RESEARCH CENTRE

Nancy - Grand Est

IN PARTNERSHIP WITH:

CNRS, Université de Lorraine

2020

ACTIVITY REPORT

Project-Team

MOCQUA

Designing the Future of Computational Models

IN COLLABORATION WITH: Laboratoire lorrain de recherche en informatique et ses applications (LORIA)

DOMAIN

Algorithmics, Programming, Software and Architecture

THEME

Proofs and Verification

Contents

Project-Team MOCQUA

Creation of the Project-Team: 2018 January 01

Keywords

Computer sciences and digital sciences

A2.3.2. – Cyber-physical systems

A2.4.1. – Analysis

A6.5. – Mathematical modeling for physical sciences

A7.1.4. – Quantum algorithms

A7.2. – Logic in Computer Science

A7.3. – Calculability and computability

A8.1. – Discrete mathematics, combinatorics

A8.3. – Geometry, Topology

A8.6. – Information theory

Other research topics and application domains

B9.5.1. – Computer science

B9.5.2. – Mathematics

B9.5.3. – Physics

1 Team members, visitors, external collaborators

Research Scientists

- Mathilde Bouvel [CNRS, from Aug 2020]
- Nazim Fatès [Inria, Researcher]
- Isabelle Gnaedig-Antoine [Inria, Researcher, Deputy team leader]
- Mathieu Hoyrup [Inria, Researcher]
- Simon Perdrix [CNRS, Researcher, HDR]
- Vladimir Zamdzhiev [Inria, ISFP, from Oct 2020]

Faculty Members

- Emmanuel Jeandel [Team leader, Univ de Lorraine, Professor, HDR]
- Emmanuel Hainry [Univ de Lorraine, Associate Professor]
- Irene Marcovici [Univ de Lorraine, Associate Professor]
- Romain Péchoux [Univ de Lorraine, Associate Professor, HDR]

Post-Doctoral Fellows

- Marc De Visme [CNRS, from Oct 2020]
- Vladimir Zamdzhiev [CNRS, until Sep 2020]

PhD Students

- Djamel Eddine Amir [Univ de Lorraine, from Sep 2020]
- Robert Booth [CNRS]
- Agustin Borgna [CNRS]
- Titouan Carette [Univ de Lorraine]
- Alexandre Clement [Univ de Lorraine]
- Pierre Mercuriali [Univ de Lorraine, until Sep 2020]
- Margarita Veshchezerova [EDF, CIFRE]

Interns and Apprentices

- Emilien Brun [Inria, until Apr 2020]
- Oceane Chazé [Univ de Lorraine, from Jun 2020 until Jul 2020]
- Yohann D'anello [Univ de Lorraine, from Jun 2020 until Jul 2020]
- Ophelie Phonchareun [Univ de Lorraine, from Jun 2020 until Jul 2020]
- Amaury Saint-Jore [Univ de Lorraine, from Mar 2020 until Aug 2020]

Administrative Assistants

- Sophie Drouot [Inria]
- Sylvie Hilbert [CNRS]

Visiting Scientist

- Mukeru Kajama Safari [University of South Africa, until Apr 2020]

2 Overall objectives

2.1 Designing the future of computational models

The goal of the Mocqua team is to tackle challenges coming from the emergence of new or future computational models. The landscape of computational models has indeed changed drastically in the last few years: the complexity of digital systems is continually growing, which leads to the introduction of new paradigms, while new problems arise due to this larger scale (tolerance to faulty behaviors, asynchronicity) and constraints of the present world (energy limitations). In parallel, new models based on physical considerations have appeared. There is thus a real need to accompany these changes, and we intend to investigate these new models and try to solve their intrinsic problems by computational and algorithmic methods.

While the bit remains undeniably the building block of computer architecture and software, it is fundamental for the development of new paradigms to investigate computations and programs working with inputs that cannot be reduced to finite strings of 0's and 1's. Our team will focus on a few instances of this phenomenon: programs working with qubits (quantum computing), programs working with functions as inputs (higher-order computation) and programs working in infinite precision (real numbers, infinite sequences, streams, coinductive data, ...).

3 Research program

3.1 Quantum Computing

While it can be argued that the quantum revolution has already happened in cryptography [47] or in optics [46], quantum computers are far from becoming a common commodity, with only a few teams around the world working on a practical implementation. In fact, one of the most commonly known examples of a quantum computer, the D-Wave 2X System, defies the usual definition of a computer: it is not general-purpose, and can only solve (approximately) a very specific hardwired problem.

Most current prototypes of a quantum computer differ fundamentally on the hardware substrate, and it is quite hard to predict which solution will finally be adopted. The landscape of quantum programming languages is also constantly evolving. Comparably to compiler design, the foundation of quantum software therefore relies on an intermediate representation that is suitable for manipulation, easy to produce from software and easily encodable into hardware. The language of choice for this is the ZX-calculus.

Regardless of the actual model that will be accepted by the industry, it is becoming clear that some of the hurdles into scaling up quantum computers from a few qubits to very large arrays will remain. As an example, current implementations of quantum computers working on hundreds of qubits indeed are not able to form and maintain all possible forms of entanglement between qubits. This raises two questions. First, does this restrict the computational power, and the supposed advantage of the quantum computer over the classical computer? Second, how to ensure that a quantum program that was designed for a theoretical quantum computer will work on the practical implementations? This will be investigated, in particular by providing static analysis methods for evaluating a priori how much entanglement a quantum program needs.

3.2 Higher-Order Computing

While programs often operate on natural numbers or finite structures such as graphs or finite strings, they can also take functions as input. In that case, the program is said to perform higher-order computations, or to compute a higher-order functional. Functional programming or object-oriented programming are important paradigms allowing higher-order computations.

While the theory of computation is well developed for first-order programs, difficulties arise when dealing with higher-order programs. There are many non-equivalent ways of presenting inputs to such programs: an input function can be presented as a black-box, encoded in an infinite binary sequence, or sometimes by a finite description. Comparing those representations is an important problem. A particularly useful application of higher-order computations is to compute with infinite objects that can be represented by functions or symbolic sequences. The theory works well in many cases (to be precise, when these objects live in a topological space with a countable basis [56]), but is not well understood in other interesting cases. For instance, when the inputs are the second-order functionals (of type $(\mathbb{N} \to \mathbb{N}) \to (\mathbb{N} \to \mathbb{N})$), the classical theory does not apply and many problems are still open.

3.3 Dynamical Systems

The most natural example of a computation with infinite precision is the simulation of a dynamical system. The underlying space might be $\mathbb{R}^n$ in the case of the simulation of physical systems, or the Cantor space $\{0,1\}^{\mathbb{Z}}$ in the case of discrete dynamical systems.

From the point of view of computation, the main point of interest is the link between the long-term behavior of a system and its initial configuration. There are two questions here: (a) predict the behavior, (b) design dynamical systems with some prescribed behavior. The first will be mainly examined through the angle of reachability and more generally control theory for hybrid systems.

The model of cellular automata will be of particular interest. This computational model is relevant for simulating complex global phenomena which emerge from simple interactions between simple components. It is widely used in various natural sciences (physics, biology, etc.) and in computer science, as it is an appropriate model to reason about errors that occur in systems with a great number of components.

The simulation of a physical dynamical system on a computer is made difficult by various aspects. First, the parameters of the dynamical systems are seldom exactly known. Secondly, the simulation is usually not exact: real numbers are usually represented by floating-point numbers, and simulations of cellular automata only simulate the behavior of finite or periodic configurations. For some chaotic systems, this means that the simulation can be completely irrelevant.

4 Application domains

4.1 Quantum Computing

Quantum Computing is currently the most promising technology to extend Moore's law, whose end is expected to be reached soon with engraving technologies struggling to reduce transistor size. Thanks to the exponential computational power quantum computing will bring, it will represent a decisive competitive advantage for those who will control it.

Quantum Computing is also a major security issue, since it allows us to break today's asymmetric cryptography. Hence, mastering quantum computing is also of the highest importance for national security concerns. Small-scale quantum computers already exist and recent scientific and technical advances suggest that the construction of the first *practical* quantum computers will be possible in the coming years.

As a result, the major US players in the IT industry have embarked on a dramatic race, mobilizing huge resources: IBM, Microsoft, Google and Intel have each invested huge sums of money, and are devoting significant budgets to attract and hire the best scientists on the planet. Some states have launched ambitious national programs, including Great Britain, the Netherlands, Canada, China, Australia, Singapore, and very recently Europe, with the 10-year FET Flagship program in Quantum Engineering. The French government also recently announced its Plan Quantique – a 1.8 billion euro initiative to develop quantum technologies.

While a large part of these resources are going towards R&D in quantum hardware, there is still an important need and real opportunities for leadership in the field of quantum software.

The Mocqua team contributes to the computer science approach to quantum computing, aka the quantum software approach. We aim at a better understanding of the power and limitations of the quantum computer, and therefore of its impact on society. We also contribute to ease the development of the quantum computer by filling the gap between the theoretical results on quantum algorithms and complexity and the recent progress in quantum hardware.

4.2 Higher-Order Computing

The idea of considering functions as first-class citizens and allowing programs to take functions as inputs has emerged since the very beginning of theoretical computer science through Church's λ-calculus and is nowadays at the core of functional programming, a paradigm that is used in modern software and by digital companies (Google, Facebook, ...). In the meantime higher-order computing has been explored in many ways in the fields of logic and semantics of programming languages.

One of the central problems is to design programming languages that capture most of, if not all, the possible ways of computing with functions as inputs. There is no Church thesis in higher-order computing and many ways of taking a function as input can be considered: allowing parallel or only sequential computations, querying the input as a black-box or via an interactive dialog, and so on.

The Kleene-Kreisel computable functionals are arguably the broadest class of higher-order continuous functionals that could be computed by a machine. However their complexity is such that no current programming language can capture all of them. Better understanding this class of functions is therefore fundamental in order to identify the features that a programming language should implement to make the full power of higher-order computation expressible in such a language.

4.3 Simulation of Dynamical Systems by Cellular Automata

We aim at developing various tools to simulate and analyse the dynamics of spatially-extended discrete dynamical systems such as cellular automata. The emphasis of our approach is on the evaluation of the robustness of the models under study, that is, their capacity to resist various perturbations.

In the framework of pure computational questions, various examples of such systems have already been proposed for solving complex problems with a simple bio-inspired approach (e.g. the decentralized gathering problem [48]). We are now working on their transposition to various real-world situations. For example when one needs to understand the behaviour of large-scale networks of connected components such as wireless sensor networks. In this direction of research, a first work has been presented on how to achieve a decentralized diagnosis of networks made of simple interacting components and the results are rather encouraging [5]. Nevertheless, there are various points that remain to be studied in order to complete this model for its integration in a real network.

We have also tackled the question of the evaluation of the robustness of a swarming model proposed by A. Deutsch to mimic the self-organization process observed in various natural systems (birds, fishes, bacteria, etc.) [2]. We now wish to develop our simulation tools to apply them to various biological phenomena where a great number of agents are implied.

We are also currently extending the range of applications of these techniques to the field of economy. We have started a collaboration with Massimo Amato, a professor in economy at the Bocconi University in Milan. Our aim is to examine how to propose a decentralized view of a business-to-business market and propose agent-oriented and totally decentralized models of such markets. Various banks and large businesses have already expressed their interest in such modelling approaches.

5 Highlights of the year

5.1 Awards

- Renaud Vilmart won the Gilles Kahn Award (Accessit) for his PhD thesis. In his PhD thesis, Vilmart describes a sound and complete axiomatisation of qubit quantum computation based on the ZX-calculus which does not rely on the usual linear-algebraic formalism.
- Our L3 student Antonin Callard obtained the VCLA Outstanding Undergraduate Research Award 2020. In this work, published in [24], Callard and Hoyrup study the relationship between topological and symbolic complexity, and to the topological properties of the underlying space.

6 New software and platforms

6.1 New software

6.1.1 FiatLux

Keywords: Cellular automaton, Multi-agent, Distributed systems

Scientific Description: FiatLux is a discrete dynamical systems simulator that allows the user to experiment with various models and to perturb them. It includes 1D and 2D cellular automata, moving agents, interacting particle systems, etc. Its main feature is to allow users to change the type of updating, for example from a deterministic parallel updating to an asynchronous random updating. FiatLux has a Graphical User Interface and can also be launched in a batch mode for the experiments that require statistics.

Functional Description: FiatLux is a cellular automata simulator in Java specially designed for the study of the robustness of the models. Its main distinctive features is to allow to perturb the updating of the system (synchrony rate) and to perturb the topology of the grid.

News of the Year: The latest version of the software incorporates improvements that were implemented by Océane Chazé, a second-year student at Télécom Nancy Engineering School. She improved the User Interface by simplifying the interactions with the grid of cells. She also improved the software by introducing new features which allow users to encode initial conditions and transitions rules in a more elaborate way. This allows users to import some well-known models from external websites, e.g. Langton's self-reproducing rule or "classical" patterns in the Game of Life.

URL: `https://project.inria.fr/fiatlux/`

Authors: Nazim Fatès, Olivier Boure

Contact: Nazim Fatès

Participants: Nazim Fatès, Olivier Boure

Partners: ENS Lyon, Université de Lorraine

6.1.2 ComplexityParser

Keywords: Complexity, Static typing, Parsing

Functional Description: ComplexityParser is a static complexity analyzer of Java programs written in Java (approximatively 5000 lines of code). The program consists in a type inference and checking program based on the data tiering principle. It allows the program to certify that the typed program has a polynomial time complexity.

Contacts: Emmanuel Hainry, Romain Péchoux

Participants: Olivier Zeyen, Emmanuel Hainry, Romain Péchoux, Emmanuel Jeandel

7 New results

7.1 Descriptive complexity on represented spaces

- Participants: Antonin Callard, Mathieu Hoyrup

Programs are able to perform computations on infinite objects such as streams of bits. It enables one to perform computation on infinite objects that can be represented by such streams, like real numbers for instance. Such infinite computations are related to topology, because computable functions are necessarily continuous (a finite segment of the output only depends on a finite segment of the input). The class of topological spaces whose points can be faithfully represented by infinite streams is well-known.

We study the descriptive complexity of subsets of such spaces. The classical one is topological complexity, which is the simplest way to express a set in terms of open sets using boolean operators; we introduce the symbolic complexity of a set, which is the topological complexity of the corresponding set of streams. The symbolic complexity is what is relevant for computations. While it is known that these two notions of complexity coincide for simple spaces (countably-based spaces), we investigate what happens in general spaces.

In [24] we show that the equivalence is even computable in countably-based spaces, and that these spaces are the only ones where it is so. We investigate a larger class of spaces, called co-Polish spaces, and relate the difference between symbolic and topological complexities to a topological property of the space (Fréchet-Urysohn).

In [29] we investigate spaces of open sets of a space X and show very precisely how the relationship between symbolic and topological complexity depend on the compactness properties of X.

7.2 Algorithmic properties of sets

- Participants: Djamel Eddine Amir, Mathieu Hoyrup

Several computability notions are available for subsets of the plane, or of other topological spaces. For instance, whether the famous Mandelbrot set is computable is an open problem, but from its definition it is easy to see that it is semicomputable (one can eventually detect that a point is outside, but not that it is inside).

It turns out that the topological properties of a set have a dramatic impact on its algorithmic properties: it was proved for instance in [55, 51] that if a set is homeomorphic to an n-dimensional sphere, more generally an n-dimensional manifold, then it is computable if and only if it is semicomputable. We say that the set has computable type.

The PhD project of Djamel Eddine Amir, started in October 2020 after an M2 internship, is to develop a general and systematic approach to this problem, with several questions to solve. The general goal is to identify which topological properties imply having computable type. We study how it relates to the algorithmic complexity of testing the property. We want to develop general and elegant tools to prove or disprove that a set has computable type.

Our first results are promising, as they already subsume the existing results and can give much insight into the problem. An article is in preparation.

7.3 Probabilistic cellular automata for problem solving

- Participants: Nazim Fatès, Emmanuel Jeandel, Irène Marcovici, Amaury Saint-Jore

Directly related to the theme exposed in Sec. 4.3, we continued to explore the problem of self-stabilisation, as introduced by Dijkstra in the 1970's, in the context of cellular automata [49]. More precisely, we extended the scope of our results from k-colourings to shift of finite type. We also presented various constructions and theorems that allow us to consolidate the framework of this problem.

We presented a bio-inspired mechanism for data clustering [44]. Our method uses amoebae which evolve according to cellular automata rules: they contain the data to be processed and emit reaction-diffusion waves at random times. The waves transmit the information across the lattice and cause other amoebae to react, by being attracted or repulsed. The local reactions produce small homogeneous groups which progressively merge and realise the clustering at a larger scale. Despite the simplicity of the local rules, interesting complex behaviour occurs, which make the model robust to various changes of its settings. We evaluated this prototype with a simple task: the separation of two groups of integer values distributed according to Gaussian laws and tested it on the famous Iris dataset by Fischer[1].

We deepened the analysis of the problem of detecting failures in a distributed network [50]. The question that drives our research is to find out how we can detect that the failure rate has exceeded a given threshold without any central authority when some components progressively break down. We started a collaboration on this topic with Régine Marchand (IECL, Université de Lorraine).

Our tutorial on the convergence properties of the 256 Elementary Cellular Automata under the fully asynchronous updating was published [20]. We presented a panorama of the different qualitative behaviours that arise when only one cell is updated at each time step. We made a synthesis of the results which had been presented in different articles and exposed a full analysis of the behaviour of finite systems with periodic boundary conditions. Our classification relies on the scaling properties of the average convergence time to a fixed point. We presented the different scaling laws that can be found, which fall in one of the following classes: logarithmic, linear, quadratic, exponential and non-converging. The techniques for quantifying this behaviour rely mainly on Markov chain theory and martingales. Most behaviours can be studied analytically but there are still many rules for which obtaining a formal characterisation of their convergence properties is still an open problem.

7.4 Diagrammatic quantum computing

- Participants: Titouan Carette, Alexandre Clément, Ross Duncan (Strathclyde University), Emmanuel Jeandel, Aleks Kissinger (University of Oxford), Simon Perdrix, Renaud Vilmart (Inria Saclay).

This year, we have contributed in several ways to the foundations and the applications of the ZX-calculus, a diagrammatic language for quantum computing.

Emmanuel Jeandel, Simon Perdrix, and Renaud Vilmart have published a journal paper [23] on the completeness of the ZX-calculus. This is a special issue of LiCS'18, presenting an extended version of the two contributions of the authors at LICS'18.

Ross Duncan (Strathclyde University), Aleks Kissinger (University of Oxford), Simon Perdrix, and John van de Wetering (Radboud University Nijmegen), have introduced a ZX-based optimising method of quantum circuits. This method strongly relies on the GFlow technics originally introduced by Browne, Kashefi, Mhalla and Perdrix in the context of Measurement-based quantum computing. The method introduced in this paper has been implemented by two of the authors (Kissinger and van de Wetering) leading to one of the currently best quantum circuit optimisation methods [52].

Emmanuel Jeandel and Titouan Carette have presented a complete characterization of all possible graphical languages for quantum computing. Surprisingly, all possible languages had already been found in the form of the ZX-calculus, the ZW-calculs and the ZH-calculus. This work was published at ICALP [25].

7.5 Coherent Control of Quantum Computations

- Participants: Alexandre Clément, Simon Perdrix.

Alexandre Clément and Simon Perdrix have introduced a new graphical language, the PBS-calculus, to represent and reason on quantum computations involving coherent control of quantum operations. Coherent control, and in particular indefinite causal order, is known to enable multiple computational and communication advantages over classically ordered models like quantum circuits. The PBS-calculus is inspired by quantum optics, in particular the polarising beam splitter. The language is equipped with an equational theory, proved to be sound and complete: two diagrams are representing the same quantum evolution if and only if one can be transformed into the other using the rules of the PBS-calculus. The equational theory is also proved to be minimal. This article has been published at MFCS'20 [26].

[1] See for instance: https://en.wikipedia.org/wiki/Iris_flower_data_set

7.6 Contextuality in multipartite pseudo-telepathy graph games

- Participants: Anurag Anshu (Perimeter Institute), Peter Høyer (U. Calgary), Mehdi Mhalla (LIG), and Simon Perdrix.

Analyzing pseudo-telepathy graph games, Anurag Anshu, Peter Høyer, Mehdi Mhalla, and Simon Perdrix proposed a way to build contextuality scenarios exhibiting the quantum advantage using graph states. A new tool, called multipartiteness width, is introduced to investigate which scenarios are hard to decompose and to show that there exist graphs generating scenarios with a linear multipartiteness width. These results have been published in the Journal of Computer and System Science [13].

7.7 Quantum Programming with Inductive Datatypes

- Participants: Romain Péchoux, Simon Perdrix, Mathys Rennela (Leiden, The Netherlands), Vladimir Zamdzhiev

Inductive datatypes in programming languages allow users to define useful data structures such as natural numbers, lists, trees, and others. However, in quantum programming, working with such datatypes is more complicated, because quantum computer science is usually concerned with finite-dimensional quantum structures, whereas inductive quantum datatypes are inherently infinite-dimensional.

In this work, we show how inductive datatypes may be used in quantum programming by describing a quantum programming language which has a formal syntax and a type-safe operational semantics. We also describe a sound mathematical model for the language and by doing so we provide the first detailed semantic treatment of user-defined inductive datatypes in quantum programming. Our semantics is entirely based on a physically natural model of von Neumann algebras, which are mathematical structures used by physicists to study quantum mechanics. Finally, we cement our results by showing our mathematical model is also computationally adequate in a strong sense.

This work was published in FoSSaCS 2020 [30] and an extended version was submitted to the Theoretical Computer Science journal (preprint: [43]).

7.8 Recursive Types in Substructural Type Systems

- Participants: Bert Lindenhovius (Tulane University, USA) Michael Mislove (Tulane University, USA), Vladimir Zamdzhiev

Recursive datatypes in programming languages generalise inductive datatypes by allowing the programmer to construct more expressive types (e.g. streams) without imposing any restrictions on the admissible logical polarities of the underlying type expressions. As part of a larger program, we wish to understand how this can be achived in quantum computation. However, before this question may be answered, it is useful to understand how such types behave in the setting of Substructural Type Systems, which quantum programming is a part of.

In this work, we describe a substructural type system with mixed linear and non-linear recursive types called LNL-FPC (the linear/non-linear fixpoint calculus). Just as in FPC, we show that LNL-FPC supports type-level recursion which in turn induces term-level recursion. We also provide sound and computationally adequate categorical models for LNL-FPC which describe the categorical structure of the substructural operations of Intuitionistic Linear Logic at all non-linear types, including the recursive ones. In order to do so, we describe a new technique for solving recursive domain equations. We also show that the requirements of our abstract model are reasonable by constructing a large class of concrete models that have found applications in classical programming, but also in emerging programming paradigms such as quantum programming and circuit description programming languages.

This work has been accepted for publication at the journal Logical Methods in Computer Science (subject to minor revisions which will be done soon) and a preprint is available at [41]. This work is also an extended version of our paper that was published in ICFP'19 [54].

7.9 Programming Quantum Circuits

- Participants: Bert Lindenhovius (Tulane University, USA) Michael Mislove (Tulane University, USA), Vladimir Zamdzhiev

Quantum algorithms and protocols are often described in terms of quantum circuits, which are diagrammatic representations of quantum primitives and operations. In this work, we show how a programming language can be used to construct such quantum circuits by describing a syntax, type-safe operational semantics (with recursion) and also by constructing a mathematical model for the language which is based on ideas from enriched category theory. This work is accepted for publication at the special issue on *Outstanding Contributions to Logic (Volume for Samson Abramsky)* and a preprint is available at [42]. It is an extended version of our LICS'18 paper [53].

7.10 Models for Quantum and Substructural Languages

- Participants: Vladimir Zamdzhiev

In this work we consider recursion for quantum and substructural programming languages and we describe a large class of mathematical models for these languages. We identify the fundamental categorical structures that are required to establish semantic properties such as soundness and adequacy. The results are presented at a large level of generality and we recover many known concrete models as special cases of our treatment. This work has been described in papers presented at ACT 2020 [31] and CMCS 2020 [32].

7.11 Characterizing type-2 polynomial time

- Participants: Emmanuel Hainry, Bruce Kapron (University of Victoria), Jean-Yves Marion (LORIA), Romain Péchoux

We adapt the tiering technique to an imperative programming language with oracles. This work is inspired by the work of Kapron and Steinberg and the restrictions they developed on oracle Turing machines in order to characterize the class of second order polynomial time computable functionals, called Basic Feasible Functionals (BFF), and known to be the second order extension of the standard notion of (first order) polynomial time computable function. We have provided the first tractable characterization of BFF using a tier-based type discipline. This work has been described in a paper presented at LICS 2020 [27].

7.12 Characterizing polynomial time over the reals

- Participants: Emmanuel Hainry, Damiano Mazza (CNRS), Romain Péchoux

We adapt the parsimonious calculus introduce by Mazza and Terui to the case of functions over the reals. The parsimonious calculus is a programming language on streams (infinite lists), which uses a linear logic based type discipline. Real numbers are encoded as infinite sequences of signed binary digits. We show that, under some restrictions, the calculus is sound and complete for functions computable in polynomial time over the reals in the sense of Ko; that is, the n-th digit of the output can be computed in time polynomial in n. This work has been described in a paper presented at FLOPS 2020 [28].

7.13 Enumeration of phylogenetic networks

- Participants: Mathilde Bouvel, Philippe Gambette (LIGM, Université Gustave Eiffel), Marefatollah Mansouri (TU Wien)

Phylogenetic networks generalize phylogenetic trees, and have been introduced in order to describe evolution in the case of transfer of genetic material between coexisting species. There are many classes of phylogenetic networks, which can all be modeled as families of graphs with labeled leaves. In this work, we focus on rooted and unrooted level-k networks and provide enumeration formulas (exact and asymptotic) for rooted and unrooted level-1 and level-2 phylogenetic networks with a given number of leaves. We also prove that the distribution of some parameters of these networks (such as their number of cycles) are asymptotically normally distributed. These results are obtained by first providing a recursive description (also called combinatorial specification) of our networks, and by next applying classical methods of enumerative, symbolic and analytic combinatorics. This has been achieved combining the different backgrounds of the participants: bioinformatics and combinatorics. This work has been published in [16].

7.14 Universality in permuton limits for permutation classes

- Participants: Frédérique Bassino (LIPN, Univ. Paris 13), Mathilde Bouvel, Valentin Féray (IECL, Univ. Lorraine), Lucas Gerin (CMAP, École polytechnique), Mickaël Maazoun (Oxford), Adeline Pierrot (LRI, Univ. Paris-Sud)

Among the topics at the interface of combinatorics and probability theory, the study of non-uniform random permutations has recently received increased interest. This is notably due to the definition of the permuton framework to describe limits of permutations, and to the description of permuton limits of some families of constrained permutations. Permutons are probability measures on the unit square with uniform marginals, which naturally extend to the continuous setting the representation of permutations *via* their diagrams, or equivalently their permutation matrices.

In this work, we consider constrained random permutations, taken uniformly inside proper substitution-closed classes and we study their limiting behavior in the sense of permutons. The limit depends on the generating series of the simple permutations in the class. Under a mild sufficient condition, the limit is an elementary one-parameter deformation of the limit of uniform separable permutations, which we previously identified as the Brownian separable permuton. This limiting object is therefore in some sense universal. We identify two other regimes with different limiting objects. The first one is degenerate; the second one is nontrivial and related to stable trees. These results are obtained thanks to a characterization of the convergence of random permutons through the convergence of their expected pattern densities. The limit of expected pattern densities is then computed by using the substitution tree encoding of permutations and performing singularity analysis on the tree series. This work has been published in [15].

8 Partnerships and cooperations

8.1 International initiatives

8.1.1 Inria associate team not involved in an IIL

Title: *TCPRO3*

Duration: 2019 - 2022

Coordinator: Romain Péchoux

Partners: Computational Logic Group, University of Innsbruck (Austria)

Inria contact: Romain Péchoux

Summary: Probabilistic languages consist in higher-order functional, imperative languages, and reduction systems with sampling and conditioning primitive instructions. While deep theoretical results have been established on the semantic properties of such languages, applications of termination and complexity analysis are restricted to academic examples so far. The associate team TCPro3 has the aim to contribute to the field by developing methods for reasoning on quantitative properties of probabilistic programs and models. Extensions of these methods on quantum programs will be studied.

8.1.2 Participation in other international programs

- Quantum Calculi (QuCa), ECOS Sud A17C03, funded by MinCyT and ECOS France (2018-2020)

8.2 European initiatives

8.2.1 FP7 & H2020 Projects

- Marie-Curie RISE Project (2017-2021) “Computing with Infinite Data”. Mathieu Hoyrup is the local coordinator.

  Acronym: CID

  Title: Computing with Infinite Data

  Duration: 2017 - 2020 (extended to 2021 or 2022 because of COVID)

  Coordinator: Dieter Spreen, University of Siegen

  Local coordinator: Mathieu Hoyrup

  Partners: 22 universities or research institutes worldwide

  Summary: The proposal brings together internationally leading researchers from both inside and outside Europe to work on theoretical and applied aspects of computing with infinite data. Ways to classify the computational hardness of problems involving infinite data will be studied as well as the question what logical structures are needed to facilitate correctness proofs. Novel computable and constructive approaches to probability and randomness will be pursued, and the computability and complexity of problems in dynamical systems investigated. Systems of this type abound in applications and are known for the hardness of their problems.

- NEASQC (Horizon 2020 RIA, FET Flagship, 2020-2024) `https://www.neasqc.eu` with AstraZeneca, ATOS-Bull (PI), CESGA, EDF, HQS, HSBC, ICHEC, TILDE, Total, University of Coruña, University of Leiden, and Loria. Emmanuel Jeandel is the local coordinator. The NEASQC (NExt ApplicationS of Quantum Computing) project brings together academic experts and industrial end-users to investigate and develop a new breed of Quantum-enabled applications that can take advantage of NISQ (Noise Intermediate-Scale Quantum) systems in the near future. NEASQC is use-case driven, addressing practical problems such as drug discovery, CO2 capture, smart energy management, natural language processing, breast cancer detection, probabilistic risk assessment for energy infrastructures or hydrocarbon well optimisation. NEASQC has the ambition to initiate an active European Community around NISQ Quantum Computing by providing a common toolset that will attract new industrial users.

8.3 National initiatives

8.3.1 ANR

- Project acronym: ANR PRCE SoftQPro (ANR-17-CE25-0009)
  Project title: Solutions logicielles pour l’optimisation des programmes et ressources quantiques.
  Duration: Dec. 2017 - Dec. 2022
  Coordinator: Simon Perdrix
  Other partners: Atos-Bull, LRI, CEA-Saclay.
  Participants: Simon Perdrix, Emmanuel Jeandel, Emmanuel Hainry, and Romain Péchoux
  Abstract: Quantum computers can theoretically solve problems out of reach of classical computers. We aim at easing the crucial back and forth interactions between the theoretical approach to quantum computing and the technological efforts made to implement the quantum computer. Our software-based quantum program and resource optimisation (SoftQPRO) project consists in developing high level techniques based on static analysis, certification, transformations of quantum graphical languages, and optimisation techniques to obtain a compilation suite for quantum programming languages. We will target various computational model back-ends (e.g. QRAM, measurement-based quantum computations) as well as classical simulation. Classical simulation is central in the development of the quantum computer, on both ends: as a way to test quantum programs but also as a way to test quantum computer prototypes. For this reason we aim at designing sophisticated simulation techniques on classical high-performance computers (HPC).

- Project acronym: ANR PRCI VanQuTe (ANR-17-CE24-0035)
Project title: Validation of near-future quantum technologies.
Duration: Fev. 2018 - Jan. 2022
Coordinator: Damian Markham (Laboratoire d'informatique de Paris 6)
Other partners: NTU (Nanyang Technological University), SUTD (Singapore University of Technology and Design), NUS (Nationl University of Singapore), LIP6 (Laboratoire d'informatique de Paris 6)
Participants: Simon Perdrix, Emmanuel Jeandel
Abstract: In the last few years we have seen unprecedented advances in quantum information technologies. Already quantum key distribution systems are available commercially. In the near future we will see waves of new quantum devices, offering unparalleled benefits for security, communication, computation and sensing. A key question to the success of this technology is their verification and validation.

  Quantum technologies encounter an acute verification and validation problem: On one hand, since classical computations cannot scale-up to the computational power of quantum mechanics, verifying the correctness of a quantum-mediated computation is challenging. On the other hand, the underlying quantum structure resists classical certification analysis. Members of our consortium have shown, as a proof-of-principle, that one can bootstrap a small quantum device to test a larger one. The aim of VanQuTe is to adapt our generic techniques to the specific applications and constraints of photonic systems being developed within our consortium. Our ultimate goal is to develop techniques to unambiguously verify the presence of a quantum advantage in near future quantum technologies.

8.3.2 Other initiatives

- Quantex. Project acronym: PIA-GDN/Quantex. (initially an ITEA3 project finally funded by the *Grands défis du Numérique / Programme d'investissements d'avenir*).
Project title: Simulation/Emulation of Quantum Computation.
Duration: Feb. 2018 - Jan 2021.
Coordinator: Huy-Nam Nguyen (Atos Bull).
Other partners: Atos-Bull, LRI, CEA Grenoble.
Participants: Simon Perdrix (WP leader), Emmanuel Jeandel
Abstract: The lack of quantum computers leads to the development of a variety of software-based simulators to assist in the research and development of quantum algorithms. This proposal focuses on the development of a combined software-based and hardware-accelerated toolbox for quantum computation. A quantum computing stack including specification language, libraries and optimisation/execution tools will be built upon a well-defined mathematical framework mixing classical and quantum computation. Such an environment will be dedicated to support the expression of quantum algorithms for the purpose of investigation and verification.

9 Dissemination

9.1 Promoting scientific activities

9.1.1 Scientific events: organisation

Member of the organizing committees

- Vladimir Zamdzhiev: Organiser for the international conference MFPS 2020 (virtual).

- Vladimir Zamdzhiev: Organiser for the international conference QPL 2020 (virtual).

- Mathieu Hoyrup: Organiser of a special session on Computable Topology at CiE 2020.

9.1.2 Scientific events: selection

Member of the conference program committees

- Mathieu Hoyrup was on the PC of CiE'20.
- Emmanuel Jeandel was on the PC of LICS'20.
- Nazim Fatès was a member of the PC of AUTOMATA'20 and ACRI'20.

Reviewer

- Nazim Fatès: Reviewer for STACS'21.
- Mathieu Hoyrup: Reviewer for CiE, CSL, LICS, MFCS, STACS.
- Romain Péchoux: Reviewer for LICS 2020, ICALP 2020(x2), and ISMVL 2020.
- Vladimir Zamdzhiev: Reviewer for CSL 2021, QIP 2020, FoSSaCS 2020, LICS 2020(x3). Reviewer for the special issue Outstanding Contributions to Logic (Volume for Samson Abramsky).

9.1.3 Journal

Member of the editorial boards

- Mathilde Bouvel: Member of the editorial board of the journal "Annals of Combinatorics".
- Mathilde Bouvel: Guest editor for a special issue of the journal DMTCS (Discrete Mathematics and Theoretical Computer Science), volume 22, no.2, following the conference Permutation Patterns 2019.
- Nazim Fatès: Guest co-editor for a special issue of the *Physica D* journal, following the conference SOLSTICE'19.
- Emmanuel Jeandel: Member of the editorial board of the journal RAIRO-ITA.

Reviewer - reviewing activities

- Mathilde Bouvel: Reviewer for JCT-A (Journal of Combinatorial Theory – Series A) and Discrete Math. (Discrete Mathematics).
- Nazim Fatès : Reviewer for the *SIAM Journal on Discrete Mathematics* (SIDMA).
- Mathieu Hoyrup: Information and Computation, Journal of Symbolic Logic, Theory of Computing Systems.
- Romain Péchoux: Science of Computer Programming, Logical Methods in Computer Science.
- Vladimir Zamdzhiev: Reviewer for LMCS (Logical Methods in Computer Science), TOPLAS (ACM Transactions on Programming Languages and Systems), ACS (Applied Categorical Structures).

9.1.4 Invited talks

- Vladimir Zamdzhiev: Inductive and Recursive Types for Quantum Programming. Invited talk at the joint special session of the international conferences QPL 2020 (Quantum Physics and Logic) and MFPS 2020 (Mathematical Foundations of Programming Semantics) on Quantum Programming Languages. June 2020. [Video].
- Mathieu Hoyrup: Descriptive Complexity on Represented Spaces. Invited talk at CCA 2020.

9.1.5 Scientific expertise

- Romain Péchoux: Expert and Rapporteur for H2020 Marie Sklodowska–Curie Actions, Individual Fellowships, 2020, European Research Agency.

9.1.6 Research administration

- Simon Perdrix is an elected member and scientific secretary of the CNRS section 6.

9.2 Teaching - Supervision - Juries

Teaching

- Licence
  - Isabelle Gnaedig:
    * To the limits of the computable, 6 hours, Opening course-conference of the collegium "Lorraine INP", Université de Lorraine, Nancy, France.
  - Emmanuel Hainry:
    * Operating Systems, 30h, L1, IUT Nancy Brabois, Université de Lorraine, Nancy, France.
    * Algorithmics, 40h, L1, IUT Nancy Brabois.
    * Dynamic Web, 60h, L1, IUT Nancy Brabois.
    * Databases, 30h, L1, IUT Nancy Brabois.
    * Object Oriented Languages, 16h, L2, IUT Nancy Brabois.
    * Complexity, 30h, L2, IUT Nancy Brabois.
  - Emmanuel Jeandel:
    * Algorithmics and Programming 1, 60h, L1 Maths-Info, Université de Lorraine, Nancy, France.
    * Algorithmics and Programming 4, 30h, L3 Informatique.
    * Networking, 15h, L3 Informatique.
    * Formal Languages, 30h, L2 Informatique.
    * Functional Programming, 14h, L3 Informatique.
- Master
  - Isabelle Gnaedig:
    * Rule-based Programming, 24 hours, M2, Telecom-Nancy, Université de Lorraine, Nancy, France.
  - Mathieu Hoyrup:
    * Mathematics for Computer Science, 15h, M1 TAL, Institut des Sciences du Digital: Management et Cognition, Université de Lorraine, Nancy, France.
  - Emmanuel Jeandel:
    * Algorithmics and Complexity, 30h, M1 Informatique, Université de Lorraine, Nancy, France.
    * Quantum Computing, 15h, M1 Informatique.
    * Combinatorial Optimization, 24h, M1 Informatique.
  - Nazim Fatès:
    * Systèmes distribués adaptatifs, 10h, Master 2, informatique, Université de Lorraine, Nancy, France.
    * Agents intelligents et collectifs, 15h, Master 1, sciences cognitives, Université de Lorraine, Nancy, France.
    * Models of computation, 12h, Master 2, informatique, Université de Lorraine, Nancy, France.
    * IAE Nancy-School of Management, Marketing et Gestion Commerciale, Lecture on Artifical intelligence and Ethics, 3h, Université de Lorraine, Nancy, France.

Supervision

- Mathieu Hoyrup supervised the M2 internship of Djamel Eddine Amir (Université Aix-Marseille), about computability of subsets of topological spaces.
- Emmanuel Hainry and Romain Péchoux supervised the internship of Théo Alison, Matthias Bertrand and Clément Koch (M1, FST, Université de Lorraine), on Shape Analysis for Complexity of Java Programs.
- Nazim Fatès supervised the internship of Émilien Brun, first-year student at the CESI engineering school, Océane Chazé and Ophélie Phochareun, both second-year students at the Telecom Nancy engineering school.
- PhD in progress: Djamel Eddine Amir, "Computability of subsets of topological spaces", Start: October 2020
- PhD in progress: Titouan Carette, "Langage diagrammatique pour l'ordinateur quantique", Start: October 2018, Advisors: Emmanuel Jeandel and Simon Perdrix.
- PhD: Pierre Mercuriali, "Calcul à base de médiane et structures médianes pour la classification", Start: October 2016, Advisors: Miguel Couceiro and Romain Péchoux. Pierre defended his PhD in december 2020.
- PhD in progress: Robert Booth, "Formalismes pour la vérification de technologies quantiques", Start: November 2018, Advisors: Damian Markham and Simon Perdrix.
- PhD in progress: Alexandre Clément, "Graphical Languages for Quantum Control", Start: September 2019, Advisors: Emmanuel Jeandel and Simon Perdrix.
- PhD in progress: Margarita Veshchezerova, "Quantum Computing for Combinatorial Optimisation", Start: October 2019, Advisors: Emmanuel Jeandel and Simon Perdrix, joint supervision with Marc Porcheron at EDF (CIFRE).

Juries

- Emmanuel Jeandel participated in the PhD defense of Pierre-Adrian Tahay (Université de Lorraine) and Paul Huyhn (Université de Lorraine).
- Emmanuel Jeandel reviewed the HdR manuscript of Giuseppe di Molfetta (Aix-Marseille Université)
- Mathilde Bouvel was part of the PhD defense committee of Mohamed Slim Kammoun, at University of Lille, defended on October 22, 2020.

HDR

- Romain Péchoux. Implicit Computational Complexity: past and future. University of Lorraine, defended on October 19, 2020.

9.3 Popularization

9.3.1 Education

Nazim Fatès co-supervised a project for making the scenario of short films on artificial intelligence, with eight Master's 1 students from the IECA (Institut Européen de Cinéma et d'Audiovisuel).

9.3.2 Interventions

Nazim Fatès participated to a one-day colloquium dedicated to the theme of *transhumanism* and presented a talk entitled « Penser avec l'intelligence artificielle » in the Lycée Saint-Sigisbert de Nancy, March 2020.

Nazim Fatès gave a three-hour lecture on the history of computing and artificial intelligence for a formation of high-school teachers of the region (Diplôme Inter-Universitaire Enseigner l'Informatique au Lycée).

10 Scientific production

10.1 Major publications

[1] A. Callard and M. Hoyrup. 'Descriptive complexity on non-Polish spaces'. In: *STACS 2020 - 37th Symposium on Theoretical Aspects of Computer Science*. Ed. by S. D.-.-L.-Z. fuer Informatik. Vol. 154. Montpellier, France, Mar. 2020, p. 16. DOI: 10.4230/LIPIcs.STACS.2020.8. URL: https://hal.inria.fr/hal-02298815.

[2] N. Fatès, V. Chevrier and O. Bouré. 'Is there a trade-off between simplicity and robustness? Illustration on a lattice-gas model of swarming'. In: *Probabilistic Cellular Automata*. Ed. by P.-Y. Louis and F. R. Nardi. Emergence, Complexity and Computation. Springer, 2018. DOI: 10.1007/978-3-319-65558-1_16. URL: https://hal.inria.fr/hal-01230145.

[3] N. Fatès, I. Marcovici and S. Taati. 'Two-dimensional traffic rules and the density classification problem'. In: *International Workshop on Cellular Automata and Discrete Complex Systems, AUTOMATA 2016*. Vol. 9664. Lecture Notes of Computer Science. Zürich, France, June 2016. DOI: 10.1007/978-3-319-39300-1_11. URL: https://hal.inria.fr/hal-01290290.

[4] H. Férée, E. Hainry, M. Hoyrup and R. Péchoux. 'Characterizing polynomial time complexity of stream programs using interpretations'. In: *Journal of Theoretical Computer Science (TCS)* 585 (Jan. 2015), pp. 41–54. DOI: 10.1016/j.tcs.2015.03.008. URL: https://hal.inria.fr/hal-01112160.

[5] N. Gauville. 'Système robuste de diagnostic décentralisé à l'aide d'automates cellulaires simples'. MA thesis. Université de Lorraine (Nancy), Sept. 2018. URL: https://hal.inria.fr/hal-01894581.

[6] I. Gnaedig and H. Kirchner. 'Proving Weak Properties of Rewriting'. In: *Theoretical Computer Science* 412 (2011), pp. 4405–4438. DOI: 10.1016/j.tcs.2011.04.028. URL: http://hal.inria.fr/inria-00592271/en.

[7] E. Hainry, B. Kapron, J.-Y. Marion and R. Péchoux. 'A tier-based typed programming language characterizing Feasible Functionals'. In: *LICS '20 - 35th Annual ACM/IEEE Symposium on Logic in Computer Science*. Saarbrücken, Germany: ACM, July 2020, pp. 535–549. DOI: 10.1145/3373718.3394768. URL: https://hal.inria.fr/hal-02881308.

[8] E. Hainry and R. Péchoux. 'Objects in Polynomial Time'. In: *APLAS 2015*. Ed. by X. Feng and S. Park. Vol. 9458. Lecture Notes in Computer Science. Pohang, South Korea: Springer, Nov. 2015, pp. 387–404. DOI: 10.1007/978-3-319-26529-2_21. URL: https://hal.inria.fr/hal-01206161.

[9] M. Hoyrup. 'Descriptive complexity on non-Polish spaces II'. In: *ICALP*. Saarbrücken, Germany, July 2020. DOI: 10.4230/LIPIcs.ICALP.2020.132. URL: https://hal.inria.fr/hal-02483114.

[10] M. Hoyrup and W. Gomaa. 'On the extension of computable real functions'. In: *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*. IEEE Computer Society, 2017, pp. 1–12. DOI: 10.1109/LICS.2017.8005067. URL: https://doi.org/10.1109/LICS.2017.8005067.

[11] E. Jeandel, S. Perdrix and R. Vilmart. 'A Complete Axiomatisation of the ZX-Calculus for Clifford+T Quantum Mechanics'. In: *The 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018*. Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science. Oxford, United Kingdom, July 2018, pp. 559–568. DOI: 10.1145/3209108.3209131. URL: https://hal.archives-ouvertes.fr/hal-01529623.

[12] E. Jeandel, S. Perdrix and R. Vilmart. 'Diagrammatic Reasoning beyond Clifford+T Quantum Mechanics'. In: *The 33rd Annual Symposium on Logic in Computer Science*. Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science. Oxford, United Kingdom, July 2018, pp. 569–578. DOI: 10.1145/3209108.3209139. URL: https://hal.archives-ouvertes.fr/hal-01716501.

10.2 Publications of the year

International journals

[13] A. Anshu, P. Høyer, M. Mhalla and S. Perdrix. 'Contextuality in multipartite pseudo-telepathy graph games'. In: *Journal of Computer and System Sciences* 107 (Feb. 2020), pp. 156–165. DOI: 10.1016/j.jcss.2019.06.005. URL: https://hal.archives-ouvertes.fr/hal-02400051.

[14] M. Backens, S. Perdrix and Q. Wang. 'Towards a Minimal Stabilizer ZX-calculus'. In: *Logical Methods in Computer Science* (2020). URL: https://hal.inria.fr/hal-01597114.

[15] F. Bassino, M. Bouvel, V. Féray, L. Gerin, M. Maazoun and A. Pierrot. 'Universal limits of substitution-closed permutation classes'. In: *Journal of the European Mathematical Society* 22.11 (2020), pp. 3565–3639. DOI: 10.4171/JEMS/993. URL: https://hal.archives-ouvertes.fr/hal-01653572.

[16] M. Bouvel, P. Gambette and M. Mansouri. 'Counting phylogenetic networks of level 1 and 2'. In: *Journal of Mathematical Biology* 81 (1st Oct. 2020), pp. 1357–1395. DOI: 10.1007/s00285-020-01543-5. URL: https://hal-upec-upem.archives-ouvertes.fr/hal-02955527.

[17] T. Carette, M. Laurière and F. Magniez. 'Extended Learning Graphs for Triangle Finding'. In: *Algorithmica* 82.4 (2020), pp. 980–1005. DOI: 10.1007/s00453-019-00627-z. URL: https://hal-cnrs.archives-ouvertes.fr/hal-02349981.

[18] M. Couceiro, E. Lehtonen, P. Mercuriali and R. Péchoux. 'On the efficiency of normal form systems for representing Boolean functions'. In: *Theoretical Computer Science* 813 (12th Apr. 2020), pp. 341–361. DOI: 10.1016/j.tcs.2020.01.009. URL: https://hal.inria.fr/hal-02153506.

[19] R. Duncan, A. Kissinger, S. Perdrix and J. van de Wetering. 'Graph-theoretic Simplification of Quantum Circuits with the ZX-calculus'. In: *Quantum* 4 (4th June 2020), p. 279. DOI: 10.22331/q-2020-06-04-279. URL: https://hal.inria.fr/hal-02995364.

[20] N. A. Fatès. 'A tutorial on Elementary cellular automata with fully asynchronous updating - General properties and convergence dynamics'. In: *Natural Computing* 19.1 (2020), pp. 179–197. DOI: 10.1007/s11047-020-09782-7. URL: https://hal.inria.fr/hal-02400792.

[21] E. Hainry and R. Péchoux. 'Theory of Higher Order Interpretations and Application to Basic Feasible Functions'. In: *Logical Methods in Computer Science* 16.4 (14th Dec. 2020), p. 25. DOI: 10.23638/LMCS-16(4:14)2020. URL: https://hal.inria.fr/hal-02499206.

[22] E. Jeandel, E. Moutot and P. Vanier. 'Slopes of multidimensional subshifts'. In: *Theory of Computing Systems* 64.1 (2020), pp. 35–61. DOI: 10.1007/s00224-019-09931-1. URL: https://hal.inria.fr/hal-02158012.

[23] E. Jeandel, S. Perdrix and R. Vilmart. 'Completeness of the ZX-Calculus'. In: *Logical Methods in Computer Science* 16.2 (4th June 2020), 11:1–11:72. DOI: 10.23638/LMCS-16(2:11)2020. URL: https://hal.archives-ouvertes.fr/hal-02400081.

International peer-reviewed conferences

[24] A. Callard and M. Hoyrup. 'Descriptive complexity on non-Polish spaces'. In: STACS 2020 - 37th Symposium on Theoretical Aspects of Computer Science. Vol. 154. Montpellier, France: https://stacs2020.sciencesconf.org/, 10th Mar. 2020, p. 16. DOI: 10.4230/LIPIcs.STACS.2020.8. URL: https://hal.inria.fr/hal-02298815.

[25] T. Carette and E. Jeandel. 'A recipe for quantum graphical languages'. In: ICALP 2020. 47th International Colloquium on Automata, Languages, and Programming (ICALP 2020). Saarbrücken, Germany, 2020. URL: https://hal.archives-ouvertes.fr/hal-02914177.

[26] A. Clément and S. Perdrix. 'PBS-Calculus: A Graphical Language for Coherent Control of Quantum Computations'. In: 45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020). Vol. 170. Leibniz International Proceedings in Informatics (LIPIcs). Prague, Czech Republic: http://mfcs.mff.cuni.cz/2020/, 18th Aug. 2020, 24:1–24:14. DOI: 10.4230/LIPIcs.MFCS.2020.24. URL: https://hal.archives-ouvertes.fr/hal-02929291.

[27] E. Hainry, B. Kapron, J.-Y. Marion and R. Péchoux. 'A tier-based typed programming language characterizing Feasible Functionals'. In: LICS '20 - 35th Annual ACM/IEEE Symposium on Logic in Computer Science. Saarbrücken, Germany, 8th July 2020, pp. 535–549. DOI: 10.1145/3373718.3394768. URL: https://hal.inria.fr/hal-02881308.

[28] E. Hainry, D. Mazza and R. Péchoux. 'Polynomial time over the reals with parsimony'. In: FLOPS 2020 - International Symposium on Functional and Logic Programming. Akita, Japan, 23rd Apr. 2020. URL: https://hal.inria.fr/hal-02499149.

[29] M. Hoyrup. 'Descriptive complexity on non-Polish spaces II'. In: ICALP. Saarbrücken, Germany: https://drops.dagstuhl.de/opus/volltexte/2020/12539/, 8th July 2020. DOI: 10.4230/LIPIcs.ICALP.2020.132. URL: https://hal.inria.fr/hal-02483114.

[30] R. Péchoux, S. Perdrix, M. Rennela and V. Zamdzhiev. 'Quantum Programming with Inductive Datatypes: Causality and Affine Type Theory'. In: International Conference on Foundations of Software Science and Computation Structures. Dublin, Ireland, 17th Apr. 2020, pp. 562–581. DOI: 10.1007/978-3-030-45231-5_29. URL: https://hal.inria.fr/hal-02995410.

[31] V. Zamdzhiev. 'Computational Adequacy for Substructural Lambda Calculi'. In: Applied Category Theory 2020. Virtual, United States, 6th July 2020. URL: https://hal.archives-ouvertes.fr/hal-03018433.

[32] V. Zamdzhiev. 'Semantics for First-Order Affine Inductive Data Types via Slice Categories'. In: International Workshop on Coalgebraic Methods in Computer Science. Virtual, France, 21st Sept. 2020. DOI: 10.1007/978-3-030-57201-3_10. URL: https://hal.archives-ouvertes.fr/hal-03018418.

Scientific book chapters

[33] O. Bournez, G. Dowek, R. Gilleron, S. Grigorieff, J.-Y. Marion, S. Perdrix and S. Tison. 'A Guided Tour of Artificial Intelligence Research - Volume III: Interfaces and Applications of Artificial IntelligenceTheoretical Computer Science: Computational Complexity'. In: *A Guided Tour of Artificial Intelligence Research - Volume III: Interfaces and Applications of Artificial IntelligenceTheoretical Computer Science: Computational Complexity (10.1007/978-3-030-06170-8)*. 2020. URL: https://hal.archives-ouvertes.fr/hal-02995771.

[34] M. Hoyrup. 'Algorithmic randomness and layerwise computability'. In: *Algorithmic Randomness – Progress and Prospects*. May 2020, p. 17. DOI: 10.1017/9781108781718. URL: https://hal.inria.fr/hal-02975222.

[35] M. Hoyrup and J. Rute. 'Computable Measure Theory and Algorithmic Randomness'. In: *Handbook of Computable Analysis*. 2020. URL: https://hal.inria.fr/hal-02938919.

[36] E. Jeandel and P. Vanier. 'The Undecidability of the Domino Problem'. In: *Substitution and Tiling Dynamics: Introduction to Self-inducing Structures*. Vol. 2273. 6th Dec. 2020, pp. 293–357. DOI: 10.1007/978-3-030-57666-0_6. URL: https://hal.inria.fr/hal-03087341.

Doctoral dissertations and habilitation theses

[37] R. Péchoux. 'Implicit Computational Complexity: past and future'. Université de Lorraine, 19th Oct. 2020. URL: https://hal.univ-lorraine.fr/tel-02978986.

Reports & preprints

[38] T. Carette and S. Perdrix. *Colored props for large scale graphical reasoning*. 10th Aug. 2020. URL: https://hal.archives-ouvertes.fr/hal-02904695.

[39] M. Hoyrup, T. Kihara and V. Selivanov. *Degree Spectra of Homeomorphism Types of Polish Spaces*. 27th Apr. 2020. URL: https://hal.inria.fr/hal-02555111.

[40] E. Jeandel and M. Rao. *An aperiodic set of 11 Wang tiles*. 18th Sept. 2020. URL: https://hal.inria.fr/hal-01166053.

[41] B. Lindenhovius, M. Mislove and V. Zamdzhiev. *LNL-FPC: The Linear/Non-linear Fixpoint Calculus*. 22nd Nov. 2020. URL: https://hal.archives-ouvertes.fr/hal-03018454.

[42] B. Lindenhovius, M. Mislove and V. Zamdzhiev. *Semantics for a Lambda Calculus for String Diagrams*. 22nd Nov. 2020. URL: https://hal.archives-ouvertes.fr/hal-03018467.

[43] R. Péchoux, S. Perdrix, M. Rennela and V. Zamdzhiev. *Quantum Programming with Inductive Datatypes*. 22nd Nov. 2020. URL: https://hal.archives-ouvertes.fr/hal-03018513.

[44] A. Saint-Jore, N. A. Fatès and E. Jeandel. *Amoebae for clustering: a bio-inspired cellular automata method for data classification*. 21st Oct. 2020. URL: https://hal.inria.fr/hal-02973830.

[45] R. Vilmart. *The Structure of Sum-Over-Paths, its Consequences, and Completeness for Clifford*. 29th May 2020. URL: https://hal.archives-ouvertes.fr/hal-02651473.

10.3 Cited publications

[46] A. Aspect, P. Grangier and G. Roger. 'Experimental tests of realistic local theories via Bell's theorem'. In: *Phys. Rev. Lett.* 47 (1981), p. 460.

[47] C. H. Bennett and G. Brassard. 'Quantum cryptography: Public key distribution and coin tossing'. In: *Proceedings of IEEE international Conference on Computers, Systems and Signal Processing, Bangalore, India*. New York: IEEE Press, 1984, p. 175.

[48] N. Fatès. 'Solving the decentralised gathering problem with a reaction-diffusion-chemotaxis scheme - Social amoebae as a source of inspiration'. In: *Swarm Intelligence* 4.2 (2010), pp. 91–115. URL: http://dx.doi.org/10.1007/s11721-010-0038-4.

[49] N. A. Fatès, I. Marcovici and S. Taati. 'Cellular automata for the self-stabilisation of colourings and tilings'. In: *Proceedings of RP 2019 (International Conference on Reachability Problems)*. Springer, Sept. 2019, pp. 121–136. DOI: 10.1007/978-3-030-30806-3_10. URL: https://hal.inria.fr/hal-02159155.

[50] N. Gauville, N. Fatès and I. Marcovici. 'Diagnostic décentralisé à l'aide d'automates cellulaires'. In: *JFSMA 2019 - 27emes Journées Francophones sur les Systèmes Multi-Agents*. ISBN 9782364937192. Institut de Recherche en informatique de Toulouse et l'Association française pour l'Intelligence Artificielle. Toulouse, France: Cépaduès, July 2019, p. 96–105. URL: https://hal.inria.fr/hal-02195799.

[51] Z. Iljazovic and I. Susic. 'Semicomputable manifolds in computable topological spaces'. In: *J. Complex.* 45 (2018), pp. 83–114. DOI: 10.1016/j.jco.2017.11.004. URL: https://doi.org/10.1016/j.jco.2017.11.004.

[52] A. Kissinger and J. van de Wetering. 'Pyzx: Large scale automated diagrammatic reasoning'. In: *arXiv preprint arXiv:1904.04735* (2019).

[53] B. Lindenhovius, M. Mislove and V. Zamdzhiev. 'Enriching a Linear/Non-linear Lambda Calculus: A Programming Language for String Diagrams'. In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*. Ed. by A. Dawar and E. Grädel. ACM, 2018, pp. 659–668. DOI: 10.1145/3209108.3209196. URL: https://doi.org/10.1145/3209108.3209196.

[54] B. Lindenhovius, M. Mislove and V. Zamdzhiev. 'Mixed Linear and Non-linear Recursive Types'. In: *Proc. ACM Program. Lang.* 3.ICFP (Aug. 2019), 111:1–111:29. DOI: 10.1145/3341715. URL: http://doi.acm.org/10.1145/3341715.

[55] J. S. Miller. 'Effectiveness for Embedded Spheres and Balls'. In: *Electronic Notes in Theoretical Computer Science* 66.1 (2002). CCA 2002, Computability and Complexity in Analysis (ICALP 2002 Satellite Workshop), pp. 127–138.

[56] K. Weihrauch. *Computable Analysis*. Springer, 2000.