in \mathbb{N}$. Yet for reasons of general applicability to arbitrary spaces U of continuum cardinality, we borrow from Weihrauch's TTE framework [Weih00, Section 3] the concept of so-called *representations*, that is, encodings of all elements $u \in U$ as infinite binary strings; and a *realizer* of a function $f : U \to V$ maps encodings of $u \in U$ to encodings of $f(u) \in V$. A *notation* is basically a representation of a merely countable set.

Definition 3. *a) A function $f :\subseteq A \to B$ between topological spaces A and B is* k-wise continuous *if there exists a covering (equivalently: a partition) Δ of $\mathrm{dom}(f) = \bigcup_{D\in\Delta} D$ with $\mathrm{Card}(\Delta) = k$ such that $f|_D$ is continuous for each $D \in \Delta$. Call $\mathfrak{C}_t(f) := \inf\{k : f \text{ is } k\text{-wise continuous}\}$ the* cardinal of discontinuity of f.

b) A function $f :\subseteq A \to B$ between represented spaces (A,α) and (B,β) is (α,β)–computable with k-wise advice *if there exists an at most countable partition Δ of $\mathrm{Card}(\Delta) = k$ and a notation δ of Δ such that the mapping $f_\Delta : (a, D) \mapsto f(a)$ is (α,δ,β)–computable on $\mathrm{dom}(f_\Delta) := \{(a, D) : a \in D \in \Delta\}$. Call $\mathfrak{C}_c(f) = \mathfrak{C}_c(f,\alpha,\beta) := \min\{k : f \text{ is } (\alpha,\beta)\text{–computable with } k\text{-wise advise}\}$ the* complexity of non-uniform (α,β)–computability of f.

c) A function $f :\subseteq A \to B$ is nonuniformly (α,β)*–computable if, for every α–computable $a \in \mathrm{dom}(f)$, $f(a)$ is β-computable.*

So continuous functions are exactly the 1-wise continuous ones; and computability is equivalent to (weak or strong) computability with 1-wise advice. Also we have, as an extension of the Main Theorem of Recursive Analysis, the following immediate

Observation 4. *If α,β are* admissible *representations in the sense of* [Weih00, Definition 3.2.7]*, then every k-wise (α,β)–computable function is k-wise continuous (but not vice versa); that is $\mathfrak{C}_t(f) \le \mathfrak{C}_c(f)$ holds.*
More precisely, every k-wise (α,β)–computable possibly multivalued function $f :\subseteq A \rightrightarrows B$ has a k-wise continuous (α,β)–realizer in the sense of [Weih00, Definition 3.1.3.4].

The above examples illustrate some interesting discontinuous functions to be computable with k-wise advice for some $k \in \mathbb{N}$. Specifically Example 2, diagonalization of real symmetric $n \times n$–matrices is n–wise computable; and Theorem 20 below will show this value n to be optimal.

Remark 5. *We advertise Computability with Finite Advice as a generalization of classical Recursive Analysis:*
a) It constitutes a hybrid approach to both discrete and continuous computation.
b) It complements Type-2 oracle computation: In the discrete realm, every *function $f : \mathbb{N} \to \mathbb{N}$ becomes computable when employing an appropriate oracle; whereas in the Type-2 case, exactly the* continuous *functions $f : \mathbb{R} \to \mathbb{R}$ are computable relative to some oracle. On the other hand, 2-wise advice can make a continuous function computable which without advice has unbounded degree of uncomputability; see Proposition 6d).*
c) Discrete advice avoids a common major point of criticism against Recursive Analysis, namely that it denounces even simplest discontinuous functions as uncomputable;
d) and such kind of advice is very practical: In applications additional discrete information about the input is often actually available and should be used. For instance a given real matrix may be known to be non-degenerate (as is often exploited in numerics) or, slightly more generally, to have k eigenvalues coincide

for some known $k \in \mathbb{N}$.
e) The topology of the members of the collection Δ from Definition 3 can usually be chosen not too wild: compare the examples considered below. In practice, we consider the discrete advice to arise with the input itself. For instance the band-width of a given matrix A may be known as 3 because A comes from a finite element triangular grid approach. Hence the collection Δ need not even be explicit (since it is usually far from unique, anyway; compare Remark 11), nor required effective in any sense.

1.3 Related Work, in particular Kolmogorov Complexity

Several approaches have been pursued in literature to make also discontinuous functions accessible for computability investigations.

Exact Geometric Computation considers the arguments $\boldsymbol{x}$ as exact rational numbers [LPY05].

Special encodings of discontinuous functions motivated by spaces in Functional Analysis, are treated e.g. in [ZhWe03]; however these do not admit evaluation.

Weakened notions of computability may refer to stronger models of computation [ChHo99]; provide more information on (e.g. the binary encoding of, rather than rational approximations with error bounds to) the argument x [Mori02,MTY05]; or expect less information on (e.g. no error bounds for approximations to) the value $f(x)$ [WeZh00].

A taxonomy of discontinuous functions, namely their *degrees* of Borel measurability, is investigated in [Brat05,Zie07a,Zie07b]:
Specifically, a function $f :\subseteq A \to B$ is continuous ($=\Sigma_1$–measurable) iff, for every closed $T \subseteq B$, its preimage $f^{-1}[T]$ is closed in $\mathrm{dom}(f) \subseteq A$; and f is computable iff this mapping $T \mapsto f^{-1}[T]$ on closed sets is $(\psi^d_>, \psi^d_>)$–computable. A degree relaxation, f is called Σ_2–measurable iff, for every closed $T \subseteq B$, $f^{-1}[T]$ is an F_δ-set.

Wadge degrees of discontinuity are an (immense) refinement of the above, namely with respect to so-called *Wadge reducibility*; cf. e.g. [Weih00, Section 8.2].

Levels of discontinuity are studied in [HeWe94,Her96a,Her96b]:
Take the set $X_0 \subseteq \mathrm{dom}(f)$ of points of discontinuity of f; then the set $X_1 \subseteq X_0$ of points of discontinuity of $f|_{X_0}$ and so on: the least index k for which X_k is empty is f's level of discontinuity.

Our approach superficially resembles the third and last ones above. A minor difference, they correspond to *ordinal* measures whereas the size of the partition considered in Definition 3 is a *cardinal*. As a major difference we now establish these measures as logically largely independent:

Proposition 6. *a) There exists a 2-wise computable function $f : [0,1] \to \{0,1\}$ which is not measurable nor on any level of discontinuity.*
b) There exists a Δ_2–measurable function $f : [0,1] \to [0,1]$ with is not k-wise continuous for any finite k.

c) If f is on the k-th level of discontinuity, it is $(k+1)$-wise continuous.
d) There exists a continuous, 2-wise computable function $f :\subseteq [0,1] \to [0,1]$ which is not computable, even relative to any prescribed oracle.
e) Every k-wise computable function is nonuniformly computable; whereas there are nonuniformly computable functions not k-wise computable for any $k \in \mathbb{N}$.

Conditions where nonuniform computability does imply (even) 1-wise computability have been devised in [Brat99]. Further related research includes

Computational Complexity of real functions; see e.g. [Ko91] and [Weih00, Section 7]. Note, however, that Definition 3 refers to a purely *information*-theoretic notion of complexity of a function and is therefore more in the spirit of

Information-based Complexity in the sense of [TWW88]. There, on the other hand, inputs are considered as real number entities given exactly; whereas we consider approximations to real inputs enhanced with discrete advice.

Finite Continuity is being studied for *Darboux Functions* in [MaPa02,Marc07]. It amounts to d-wise continuity for some $d \in \mathbb{N}$ according to Definition 3a).

Kolmogorov Complexity has been investigated for finite strings and, asymptotically, for infinite ones; cf. e.g. [LiVi97, Section 2.5] and [Stai99]. Also a kind of advice is part of that theory in form of *conditional* complexity [LiVi97, Definition 2.1.2].

We quote from [LiVi97, Exercise 2.3.4abe] the following

Fact 7. *An infinite string $\bar{\sigma} = (\sigma_n)_{n\in\omega} \in \Sigma^\omega$ is computable (e.g. printed onto a one-way output tape by some so-called Type-2 or monotone machine; cf.* [Weih00,Schm02]*)*

a) iff its initial segments $\bar{\sigma}_{1:n} := (\sigma_1, \ldots \sigma_n)$ have Kolmogorov complexity $\mathcal{O}(1)$ conditionally to n, *i.e., iff $C(\bar{\sigma}_{1:n}|n)$ is bounded by some $c = c(\bar{\sigma}) \in \mathbb{N}$ independent of n.*
b) Equivalently: the uniform *complexity $C_{\mathrm{u}}(\bar{\sigma}_{1:n}) := C(\bar{\sigma}_{1:n}; n)$ in the sense of* [LiVi97, Exercise 2.3.3] *(that is the complexity of the function $\{1, \ldots, n\} \ni i \mapsto \sigma_i$ from* [LiVi97, Exercise 2.1.12] *but additionally relativized to the size n of the domain) is bounded by some c for infinitely many n.*

Definition 8. *a) For $\bar{\sigma} \in \Sigma^\omega$, write $C(\bar{\sigma}) := \sup_n C\big(\bar{\sigma}_{1:n}|n\big)$ and $C(\bar{\sigma}|\bar{\tau}) := \sup_n C\big(\bar{\sigma}_{1:n}|n, \bar{\tau}\big)$, where the Kolmogorov complexity conditional to an* in*finite string is defined literally as for a finite one* [LiVi97, Definition 2.1.1].
b) Similarly, let $C_{\mathrm{u}}(\bar{\sigma}|\bar{\tau}) := \sup_n C_{\mathrm{u}}\big(\bar{\sigma}_{1:n}|\bar{\tau}\big)$.
c) For a represented space (A, α) and $a \in A$, write $C(a) := \inf\{C(\bar{\sigma}) : \alpha(\bar{\sigma}) = a\}$ and $C_{\mathrm{u}}(a) := \inf\{C_{\mathrm{u}}(\bar{\sigma}) : \alpha(\bar{\sigma}) = a\}$.

Note that we purposely do not consider some *normalized* form like $\mathrm{C}(\bar{\sigma}_{1:n}|n)\boldsymbol{/n}$ in order to establish the following

Proposition 9. *A function $F :\subseteq \Sigma^\omega \to \Sigma^\omega$ is computable with finite advice iff the Kolmogorov complexity $C_{\mathrm{u}}\big(F(\bar{\sigma})|\bar{\sigma}\big)$ is bounded by some c independent of $\bar{\sigma} \in \mathrm{dom}(F)$.*

2 Complexity of Nonuniform Computability

Lemma 10. *a) Let $f : A \to B$ be d-wise continuous (computable) and $A' \subseteq A$. Then the restriction $f|_{A'}$ is again d-wise continuous (computable).*
b) Let $f : A \to B$ be d-wise continuous (computable) and $g : B \to C$ be k-wise continuous (computable). Then $g \circ f : A \to C$ is $d \cdot k$-wise continuous (computable).
c) If $f : A \to B$ is (α, β)–computable with d-wise advice and $\alpha' \preceq \alpha$ and $\beta \preceq \beta'$, then f is also (α', β')–computable with d-wise advice.

A minimum size partition Δ of $\mathrm{dom}(f)$ to make f computable on each $D \in \Delta$ need not be unique: Alternative to Example 1, we

Remark 11. *Given a ρ–name of $x \in \mathbb{R}$ and indicating whether $\lfloor x \rfloor \in \mathbb{Z}$ is even or odd suffices to compute $\lfloor x \rfloor$:*
Suppose $\lfloor x \rfloor = 2k \in 2\mathbb{Z}$ (the odd case proceeds analogously). Then $x \in [2k, 2k+1)$. Conversely, $x \in [2k-1, 2k+2)$, together with the promise $\lfloor x \rfloor \in 2\mathbb{Z}$, implies $\lfloor x \rfloor = 2k$. Hence, given $(q_n) \in \mathbb{Q}$ with $|x - q_n| \leq 2^{-n}$, $k := 2 \cdot \lfloor q_1/2 + \frac{1}{4} \rfloor$ (calculated in exact rational arithmetic) will yield the answer. □

2.1 Witness of k-wise Discontinuity

Recall that the partition Δ in Definition 3 need not satisfy any (e.g. topological regularity) conditions. The following notion turns out as useful in lower bounding the cardinality of such a partition:

Definition 12. *a) A d-dimensional flag $\mathcal{F}$ in a topological Hausdorff space X is a collection*

$$x, \quad (x_n)_n \ , \quad (x_{n,m})_{n,m} \ , \quad (x_{n,m,\ell})_{n,m,\ell} \ , \quad \ldots, \quad (x_{n_1,\ldots,n_d})_{n_1,\ldots,n_d}$$

of a point and of (multi-)sequences[†] in X such that, for each (possibly empty) multi-index $\bar{n} \in \mathbb{N}^k$ $(0 \leq k < d)$, it holds $x_{\bar{n}} = \lim_{m\to\infty} x_{\bar{n},m}$.
b) $\mathcal{F}$ is uniform if furthermore, again for each $\bar{n} \in \mathbb{N}^k$ $(0 \leq k < d)$ and for each $1 \leq \ell \leq d-k$, it holds $x_{\bar{n}} = \lim_{m\to\infty} x_{\bar{n},\underbrace{m,\ldots,m}_{\ell \text{ times}}}$.
c) For $f :\subseteq X \to Y$ and $x \in \mathrm{dom}(f)$ a witness of discontinuity of f at x is a sequence $x_n \in \mathrm{dom}(f)$ such that $\lim_{n\to\infty} f(x_n)$ exists but differs from $f(x)$.
d) For $f :\subseteq X \to Y$, a witness of d-wise discontinuity of f is a uniform d-dimensional flag $\mathcal{F}$ in $\mathrm{dom}(f)$ such that, for each $k = 0, 1, \ldots, d-1$ and for each $\bar{n} \in \mathbb{N}^k$ and for each $1 \leq \ell \leq d-k$, $\left(x_{\bar{n},\underbrace{m,\ldots,m}_{\ell \text{ times}}}\right)_m$ is a witness of discontinuity of f at $x_{\bar{n}}$.

[†] The generally more appropriate concept is that of a *Moore-Smith* sequence or *net*. However, being interested in second countable spaces, we may and shall restrict to ordinary sequences. Similarly, the Hausdorff condition is invoked for mere convenience.

Observe that, since d is finite, we may always (although not effectively) proceed from a flag to a uniform one by iteratively taking appropriate subsequences. In fact, sub(multi)sequences of d-flags and of witnesses of discontinuity are again d-flags and witnesses of discontinuity.

Lemma 13. *Let X, Y be Hausdorff, $f : X \to Y$ a function, and suppose there exists a witness of d-wise discontinuity of f. Then $\mathfrak{C}_{\mathrm{t}}(f) > d$.*

2.2 First Example: Matrix Rank

Observe that for an $N \times M$-matrix A and $d := \min(N, M)$, $\mathrm{rank}(A)$ is an integer between 0 and d; and knowing this number makes rank trivially computable. Conversely, such $(d+1)$–fold information is necessary by Lemma 13 and

Example 14. *Consider the space $\mathbb{R}^{N\times M}$ of rectangular matrices and let $d := \min(N, M)$. For $i \in \{0, 1, \ldots, d\}$ write*

$$E_i \quad := \quad \sum_{j=1}^{i} \left((0, \cdots, 0, \underbrace{1}_{j\text{-}th}, 0, \cdots, \underbrace{0}_{n\text{-}th})^{\dagger} \otimes (0, \cdots, 0, \underbrace{1}_{j\text{-}th}, 0, \cdots, \underbrace{0}_{m\text{-}th}) \right) .$$

$$X := 0, \qquad X_{n_1,\ldots,n_i} \quad := \quad E_1/n_1 \; + \; E_2/n_2 \; + \; \cdots \; + \; E_i/n_i$$

has $\lim_{m\to\infty} X_{n_1,\ldots,n_i,m,\ldots,m} = X_{n_1,\ldots,n_i}$, hence constitutes a uniform d-dimensional flag. Moreover, $\mathrm{rank}(E_i) = i = \mathrm{rank}(X_{n_1,\ldots,n_i}) \neq i + \ell = \mathrm{rank}(X_{n_1,\ldots,n_i,\underbrace{m,\ldots,m}_{\ell \text{ times}}})$ shows it is a witness of d-wise discontinuity of $\mathrm{rank} : \mathbb{R}^{N\times M} \to \{0, 1, \ldots, d\}$. □

3 Multivalued Functions, i.e. Relations

Many applications involve functions which are 'non-deterministic' in the sense that, for a given input argument x, several values y are acceptable as output; recall e.g. Items i) and ii) in Section 1. Also in linear algebra, given a singular matrix A, we want to find *some* (say normed) vector $\boldsymbol{v}$ such that $A \cdot \boldsymbol{v} = 0$. This is reflected by relaxing the mapping $f : x \to y$ to be not a function but a relation (also called multivalued function); writing $f : X \rightrightarrows Y$ instead of $f : X \to 2^Y \setminus \{\emptyset\}$ to indicate that for an input $x \in X$, *any* output $y \in f(x)$ is acceptable. Many practical problems have been shown computable as multivalued functions but admit no computable single-valued so-called *selection*; cf. e.g. [Weih00, Exercise 5.1.13], [ZiBr04, Lemma 12 or Proposition 17]. On the other hand, even relations often lack computability merely for reasons of continuity—and appropriate additional discrete advice renders them computable, recall Example 2.

3.1 Dis-/Continuity for Multivalued Mappings

Like single-valued computable functions (recall the Main Theorem), also computable relations satisfy certain topological conditions. However for such multivalued mappings, literature knows a variety of easily confusable notions [ScNe07].

Hemicontinuity for instance is not necessary for real computability. It may be tempting to regard *computing* a multivalued mapping f as the task of calculating, given x, the set-value $f(x)$ [Spre08]. In our example applications, however, one wants to capture that a machine is permitted, given x, to 'nondeterministically' choose and output *some* value $y \in f(x)$. Note that this coincides with [Weih00, Definition 3.1.3]. In particular we do not insist that, upon input x, *all* $y \in f(x)$ occur as output for *some* nondeterministic choice—as required in [Brat03, Section 7]. Instead, let us generalize Definition 12 as follows:

Definition 15. *Fix some possibly multivalued mapping $f :\subseteq X \rightrightarrows Y$ and write $\mathrm{dom}(f) := \{x \in X : f(x) \neq \emptyset\}$. Call f continuous at $x \in X$ if there is some $y \in f(x)$ such that for every open neighbourhood V of y there exists a neighbourhood U of x such that $f(z) \cap V \neq \emptyset$ for all $z \in U$.*

For ordinary (i.e. single-valued) functions f, $\mathrm{dom}(f)$ amounts to the usual notion; and such f is obviously continuous (at x) iff it is continuous (at x) in the original sense. Indeed, Lemma 18a) below is an immediate extension of the Main Theorem of Recursive Analysis, showing that any computable *multi*valued mapping is necessarily continuous.

Lemma 10a) literally applies also to multivalued mappings $f : A \rightrightarrows B$. We failed to similarly fully generalize Lemma 10b); but already the following partial generalization turns out as useful:

Lemma 16. *a) Let $f : A \to B$ be single-valued and $g : B \rightrightarrows C$ multivalued. If f is d-wise continuous (computable) and g is k-wise continuous (computable), then $g \circ f : A \rightrightarrows C$ is $d \cdot k$-wise continuous (computable).*

b) Let $f : A \rightrightarrows B$ and $g : B \rightrightarrows C$ be multivalued. If f is d-wise continuous (computable) and g is continuous (computable), then $g \circ f : A \rightrightarrows C$ is again d-wise continuous (computable).

Definition 17. *a) For $x \in \mathrm{dom}(f)$, a witness of discontinuity of f at x is a sequence $(x_n) \in \mathrm{dom}(f)$ converging to x such that, for every $y \in f(x)$ there is some open neighbourhood V of y disjoint from $f(x_n)$ for infinitely many $n \in \mathbb{N}$.*

b) A uniform d-dimensional flag $\mathcal{F}$ in X is a witness of d-wise discontinuity of f if, for each $0 \leq k < d$ and for each $\bar{n} \in \mathbb{N}^k$ and for each $1 \leq \ell \leq d-k$ and for each $y \in f(x_{\bar{n}})$, $\big(x_{\bar{n},\underbrace{m,\ldots,m}_{\ell \text{ times}}}\big)_m$ is a witness of discontinuity of f at $x_{\bar{n}}$.

If multivalued f admits a witness of discontinuity at x, then f is not continuous. Conversely, if X is first-countable, discontinuity of f at x yields the existence of a witness of discontinuity at x. Also, witnesses of 1-wise discontinuity coincide with witnesses of discontinuity; and they generalize the definition from the single-valued case. Lemma 18 below extends Lemma 13 in showing that a witness of d-wise discontinuity of f inhibits d-wise computability.

Lemma 18. *Let (A,α) and (B,β) be effective metric spaces*[‡] *with corresponding Cauchy representations and $f :\subseteq A \rightrightarrows B$ a possibly multivalued mapping.*

a) If f admits a witness of discontinuity, then it is not (α,β)–continuous.
b) If f admits a witness of d-wise discontinuity, f is not d-wise (α,β)–continuous.

4 Applications to Effective Linear Algebra

Based on Lemma 13b), we now determine the complexity of non-uniform computability for several concrete standard problems in linear algebra and in particular of Example 2. But first consider the problem of solving a system of linear equations; more precisely of finding a nonzero vector in the kernel of a given singular matrix. It is for mere notational convenience that we formulate for the case of real matrices: complex ones work just as well.

Theorem 19. *Fix $n,m \in \mathbb{N}$, $d := \min(n, m-1)$, and consider the space $\mathbb{R}^{n\times m}$ of $n \times m$ matrices, considered as linear mappings from $\mathbb{R}^m$ to $\mathbb{R}^n$. Then the multivalued mapping*

$$\mathrm{LinEq} : A \;\mapsto\; \mathrm{kernel}(A)\backslash\{0\}, \qquad \mathrm{dom}(\mathrm{LinEq}) \;:=\; \{A \in \mathbb{R}^{n\times m} : \mathrm{rank}(A) \le d\}$$

is well-defined and has complexity $\mathfrak{C}_{\mathrm{t}}(\mathrm{LinEq}) = \mathfrak{C}_{\mathrm{c}}(\mathrm{LinEq}, \rho^{n\times m}, \rho^m) = d+1$.

Concerning diagonalization of symmetric real matrices, we can prove

Theorem 20. *Fix $d \in \mathbb{N}$ and consider the space $\mathbb{R}^{\binom{d}{2}}$ of real symmetric $d \times d$ matrices. Then the multivalued mapping*

$$\mathrm{Diag} : \mathbb{R}^{\binom{d}{2}} \quad \ni \quad A \quad \mapsto \quad \big\{(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_d) \text{ basis of } \mathbb{R}^d \text{ of eigenvectors to } A\big\}$$

has complexity $\mathfrak{C}_{\mathrm{t}}(\mathrm{Diag}) = \mathfrak{C}_{\mathrm{c}}\big(\mathrm{Diag}, \rho^{\binom{d}{2}}, \rho^{d\times d}\big) = d$.

The lack of continuity of the mapping Diag is closely related to inputs with degenerate eigenvalues [ZiBr04, Example 18]. In fact our below proof yields a witness of d-wise discontinuity by constructing an iterated sequence of *symmetry breakings* in the sense of Mathematical Physics. On the other hand even in the non-degenerate case, Diag is inherently multivalued since any permutation of a basis constitutes again a basis.

4.1 Finding Some Eigenvector

Instead of computing an entire basis of eigenvectors, we now turn to the problem of determining just one arbitrary eigenvector to a given real symmetric matrix. This turns out to be considerably less 'complex':

[‡] Cf. [Weih00, Section 8.1] for a formal definition and imagine Euclidean spaces $\mathbb{R}^k$ as major examples and focus of interest for our purpose.

Theorem 21. *For a real symmetric $n \times n$-matrix A, consider the quantity*

$$m \quad := \quad \min\big\{ \dim \mathrm{kernel}(A - \lambda\,\mathrm{id}) : \lambda \in \sigma(A) \big\} \quad \in \quad \{1, \ldots, n\} \ .$$

Given $d := \lfloor \log_2 m \rfloor \in \{0, 1, \ldots, \lfloor \log_2 n \rfloor\}$ and a $\rho^{\binom{n}{2}}$–name of A, one can ρ^n–compute (i.e. effectively find) some eigenvector of A.

The proof employs the following tool about computability of finite multi-sets.

Lemma 22. *Let $(x_1, \ldots, x_n)$ denote an n-tuple of real numbers and consider the induced partition $\mathcal{I} := \big\{ \{1 \le i \le n : x_i = x_j\} : 1 \le j \le n \big\}$ of the index set $\{1, \ldots, n\} =: [n]$ according to the equivalence relation $i \equiv j :\Leftrightarrow x_i = x_j$. Furthermore let $m := \min\big\{ \mathrm{Card}(I) : I \in \mathcal{I} \big\}$.*

a) Consider $I \subseteq [n]$ with $1 \le \mathrm{Card}(I) < 2m$. Then the following implies $I \in \mathcal{I}$:

$$x_i \neq x_j \quad \text{for all} \quad i \in I \quad \text{and all} \quad j \in [n] \setminus I \ . \tag{1}$$

b) Suppose $k \in \mathbb{N}$ is such that $k \le m < 2k$. Then there exists $I \in \mathcal{I}$ with $k \le \mathrm{Card}(I) < 2k$ satisfying (1). Conversely every $I \subseteq [n]$ with $k \le \mathrm{Card}(I) < 2k$ satisfying (1) has $I \in \mathcal{I}$.

c) Given a ρ^n–name of $(x_1, \ldots, x_n)$ and given $k \in \mathbb{N}$ with $k \le m < 2k$, one can computably find some $I \in \mathcal{I}$.

d) Given a ρ^n–name of $(x_1, \ldots, x_n)$ and given $\mathrm{Card}(\mathcal{I})$, one can compute $\mathcal{I}$.

Claim c) can be considered a weakening of Claim d) which had been established in [ZiBr04, Proposition 20].

Proof (Theorem 21). Compute according to [ZiBr04, Proposition 17] some (ρ^n–name of an) n-tuple of eigenvalues $(\lambda_1, \ldots, \lambda_n)$ of A, repeated according to their multiplicities. Now due to [ZiBr04, Theorem 11], (some eigenvector in) the eigenspace $\mathrm{kernel}(A - \lambda_i\,\mathrm{id})$ can be computably found when knowing $\mathrm{rank}(A - \lambda_i\,\mathrm{id})$ (recall Theorem 19), that is the multiplicity of λ_i in the multi-set $(\lambda_1, \ldots, \lambda_n)$. To this end we apply Lemma 22c), observing $k := 2^d \le m < 2k$ since $d = \lfloor \log_2 m \rfloor$. □

Theorem 23. *The multivalued mapping*

$$\mathrm{EVec}_n : \mathbb{R}^{\binom{n}{2}} \quad \ni \quad A \quad \mapsto \quad \{\boldsymbol{w} \textit{ eigenvector of } A\}$$

has complexity $\mathfrak{C}_t(\mathrm{EVec}_n) = \mathfrak{C}_c\big(\mathrm{EVec}_n, \rho^{\binom{n}{2}}, \rho^n\big) = \lfloor \log_2 n \rfloor + 1$.

5 Conclusion and Perspectives

We claim that a major source of criticism against Recursive Analysis misses the point: Although computable functions f *are* necessarily continuous when given approximations to the argument x *only*, most practical f's do become computable when providing in addition some discrete information about x. Such

'advice' usually consists of some very natural and mathematically explicit integer value from a bounded range (e.g. the rank of the matrix under consideration) and is readily available in practical applications.

We have then turned this observation into a complexity theory, investigating the minimum *size* (=cardinal) of the range this discrete information comes from. And we have devised mathematical tools and used them to determine this quantity for several simple and natural problems from linear algebra: calculating the rank of a given matrix, solving a system of linear equalities, diagonalizing a symmetric matrix, and finding some eigenvector to a given symmetric matrix. The latter three are inherently multivalued. And they exhibit a considerable difference in complexity: for input matrices of format $n \times n$, usually discrete advice of order $\Theta(n)$ is necessary and sufficient; whereas some single eigenvector can be found using only $\Theta(\log n)$–fold advice: namely the quantity $\lfloor \log_2 \min \{ \dim \mathrm{kernel}(A - \lambda\,\mathrm{id}) : \lambda \in \sigma(A) \} \rfloor$. The algorithm exploits this data based on some combinatorial considerations—which nicely complement the heavily analytical and topological arguments usually dominant in proofs in Recursive Analysis.

References

[BKOS97] M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf: "*Computational Geometry, Algorithms and Applications*", Springer (1997).

[BrCo06] M. Braverman, S. Cook: "Computing over the Reals: Foundations for Scientific Computing", pp.318–329 in *Notices of the AMS* vol.**53:3** (2006).

[Brat99] V. Brattka: "Computable Invariance", pp.3–20 in *Theoretical Computer Science* vol.**210:1** (1999).

[Brat03] V. Brattka: "Computability over Topological Structures", pp.93–136 in *Computability and Models* (S.B.Cooper and S.S. Goncharov, Edts), Springer (2003).

[Brat05] V. Brattka: "Effective Borel measurability and reducibility of functions", pp.19–44 in *Mathematical Logic Quarterly* vol.**51** (2005).

[ChHo99] T. Chadzelek, G. Hotz: "Analytic Machines", pp.151–165 in *Theoretical Computer Science* vol.**219**, Elsevier (1999).

[GeNe94] X. Ge, A. Nerode: "On Extreme Points of Convex Compact Turing Located Sets", pp.114–128 in *Logical Foundations of Computer Science*, Springer LNCS vol.**813** (1994).

[HeWe94] P. Hertling, K. Weihrauch: "Levels of Degeneracy and Exact Lower Complexity Bounds for Geometric Algorithms", pp.237–242 in *Proc. 6th Canadian Conference on Computational Geometry* (CCCG 1994).

[Her96a] P. Hertling: "Unstetigkeitsgrade von Funktionen in der effektiven Analysis", Dissertation Fern-Universität Hagen, *Informatik-Berichte* **208** (1996).

[Her96b] P. Hertling: "Topological Complexity with Continuous Operations", pp.315–338 in *Journal of Complexity* vol.**12** (1996).

[Ko91] K. Ko: "*Complexity Theory of Real Functions*", Birkhäuser (1991).

[Koep01] W. Koepf: "Besprechungen zu Büchern der Computeralgebra: Klaus Weihrauch Computable Analysis", p.29 in *Computeralgebra Rundbrief* vol.**29** (2001), http://www.fachgruppe-computeralgebra.de/CAR/CAR29/node19.html

[LiVi97] M. Li, P. Vitanyi: "*In Introduction to Kolmogorov Complexity and its Applications*" (Second Edition), Springer (1997).

[LPY05] C. Li, S. Pion, C.K. Yap: "Recent Progress in Exact Geometric Computation", pp.85–111 in *Journal of Logic and Algebraic Programming* vol.**64** (2005).

[MaPa02] M. Marciniak, R.J. Pawlak: "On the Restrictions of Functions. Finitely Continuous Functions and Path Continuity", pp.65–77 in *Tatra Mountains Mathematical Publications* vol.**24** (2002).

[Marc07] M. Marciniak: "On Finitely Continuous Darboux Functions and Strong Finitely Continuous Functions", pp.15–22 in *Real Analysis Exchange* vol.**33:1** (2007).

[Mori02] T. Mori: "On the Computability of Walsh Functions", pp.419–436 in *Theoretical Computer Science* vol.**284** (2002).

[MTY05] T. Mori, Y. Tsujii, M. Yasugi: "Fine-Computable Functions and Effective Fine Convergence", pp.177–197 in *Proc. 2nd Int. Conf. Computability and Complexity in Analysis* (CCA'05);
revised version accepted by *Mathematics Applied in Science and Technology.*

[Schm02] J. Schmidhuber: "Hierarchies of Generalized Kolmogorov Complexity and Nonenumerable Measures Computable in the Limit", pp.587–612 in *Int. J. of Foundations of Computer Science* vol.**13:4**, World Scientific Publishing (2002)

[ScNe07] P. Schodl, A. Neumaier: "Continuity Notions for Multi-Valued Mappings", pre-print http://www.mat.univie.ac.at/ neum/ms/continuity.pdf, University of Vienna (2007).

[Spre08] D. Spreen: "On the Continuity of Effective Multifunctions", pp.267–282 in *Prof. Fifth Int. Conf. on Computability and Complexity in Analysis* (CCA 2008), Informatik Berichte FernUniversität in Hagen vol.**348** (August 2008).

[Stai99] L. Staiger: "The Kolmogorov Complexity of Real Numbers", pp.536–546 in *Proc. Fundamentals of Computation Theory* (FCT'99), Springer LNCS vol.**1684**.

[TWW88] J.F. Traub, G.W. Wasilkowski, H. Woźniakowski: "*Information-Based Complexity*", Academic Press (1988).

[Turi36] A.M. Turing: "On Computable Numbers, with an Application to the Entscheidungsproblem", pp.230–265 in *Proc. London Math. Soc.* vol.**42(2)** (1936).

[Weih00] K. Weihrauch: "*Computable Analysis*", Springer (2000).

[WeZh00] K. Weihrauch, X. Zheng: "Computability on continuous, lower semi-continuous and upper semi-continuous real functions", pp.109–133 in *Theoretical Computer Science* vol.234 (2000).

[ZiBr04] M. Ziegler, V. Brattka: "Computability in Linear Algebra", pp.187–211 in *Theoretical Computer Science* vol.**326** (2004).

[Zie07a] M. Ziegler: "Real Hypercomputation and Continuity", pp.177–206 in *Theory of Computing Systems* vol.**41** (2007).

[Zie07b] M. Ziegler: "Revising Type-2 Computation and Degrees of Discontinuity", pp.255–274 in *Proc. 3rd International Conference on Computability and Complexity in Analysis* (CCA'06), Electronic Notes in Theoretical Computer Science vol.**167** (2007).

[ZhWe03] N. Zhong, K. Weihrauch: "Computability Theory of Generalized Functions", pp.469–505 in *Journal of the ACM* vol.**50** (2003).