INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Team CARTE

Theoretical Adverse Computations, and Safety

Nancy - Grand Est

THEME SYM

Activity Report

2007

Table of contents

1. **Team** **1**
2. **Overall Objectives** **1**
   2.1. Introduction 1
   2.2. Highlights of the year 2
3. **Scientific Foundations** **2**
   3.1. Introduction 2
   3.2. Continuous Computation Theory 2
   3.3. Rewriting 2
   3.4. Algorithmic Game Theory 3
   3.5. Computer Virology 3
4. **Application Domains** **4**
   4.1. Computations by Dynamical Systems 4
      4.1.1. Continuous time computation theories 4
      4.1.2. Analysis and verification of adversary systems 4
   4.2. Robust and Distributed Algorithms, Algorithmic Game Theory 5
   4.3. Computer Virology 6
      4.3.1. The theoretical track. 6
      4.3.2. The virus detection track. 6
      4.3.3. The virus protection track. 6
      4.3.4. The experimentation track. 6
5. **Software** **6**
   5.1. CARIBOO 6
   5.2. CROCUS 7
6. **New Results** **7**
   6.1. Analog Computations 7
   6.2. Analysis and verification of adversary systems 7
   6.3. Implicit Computational Complexity 8
   6.4. Models of dynamics in Networks 9
   6.5. Computer virology 9
7. **Other Grants and Activities** **10**
   7.1. Regional Actions 10
   7.2. National Actions 10
      7.2.1. Agence Nationale de la Recherche (ANR) project SOGEA 10
      7.2.2. Agence Nationale de la Recherche (ANR) project Virus 11
   7.3. European Actions 11
   7.4. Visits and invitations of researchers 11
8. **Dissemination** **11**
   8.1. Activities within the scientific community 11
   8.2. Workshop and conference organisation 12
   8.3. Program Committees 12
   8.4. Teaching 12
   8.5. Academic Supervision 12
   8.6. Thesis and admission committees 13
   8.7. Participation to colloquia, seminars, invitations 13
9. **Bibliography** **13**

1. Team

Head of project team
Jean-Yves Marion [Professor, École des Mines de Nancy, HdR]

Vice-head of project team
Olivier Bournez [CR INRIA, HdR]

Administrative assistant
Chantal Llorens

Permanent researchers
Johanne Cohen [CR CNRS]
Isabelle Gnaedig [CR INRIA]

Faculty members
Guillaume Bonfante [Assistant Professor, École des Mines de Nancy, en détachement INRIA]
Emmanuel Hainry

Ph. D. Students
Philippe Beaucamps [BDI CNRS-DGA (since October 1st), defense planned in 2010]
Octave Boussaton [Bourse Région]
Matthieu Kaczmarek [BDI CNRS-Région, defense planned in 2008]
Marco Gaboardi [co-tutelle avec U. Turin, defense Dec. 12th, 2007]
Romain Péchoux [MESR fellow, defense Nov. 14th, 2007]
Daniel Reynaud [BDI CNRS-Région (since October 1st), defense planned in 2010]

2. Overall Objectives

2.1. Introduction

Keywords: *Algorithmic*, *Algorithms*, *Complex Systems*, *Complexity*, *Computability*, *Computational Models*, *Formal Systems*, *Logic*, *Malwares*, *Networks*, *Program Properties*, *Resource Analysis*, *Termination*, *Virus*.

The aim of the CARTE research team is to take into account adversity in computations, which is implied by actors whose behaviors are unknown or unclear. We call this notion adversary computation.

The project combines two approaches, and we think that their combination will be fruitful. The first one is the analysis of the behavior of a wide-scale system, using tools coming from both Continuous Computation Theory and Game Theory. The second approach is to build defenses with tools coming rather from logic, rewriting and, more generally, from Programming Theory.

The activities of the CARTE team are organized around three research actions:

- Computations by Dynamical Systems
- Robust and Distributed Algorithms, Algorithmic Game Theory
- Computer Virology.

2.2. Highlights of the year

- Olivier Bournez won the prize of the Stanislas Académie.
- Publication of the book chapter *A Survey on Continuous Time Computation Theory*, [7].
- Publication of *Polynomial differential equations compute all real computable functions on computable compact intervals* which shows that analogical computational models have the same power than the digital ones, [8].
- A characterization of viruses based on Kleene's recursion theorems has been given, which provides a virus classification and gives rigorous scientific ground to computer virology.
- A first automatable technique has been proposed, to ensure that the mean length of computations is finite in the adversary context of non terminating probabilistic rewrite systems.
- Two papers related to game theory [13] and [2].
- Polygraphs have been recognized as a good model candidate for implicit complexity and sup-interpretation methods have demonstrate their abilities to determine resopurce upper bound.

3. Scientific Foundations

3.1. Introduction

We survey the different fields, which underline the scientific basis of CARTE, enhancing the aspects around adverse computations.

3.2. Continuous Computation Theory

Today's classical computability and complexity theory deals with discrete time and space models of computation. However, discrete time models of machines working on a continuous space have been considered: see e.g. *Blum Shub and Smale* machines [34], or recursive analysis [79]. Models of machines working with continuous time and space can also be considered: see e.g. the General Purpose Analog Computer of Claude Shannon [74].

Continuous models of computation lead to particular continuous dynamical systems. More generally, continuous time dynamical systems arise as soon as one attempts to model systems that evolve over a continuous space with a continuous time. They can even emerge as natural descriptions of discrete time or space systems. Utilizing continuous time systems is a common approach in fields such as biology, physics or chemistry, when a huge population of agents (molecules, individuals, ...) is abstracted into real quantities such as proportions or thermodynamic data.

Computation theory of continuous dynamical systems allow us to understand both the hardness of questions related to continuous dynamical systems and the computational power of continuous analog models of computations.

A survey on continuous-time computation theory, with discussions of relations between both approaches, co-authored by Olivier Bournez and Manuel Campagnolo, can be found in [37].

3.3. Rewriting

Rewriting has reached some maturity and the rewriting paradigm is now widely used for specifying, modelizing, programming and proving. It allows for easily expressing deduction systems in a declarative way, for expressing complex relations on infinite sets of states in a finite way, provided they are countable. Programming languages and environments have been developed, which have a rewriting based semantics. Let us cite *ASF+SDF* [40], *Maude* [42], and *Tom* [67].

For basic rewriting, many techniques have been developed to prove properties of rewrite systems like confluence, completeness, consistency or various notions of termination. In a weaker proportion, proof methods have also been proposed for extensions of rewriting like equational extensions, consisting of rewriting modulo a set of axioms, conditional extensions where rules are applied under certain conditions only, typed extensions, where rules are applied only if there is a type correspondence between the rule and the term to be rewritten, and constrained extensions, where rules are enriched by formulas to be satisfied [33], [44], [77].

An interesting aspect of the rewriting paradigm is that it allows automatable or semi-automatable correctness proofs for systems or programs. Indeed, properties of rewriting systems as those cited above are translatable to the deduction systems or programs they formalize and the proof techniques may directly apply to them.

Another interesting aspect is that it allows characteristics or properties of the modelized systems to be expressed as equational theorems, often automatically provable using the rewriting mechanism itself or induction techniques based on completion [29]. Note that the rewriting and the completion mechanisms also enable transformation and simplification of formal systems or programs.

3.4. Algorithmic Game Theory

Game theory aims at discussing situations of competition between rational players [73]. After the seminal works of Emile Borel and John von Neumann, one key events was the publication in 1944 of the book [81] by John von Neumann and Oskar Morgenstern. Game theory then spent a long period in the doldrums. Much effort was devoted at that time towards the mathematics of two-person, zero-sum games.

For general games, the key concept of Nash equilibrium was proposed in the early 50s by John Nash in [70], but it was not until the early 70s that it was fully realized what a powerful tool Nash has provided in formulating this concept. This is now a central concept in economics, biology, sociology and psychology to discuss general situations of competition, as attested for example by several Nobel prizes of economics.

Algorithmic game theory differs from game theory by taking into account algorithmic and complexity aspects. Indeed, historically main developpments of classical game theory have been realized in a mathematical context, without true considerations on effectivness of constructions.

Game theory and algorithmic game theory have large domains of applications in theoretical computer science: it has been used to understand complexity of computing equilibria [64], the loss of performance due to individual behavior in distributed algorithmics [31], the design of incentive mechanisms [71], the problems related to the pricing of services in some protocols [45]...

3.5. Computer Virology

From an historical point of view, the first official virus appeared in 1983 on Vax-PDP 11. In the very same time, a series of papers was published which always remain a reference in computer virology: Thompson [78], Cohen [43] and Adleman [30].

The literature which explains and discusses about practical issues is quite extensive, see for example Ludwig's book [60] or Szor's one [76] and all web sites...But, we think that the best references are both books of Filiol [46] (English translation [47]) and [49]. However, there are only a few theoretical/scientific studies, which attempt to give a model of computer viruses.

A virus is essentially a self-replicating program inside an adversary environment. Self-replication has a solid background based on works on fixed point in λ-calculus and on studies of Von Neumann [80]. More precisely we establish in [35] that Kleene's second recursion theorem [59] is the cornerstone from which viruses and infection scenarios can be defined and classified. The bottom line of a virus is behavior is

1. A virus infect programs by modifying them
2. A virus copies itself and can mutate
3. Virus spreads throughout a system

The above scientific foundation justifies our position to use the word virus as a generic word for self-replicating malwares. (There are yet a difference. A malware has a payload, and virus may not have one.) For example, worms are an autonous self-replicating malwares and so fall into our definition. In fact, the current malware taxonomy (virus, worms, trojans, ...) is unclear and subject to debate.

4. Application Domains

4.1. Computations by Dynamical Systems

Keywords: *Analog Models of Computations*, *Complexity*, *Computability*.

4.1.1. Continuous time computation theories

Continuous time systems arise as soon as one attempts to model systems that evolve over a continuous space with a continuous time. They can even emerge as natural descriptions of discrete time or space systems. Utilizing continuous time systems is a common approach in fields such as biology, physics or chemistry, when a huge population of agents (molecules, individuals, ...) is abstracted into real quantities such as proportions or thermodynamic data [58], [69].

Understanding computation theories for continuous systems leads to understand hardness of verification and control of these systems. This has been used to discuss problems in fields as diverse as verification (see e.g. [32]), control theory (see e.g. [41]), VLSI design (see e.g. [65]), neural networks (see e.g. [72])

Some systems carrying out computations are intrinsically continuous. This is the case, for example, of some more or less futuristic models such as quantum models of computations, or models based on space-time curves, but also of some mechanical [74] or electronic [36] machines.

Understanding computation theories for continuous systems leads, in this context, to characterize the computational power of these machines. This has been used to discuss, for example, the power of models such as the General Purpose Analog Computer of Shannon [74], Formal Neural Network models [72].

Since continuous time systems are conducive to modeling huge populations, one might speculate that they will have a prominent role in analyzing massively parallel systems such as the Internet. This is one deep motivation of our work on computation theories for continuous systems.

4.1.2. Analysis and verification of adversary systems

The other research direction on dynamical systems we are interested in is the study of properties of adversary systems or programs, i.e. of systems whose behavior is unknown or indistinct, or which do not have classical expected properties. We would like to offer proof and verification tools, to guarantee the correctness of such systems.

One one hand, we are interested in continuous and hybrid systems. In a mathematical sense, a hybrid system can be seen as a dynamical system, whose transition function does not satisfy the classical regularity hypotheses, like continuity, or continuity of its derivative. The properties to be verified are often expressed as reachability properties. For example, a safety property is often equivalent to (non-)reachability of a subset of unsure states from an initial configuration, or to stability (with its numerous variants like asymptotic stability, local stability, mortality, etc ...). Thus we will essentially focus on verification of these properties in various classes of dynamical systems.

We are also interested by rewriting techniques, used to describe dynamic systems, in particular in the adversary context. As they were initially developed in the context of automated deduction, the rewriting proof techniques, although now numerous, are not yet adapted to the complex framework of modelization and programming. An important stake in the domain is then to enrich them to provide realistic validation tools, both in providing finer rewriting formalisms and their associated proof techniques, and in developing new validation concepts in the adversary case, i.e. when usual properties of the systems like, for example, termination are not verified.

For more than five years, we have been developing specific procedures for property proofs of rewriting, for the sake of programming, in particular with inductive techniques, already applied with success to termination under strategies [51], [52], [53], to weak termination [54] and to sufficient completeness [57].

The last two results take place in the context of adversary computations, since they allow for proving that even a divergent program, in the sense where it does not terminate, can give the expected results.

We especially intend to continue to study the termination property, and its impact on other properties, since it seems to be central in several research directions of the project:

- Until now it was very used for deciding properties like confluence, completeness or consistency. Its weakening or its absence, very frequent in real cases, throws us into the adversary context.
- It is a tool for the analysis of resources.
- Thoroughly analyzing divergence by non-termination can be an approach for modeling the continuous.

A crucial element of safety and security of software systems is the problem of resources. We are working in the field of Implicit Computational Complexity. Interpretation based methods like Quasi-interpretations (QI) or sup-interpretations, are the approach we have been developing these last five years, see [61], [62], [63]. Implicit complexity is an approach to the analysis of the resources that are used by a program. Its tools come essentially from proof theory. The aim is to compile a program while certifying its complexity.

4.2. Robust and Distributed Algorithms, Algorithmic Game Theory

Keywords: *Algorithmic Game Theory*, *Distributed Systems*, *Models for Dynamics*.

One of the problems related to distributed algorithmics corresponds to the minimization of resources (time of transit, quality of services) in problems of transiting information (routing problems, group telecommunications) in telecommunication networks.

Each type of network gives rise to natural constraints on models. For example, a network is generally modeled by a graph. The material and physical constraints on each component of the network (routers, communication media, topology, etc ...) result in different models. One natural objective is then to build algorithms to solve those types of problems on various models. One can also constraint solutions to offer certain guarantees: for example the property of self-stabilization, which expresses that the system must end in a correct state whatever its initial state is; or certain guarantees of robustness: even in the presence of a small proportion of Byzantine actors, the final result will remain correct; even in the presence of rational actors with divergent interests, the final result will remain acceptable.

Algorithms of traditional distributed algorithmics were designed with the strong assumption that the interest of each actor does not differ from the interest of the group. For example, in a routing problem, classical distributed algorithms do not take into account the economic interests of the various autonomous systems, and only try to minimize criteria such as shortest distances, completely ignoring the economical consequences of decisions for involved agents.

If one wants to have more realistic models, and take into account the way the different agents behave, one gets more complex models.

However, today, one gets models which are hard to analyse. For example,

- Models of dynamism are missing: e.g., how to model a negotiation in a distributed auction mechanism for the access to a telecommunications service,
- only few methods are known to guarantee that the equilibrium reached by such systems remains in some domains that could be qualified as safe or reasonable,
- there is almost no method discussing the speed of convergence, when there is convergence,
- only a little is known about the time and space resources necessary to establish some techniques to guarantee correct behavior.

Thus, it is important to reconsider the algorithms of the theory of distributed algorithmics, under the angle of the competitive interests that involved agents can have (Adversary computation). This requires to include/understand well how to reason on these types of models.

4.3. Computer Virology

Keywords: *Virus*, *attack*, *defense*, *formal grammars*, *propagation*, *recursion theory*, *self-replication*, *worms*.

Nowadays, our thoughts lead us to define four different research tracks, that we are describing below.

4.3.1. The theoretical track.

It is rightful to wonder why there is only a few fundamental studies on computer viruses while it is one of the important flaws in software engineering. The lack of theoretical studies explains maybe the weakness in the anticipation of computer diseases and the difficulty to improve defenses. For these reasons, we do think that it is worth exploring fundamental aspects, and in particular self-reproducing behaviors.

4.3.2. The virus detection track.

The crucial question is how to detect viruses or self-replicating malwares. Cohen demonstrated that this question is undecidable. The anti-virus heuristics are based on two methods. The first one consists in searching for virus signatures. A signature is a regular expression, which identifies a family of viruses. There are obvious defect. For example, an unknown virus will not be detected, like ones related to a 0-day exploit. We strongly suggest to have a look at the independent audit [48] in order to understand the limits of this method. The second one consists in analysing the behavior of a program by monitoring it. Following [50], this kind of methods is not yet really implemented. Moreover, the large number of false-positive implies this is barely usable. To end this short survey, intrusion detection encompasses virus detection. However, unlike computer virology, which has a solid scientific foundation as we have seen, the IDS notion of “malwares” with respect to some security policy is not well defined. The interesting reader may consult [68].

4.3.3. The virus protection track.

The aim is to define security policies in order to prevent malaware propagation. For this, we need to (i) to define what is a computer in different programming languages and setting, (ii) to take into consideration resources like time and space. We think that formal methods like rewriting, type theory, logic, or formal languages, should help to define the notion of a *formal immune system*, which defines a certified protection.

4.3.4. The experimentation track.

This study on computer virology leads us to propose and construct a “high security lab” in which experiments can be done in respect with the French law. This project of “high security lab” in one of the main project of the CPER 2007-2013.

5. Software

5.1. CARIBOO

Keywords: *Abstraction*, *Induction*, *Narrowing*, *Strategy*, *Termination*.

Participant: Isabelle Gnaedig [correspondant].

In the context of our study of rule-based program proof and validation, we develop and distribute CARIBOO (http://protheo.loria.fr/softwares/cariboo/), an environment dedicated to specific termination proofs under strategies like the innermost, the outermost or local strategies.

Written in Elan and Java, it has a reflexive aspect, since Elan is itself a rule-based language. CARIBOO was partially developed in the Toundra QSL project, and reinforced in the framework of the Modulogic ACI [55], [56].

5.2. CROCUS

Keywords: *Implicit computational complexity*, *Polynomial inequation*, *Resource analysis*.

Participants: Guillaume Bonfante [correspondant], Jean-Yves Marion, Romain Péchoux.

The CROCUS software aims at synthesizing quasi-interpretations. It takes programs as input and returns the corresponding quasi-interpretation. Doing this, it can guarantee some bounds on the memory used along computations by the input program. The currently analyzed programs are written in a subset of the CAML language, more precisely a first-order functional language subset of CAML.

6. New Results

6.1. Analog Computations

Keywords: *Analog Computations*, *Complexity*, *Computability*.

Participants: Olivier Bournez, Daniel Graça, Emmanuel Hainry.

There are many ways to model analog computers. Unifying those models is therefore a crucial matter as opposed to the discrete case, as there is no property stating that those models are equivalent. It is possible to distinguish two groups in these models: on one side, continuous time models; on the other side, discrete-time models working on continuous structures as a model derived from Turing machines. The first group contains in particular some sets of functions defined by Moore in [66]. The main representative of the second group are the real computable functions and a subclass of this set: the set of elementary computable functions.

There are few comparisons between classes of functions from the first group and from the second group, and in particular, there were almost no result of equality.

It is known that Euler's Gamma function is computable according to computable analysis, while it cannot be generated by Shannon's General Purpose Analog Computer (GPAC). This example has often been used in order to argue that the GPAC is less powerful than digital computations.

However, as we demonstrated in [8], when computability with GPACs is considered in the framework of recursive analysis, we obtain two equivalent models of analog computation. Since GPACs are equivalent to systems of polynomial differential equations then we show that all real computable functions can be defined by such models.

In [7], in collaboration with Manuel Campagnolo in Lisbon, we wrote a survey on continuous time models of computations. In this paper, we provide an overview of computation theories of continuous time computation. These theories allow us to understand both the hardness of questions related to continuous time dynamical systems and the computational power of continuous time analog models. We survey the existing models, summarizing results, and point to relevant references in the literature.

In [21], we reviewed some results about the computational power of several computational models. Considered models have in common to be related to continuous dynamical systems. We described the relations between polynomial Cauchy problems, GPAC-generable and GPAC-computable functions. We recalled planar mechanisms. We presented population protocols, and some models of computations inspired by models of dynamics in game theory.

6.2. Analysis and verification of adversary systems

Keywords: *Abstraction*, *Completeness*, *Induction*, *Narrowing*, *Rewriting*, *Strategy*, *Termination*.

Participant: Isabelle Gnaedig.

In previous works, we had proposed a method for specifically proving, by explicit induction, termination of rewriting under strategies. The proof principle consists, for a given term rewriting system, in establishing on the ground term algebra that every term terminates i.e. has only finite derivations, assuming that any smaller term terminates. For that, we develop proof trees representing the possible derivations from any term using the term rewriting system. Two mechanisms are used, namely abstraction, introducing variables that represent ground terms in normal form and schematize normalization of subterms, and narrowing, schematizing rewriting in different ways according to the possible ground instances of the terms studied. These two mechanisms deal with constraints used to ensure the inductive process, and to control the growth of the proof trees.

From three distinct procedures, initially given for respectively proving termination of rewriting with the innermost, the outermost and the local strategies, factorizing the common mechanisms, we have obtained a generic algorithm, combining three proof steps, that exactly correspond to the basic principles of the induction mechanism: an abstraction, a narrowing and a stop step. We have then studied the ordering constraints generated by our inductive proof principle, in order to automate the proof algorithm for the different strategies. This year, we have completed our results, giving a completeness theorem for our procedure, and discussing the limits of abstraction and narrowing to handle strategies, which finishes an article to appear in [10].

Inspired by the applicability of our inductive approach for other properties than termination [54], [57], going one step further, we have extended it to the proof of a property P of a reduction relation on a $\mathcal{F}$-algebra. Generalizing the notion of normal form when P is termination, and of constructor form when P is completeness, we have defined the notion of P-canonical form. Our abstraction mechanism then consists in replacing subterms by P-canonical forms. We have formalized the notion of simulation of a reduction relation $\rightarrow$ by another one, in general a narrowing relation, to prove P. The induction mechanism then works on two distinct sets: a basis set containing the irreducible elements for $\rightarrow$, and a set of patterns for which proof trees are developed, simulating rewriting trees. We finally showed how this mechanism is instantiated to express the proof procedures already given for termination under strategies, and weak properties like weak termination or completeness [23].

The probabilistic framework, recently proposed in [39], is interesting to modelize uncertain phenomenons, and to capture properties like termination, which are not verified in the general case. We then have studied whether our inductive approach could tackle the termination problem of the above probabilistic rewriting relation. We have showed that the mechanism of probabilistic rewriting can be simulated, like the non-probabilistic one, by abstraction and narrowing, providing a probabilistic lifting lemma. We have extended our technique to prove the property of positive almost sure termination [38], expressing that the mean length of the rewriting chains starting from any term is finite. For that, we have generalized our method to infinite proof trees, under some syntactic conditions on the probabilistic rewrite rules. For rewrite systems on constants, very interesting to modelize probabilistic protocols, our procedure is completely automatable [22].

6.3. Implicit Computational Complexity

Keywords: *complexity*, *light linear logic*, *quasi-interpretation*, *ramification*, *resource upper bound*, *sup-interpretation*, *tiering*.

Participants: Jean-Yves Marion, Guillaume Bonfante, Romain Péchoux.

The goal of implicit computational complexity is to give ontogenetic model of computational complexity. We follow two lines of research. The first is related to the initial ramified recursion theory due to Leivant and Marion. The second is related to interpretation methods, quasi-interpretation and sup-interpretation, in order to provide an upper bound on some computational resources, which are necessary for a program execution. This approach seems to have some practical interests, and we develop a software Crocus that automatically infer complexity upper bounds of functional programs.

In [24], Jean-Yves Marion came back to ramified recursion theory. Ackemann's construction of a non-primitive function is based on a enumeration level by level of all primitive recursions. Then, a diagonalization argument provides the Ackemann's function, see Simmons's book [75] for a modern explanation. Following the same idea, we construct an hierarchy of ramified functions , which characterized exactly the set of functions whose runtime is bounded by $O(n^k)$ (for each k, where n is the input size, over a RAM type of machine). Next, by a diagonalization argument, we obtain a function, which jumps outside polynomial time. The whole system is presented as a lambda-calculus with dependent type.

We pursue our investigation on quasi-interpretations and sup-interpretations. In [20] Guillaume Bonfante, Jean-Yves Marion and Romain Péchoux has studied various heuristic to compute quasi-interpretations by decomposing first order functional programs. There are at least three consequences of this study. The first is the design of better algorithms to determine quasi-interpretation. The second is the ability to find quasi-interpretations to more programs than the traditional approach. The last consequence is the application of quasi-interpretations to determine complexity of higher order functional programs. In [25], Jean-Yves Marion and Romain Péchoux has extended the notion of sup-interpretations to a fragment of Java.

On a more theoretical side, Guillaume Bonfante and Yves Guiraud began the study of polygraphs from their intensional properties. Compared to the usual rewriting framework, polygraphs intrinsically allows functions with more than one output. They showed in a first step that polygraphs are well suited to divide and conquer programs [14]. Actually, they give a characterization of PTIME in [4]. An important observation — which will be largely used in short terms — is that in this work, Bonfante and Guiraud managed to treat time and space on different levels.

6.4. Models of dynamics in Networks

Keywords: *algorithmic game theory*, *dynamics*, *graphs*, *networks*, *telecommunications*.

Participants: Olivier Bournez, Johanne Cohen, Octave Boussaton.

We considered a model of interdomain routing proposed by a partner from SOGEA project. We proved that the model has no pure Nash equilibria, even for 4 nodes. Proof of convergence of the fictious player dynamics for the corresponding network has been established for some specific cases.

We reviewed the different models of dynamism in literature in game theory, in particular models from evolutionary game theory. We presented some ways to use them to realize distributed computations in [21]. Considered models are particular continuous time models, and hence are also covered by the survey [7].

Octave Boussaton is currently working on the theory of learning equilibria, in particular in Wardrop routing networks. The proof of the convergence of a specific learning strategy has been established for some networks. The result has been presented in [12], and should soon lead to a submission.

erover, we presented both a game theoretic and a distributed algorithmic approach for the transit price negotiation problem in the interdomain routing framework. We analyzed the behavior of providers on a specific scenario, mainly by considering the simple but not simplistic case of one source and one destination. The analysis of the centralized transit price negotiation problem shows that the only one non cooperative equilibrium is when the lowest cost provider takes all the market. The perspective of the game being repeated makes cooperation possible while maintaining higher prices. Then, we considered the system under a distributed framework. Indeed, in reality the nodes only have a local view of the game including the topology and thus the nature and the length of the possible routes. We simulated the behavior of the distributed system under a simple price adjustment strategy and analyzed whether it matches the theoretical results or not. This work is published in [13].

6.5. Computer virology

Keywords: *Malware*, *art of war*, *detection*, *learning*, *self-reproduction*.

Participants: Guillaume Bonfante, Matthieu Kaczmarek, Jean-Yves Marion.

We work on computer virology since 2005. Our research has led us to develop three subjects on computer virology.

The first subject is about the fundamental aspect of computer virology. As we have said previously, computer virology leans on self-reproduction system theory and in particular on Kleene's recursion Theorem. In particular, we obtained a classification of viruses based on the strenghnens of the recursion theorem which is necessary to use in order to construct a virus. Following our previous study, Guillaume Bonfante, Matthieu Kaczmarek and Jean-Yves Marion in [15] have showed that those, a priori theoretical construction, might be quite easily implemented in an imperative programming language. Moreover, they construct various highly polymorphic virus engines. This is a part of the Theoretical track.

The second subject concerns the defense by new malware detection methods. We have presented in [17] and in [18] a method to detect malware based on the static analysis of program data flo graphs. We dub this approach *morphological detection*. We try to show that it is possible to recognize a malware by identifying its data flow graphs. We are currently trying to experiment our ideas and to validate it. For this, we use various tree automata techniques. This is a part of the detection track.

The goal of the last subject is to understand the impact of computer virology in the society and to explain it. Anne Bonfante and Jean-Yves Marion, in [5], discussed the notion of defense in computer virology with respect to the traditional studies on the art of war. In passing, the recent news in September and October 2007 have shown that this kind of study may shed some lights on the relations between computer security and society issues. ("Les Etats-unis, l'Allemage, la France et la Nouvelle-Zélande ont annoncé avoir subi des cyber-attaques provenant de la Chine", Le Monde 05/10/07). It is worth pointing out the huge difficulty to lead a real pluri-disciplinary research despite the official discourse. Jean-Yves Marion with Eric Filiol, in [9] has tried to explain in simple term what is computer virology.

Finally, it is worth to mention that Guillaume Bonfante and Jean-Yves Marion was guest editors of the Journal of Computer Virology, Springer. See [6]. We have organized the second international workshop on Theory of computer viruses (TCV) at Nancy.

And because there is no special drawer to put this result, Jérôme Besombes and Jean-Yves Marion have published in [3] an algorithms to infer regular tree languages from positive examples and membership queries.

7. Other Grants and Activities

7.1. Regional Actions

- CARTE is part of the "Sécurité et Sûreté des Système (SSS)" theme of the "contrat de plan État-Région". Olivier Bournez is the head of the research operation TATA. Jean-Yves Marion is the co-head of the research operation LHS.
- CARTE is part of the "Qualité et sûreté des Logiciels (QSL)" theme of the "contrat de plan État-Région". Jean-Yves Marion is head of the QSL theme; Guillaume Bonfante is the head of the research operation VVV on viruses, and Johanne Cohen is the head of the operation Tagada.

  The web page of this action is http://qsl.loria.fr.

7.2. National Actions

7.2.1. Agence Nationale de la Recherche (ANR) project SOGEA

The three-year ARA Sécurité, Systèmes embarqués et Intelligence ambiante SOGEA began on September 2006. It deals with models of dynamics in algorithmic game theory, and applications in networks. It involves some members of CARTE, and some members of the group Parallelism and some members of the group Complexity in Orsay, as well as some members of PRISM Laboratory in Versailles. The head of the project is Olivier Bournez.

7.2.2. *Agence Nationale de la Recherche (ANR) project Virus*

The three-year "Action de Recherche Amont" "Sécurité, Systèmes embarqués et Intelligence ambiante" "Virus" began on September 2006. It deals with the theory of computer viruses. It involves some members of CARTE and the group of Éric Filiol at the ESAT-Rennes, a group leader in computer virology. The head of the Project is Jean-Yves Marion.

7.3. European Actions

- Some members of the project are involved in Computatibiliy in Europe Network. The head of the French Node is Olivier Bournez.

7.4. Visits and invitations of researchers

- Guillaume Bonfante has been invited as regular visitor for two months (June-July) by the CMAF (Centro de Matemàtica e Applicações Fundamentais) in Lisbon.
- Kerry Ojakian was invited to work one week in our project by Olivier Bournez and Emmanuel Hainry.

8. Dissemination

8.1. Activities within the scientific community

- Guillaume Bonfante: member of the engineering part of the Comipers hiring committee at LORIA.
- Olivier Bournez:
  - award winner of the Prix Scientifique 2007 de l'Académie Stanislas,
  - member of the board of directors of AFIT, the french chapter of the EATCS,
  - elected member (SGEN-CFDT) of the board of directors of INRIA,
  - member (titulaire since 2006) of the hiring committee (CS) at the University of Metz, section 27, since September 2004,
  - member (titulaire since 2007) of the hiring committee (CS) at INPL, section 27, since September 2004,
  - member (suppléant since 2004) of the hiring committee (CS) at INPL, section 27, since September 2001,
  - member of the "Comité Espace Transfert" of LORIA,
  - member of the list of experts of ANR and of FP7.
- Johanne Cohen:
  - member of the hiring committee (CS) at INPL, section 27, since September 2003,
  - member of the hiring committee (CS) at University of Nancy I, section 27.
- Jean-Yves Marion:
  - member of the steering committee of the International workshop on Logic and Computational Complexity (LCC/ICC),
  - member of the hiring committee (CS) at the University of Metz, section 27, since September 2004,
  - member of the hiring committee at INPL (Professors and Lecturers), section 27, since June 2004, and President since 2007.

  - elected to the scientific council of INPL in July 2003 and member of the board,
  - expert for DSPT 9, and of the PEDR committe.
  - expert for the Austrian Science Fundation,
  - member of the board of GIS 3SGS,
  - member of the "équipe de direction du Loria".
  - member of the invited professor part of the Comipers hiring committee at LORIA, since 2007.

8.2. Workshop and conference organisation

- Olivier Bournez:
  - co-organized with Pierrick Gaudry, and Natacha Portier the *Ecole Jeunes Chercheurs en Algorithmique et Calcul Formel*, from March 18th to March 24th in Nancy,
  - co-organized with Didier Galmiche the session 2007 of the annual *Complexité Modèles Finis et Bases de Données* workshop, from May 21st to May 22th in Nancy,
  - co-organized with Paola Bonizzoni a session *"Logic and New Computational Paradigms"* at conference Computability in Europe 2007, from June 18th to June 23th in Siena,
  - organized a session *Computational Models over the Reals* at the "Journées Arithmétiques du GDR Informatique Mathématique" days, in Montpellier, on January 24th.
- Guillaume Bonfante and Matthieu Kaczmarek co-orgnized TCV'07 at Loria in May 07.

8.3. Program Committees

- Olivier Bournez: program committee of RP'2007.
- Isabelle Gnaedig: program committee of the Ninth International Workshop on Termination (WST 2007).
- Jean-Yves Marion: guest editor of the Transaction on computational logic (TOCL-ACM), program committees of MCU'07 and WOLLIC'07.
- Guillaume Bonfante and Jean-Yves Marion: program comittee of the workshop Theory of computer viruses (TCV'07).

8.4. Teaching

- Olivier Bournez teached "Algorithmic and Complexity" in Master 1st year of Nancy I University, and "Algorithmic Verification" in Master 2nd year of Nancy I University.
- Johanne Cohen teached "Distributed Algorithmic " in Master 1st year of Nancy I University, and in 2nd year of ESIAL. She also teached "Algorithmique des télécommunications" in master 2nd year of Versailles University.
- Isabelle Gnaedig is coordinator of the course on "Program and Software Specification" at ESIAL, 3rd year. In this context, she also gave courses and supervised practical works on "Algebraic Specifications, the LOTOS Language and the Semantics of Concurrent Processes".
- Jean-Yves Marion is in charge of the option "Ingénierie des systèmes informatiques" at École des Mines de Nancy and teaches, a full service, in second and third year of ENSMN.

8.5. Academic Supervision

- Florent Garnier, co-supervised by Olivier Bournez, defended his PhD on September 17th.

- Johanne Cohen is co-supervising the thesis work of Octave Boussaton with Dominique Barth (University of Versailles).
- Jean-Yves Marion and Eric Filiol (ESAT) is supervising the thesis work of Philippe Beaucamps from November 2007.
- Jean-Yves Marion and Eric Filiol (ESAT) are supervising the thesis work of Daniel Reynaud from November 2007.
- Jean-Yves Marion is supervising the thesis work of Romain Péchoux from September 2004. The defense November 14th, 2007.
- Jean-Yves Marion and Simona Ronchi Della Rocca (Torino university) are co-supervising the thesis work of Marco Gaboardi. The defense December 12th, 2007.
- Jean-Yves Marion and Guillaume Bonfante are co-supervising the thesis work of Matthieu Kaczmarek from September 2005.

8.6. Thesis and admission committees

- Olivier Bournez: referee for the PhD of Daniel Graça, in Portugal, in September 2007.
- Isabelle Gnaedig: ESIAL admission committee.
- Jean-Yves Marion: referee for the Phd of F. Dabrowski, June 2007, and exmaminer for the Phd of C. Riba, Dec 2007.

8.7. Participation to colloquia, seminars, invitations

- Guillaume Bonfante:
  - invited at the University of Innsbruck (May 21-25th),
  - invited for a seminar at the University of Coimbra (June 13th),
  - invited for a seminar at the CMAF (Lisbon) (June 21th),
  - invited for a seminar at the University of Lisbon (July 24th).
- Olivier Bournez:
  - invited speaker at MCU'2007 (September 10th),
  - invited for a seminar at ENS Paris (September 18th) ,
  - invited for a seminar at LIRMM (January 11th),
  - invited for a seminar at "Journées Interdisciplinaires Informatique, Economie et Mathématique. Autour des réseaux de Télécommunication", at Sorbonne University, Paris (September 28th)
- Johanne Cohen: invited for a seminar "Optimal Linear Arrangement of Interval Graphs" at University of Mulhouse.
- Jean-Yves Marion:
  - invited at Turin, December 11th-13th, 2007,
  - invited at Turin, June 4th-June 9th, 2007,
  - invited for a seminar at Paris 13 (Feb. 12th), at CMFBD at Loria (May 21th), and at Paris 7 (Nov. 6).

9. Bibliography

Year Publications

Articles in refereed journals and book chapters

[1] D. BARTH, J. COHEN, A. DENISE, R. RIVIÈRE. *Shuffling biological sequences with motif constraints.*, in "Journal of Discrete Algorithms", A paraitre, accès en ligne depuis le 9 juin 2007, 2007, http://hal.inria.fr/inria-00187827/en/.

[2] D. BARTH, J. COHEN, T. FAIK. *On the b-continuity property of graphs*, in "Discrete Applied Mathematics", vol. 155, 2007, p. 1761-1768, http://hal.inria.fr/inria-00181941/en/.

[3] J. BESOMBES, J.-Y. MARION. *Learning tree languages from positive examples and membership queries*, in "Theoretical Computer Science", vol. 382, n^{o} 3, 09 2007, p. 183-197, http://hal.archives-ouvertes.fr/hal-00175415/en/.

[4] G. BONFANTE, Y. GUIRAUD. *Polygraphic programs and polynomial-time functions*, in "Logical Methods in Computer Science (LMCS)", 38 pages, under revision, 2007, http://hal.inria.fr/inria-00122932/en/.

[5] A. BONFANTE, J.-Y. MARION. *On the defence notion*, in "Journal in Computer Virology", vol. 3, n^{o} 4, 11 2007, p. 247-251, http://hal.archives-ouvertes.fr/hal-00184985/en/.

[6] G. BONFANTE, J.-Y. MARION. *Foreword of WTCV 06*, in "Journal in Computer Virology", vol. 3, n^{o} 1, 03 2007, p. 1-2, http://hal.archives-ouvertes.fr/hal-00184987/en/.

[7] O. BOURNEZ, M. L. CAMPAGNOLO. *A Survey on Continuous Time Computation*, in "New Computational Paradigms", B. COOPER (editor), Springer, 2007, http://hal.inria.fr/inria-00102948/en/.

[8] O. BOURNEZ, M. L. CAMPAGNOLO, D. S. GRAÇA, E. HAINRY. *Polynomial differential equation compute all real computable functions*, in "Journal of Complexity", vol. 23, 2007, p. 317-335, http://hal.inria.fr/inria-00102947/en/.

[9] E. FILIOL, J.-Y. MARION. *La virologie informatique - les virus, ennemis utiles*, in "Pour la Science", n^{o} 55, 04 2007, p. 60-64, http://hal.archives-ouvertes.fr/hal-00184983/en/.

[10] I. GNAEDIG, H. KIRCHNER. *Termination of Rewriting under Strategies*, in "ACM Transactions on Computational Logic", To appear, 2007, http://hal.inria.fr/inria-00182432/en/.

[11] M. KACZMAREK. *ELF et virologie informatique*, in "GNU Linux Magasine France", 2007, p. 12–19, http://hal.inria.fr/inria-00176238/en/.

Publications in Conferences and Workshops

[12] D. BARTH, O. BOURNEZ, O. BOUSSATON, J. COHEN. *Convergences et dynamiques du routage dans les réseaux*, in "Journées Pôle ResCom, Toulouse France", 2007, http://hal.inria.fr/inria-00182739/en/.

[13] D. BARTH, J. COHEN, L. ECHABBI, C. HAMLAOUI. *Transit prices negotiation: Combined repeated game and distributed algorithmic approach*, in "First EuroFGI International Conference on Network Control and Optimization - NET-COOP 2007 Network Control and Optimization First EuroFGI International Conference, NET-COOP 2007, Avignon, France, June 5-7, 2007. Proceedings Lecture Notes in Computer Science, Avignon France", T. CHAHED, B. TUFFIN (editors), Lecture Notes in Computer Science, ISBN 978-3-540-72708-8, vol. 4465, Springer Berlin / Heidelberg, 2007, p. 266-275, http://hal.inria.fr/inria-00180914/en/.

[14] G. BONFANTE, Y. GUIRAUD. *Intensional properties of polygraphs*, in "4th International Workshop on Computing with Terms and Graphs - TERMGRAPH 2007 Electronic Notes in Theoretical Computer Science, Braga Portugal", Electronic Notes in Theoretical Computer Science, 12 pages, to appear, 2007, http://hal.inria.fr/inria-00129391/en/.

[15] G. BONFANTE, M. KACZMAREK, J.-Y. MARION. *A Classification of Viruses through Recursion Theorems*, in "Computability in Europe Computation and Logic in the Real World, Third Conference on Computability in Europe Lecture Notes in Computer Science, Sienna Italie", S. B. COOPER, B. LÖWE, A. SORBI (editors), Lecture Notes in Computer Science, The original publication is available at www.springerlink.com ; ISBN 978-3-540-73000-2 ; ISSN 0302-9743 (Print) 1611-3349 (Online), vol. 4497, Springer, 2007, p. 73-82, http://hal.inria.fr/inria-00175301/en/.

[16] G. BONFANTE, M. KACZMAREK, J.-Y. MARION. *A Classification of Viruses through Recursion Theorems (Preliminary version)*, in "International Workshop on the Theory of Computer Viruses TCV'07, Nancy France", E. FILIOL, J.-Y. MARION, G. BONFANTE (editors), 2007, http://hal.inria.fr/inria-00176236/en/.

[17] G. BONFANTE, M. KACZMAREK, J.-Y. MARION. *Control Flow Graphs as Malware Signatures*, in "International Workshop on the Theory of Computer Viruses TCV'07, Nancy France", E. FILIOL, J.-Y. MARION, G. BONFANTE (editors), 2007, http://hal.inria.fr/inria-00176235/en/.

[18] G. BONFANTE, M. KACZMAREK, J.-Y. MARION. *Control Flow to Detect Malware*, in "Inter-Regional Workshop on Rigorous System Development and Analysis 2007, Nancy France", 2007, http://hal.inria.fr/inria-00176241/en/.

[19] G. BONFANTE, J. LE ROUX. *Intersection Optimization is NP-Complete*, in "Sixth International Workshop on Finite-State Methods and Natural Language Processing - FSMNLP 2007, Postdam Allemagne", 2007, http://hal.inria.fr/inria-00185232/en/.

[20] G. BONFANTE, J.-Y. MARION, R. PÉCHOUX. *Quasi-interpretation Synthesis by Decomposition : An application to higher-order programs*, in "ICTAC, Macao Chine", 2007, http://hal.inria.fr/inria-00130920/en/.

[21] O. BOURNEZ, E. HAINRY. *On the Computational Capabilities of Several Models*, in "5th International Conference on Machines, Computations and Universality - MCU 2007 Machines, Computations and Universality 5th International Conference, MCU 2007, Orléans, France, September 10-13, 2007. Proceedings Lecture Notes in Computer Science, Orléans France", J. DURAND-LOSE, M. MARGENSTERN (editors), Lecture Notes in Computer Science, ISBN 978-3-540-74592-1, vol. 4664, Springer Berlin / Heidelberg, 2007, p. 12-23, http://hal.inria.fr/inria-00182738/en/.

[22] I. GNAEDIG. *Induction for Positive Almost Sure Termination*, in "Proceedings of the 9th ACM-SIGPLAN International Symposium on Principles and Practice of Declarative Programming - PPDP 2007, Wroclaw, Pologne", ACM, 2007, p. 167-177, http://hal.inria.fr/inria-00182435/en/.

[23] I. GNAEDIG, H. KIRCHNER. *Narrowing, Abstraction and Constraints for Proving Properties of Reduction Relations*, in "Rewriting, Computation and Proof - Essays Dedicated to Jean-Pierre Jouannaud on the Occasion of His 60th Birthday, Paris, France", H. COMON, C. KIRCHNER, H. KIRCHNER (editors), Lecture Notes in Computer Science, vol. 4600, Springer, 2007, p. 44-67, http://hal.inria.fr/inria-00182434/en/.

[24] J.-Y. MARION. *Predicative Analysis of Feasibility and Diagonalization*, in "Typed Lambda Calculi and Applications, 8th International Conference - TLCA 2007, Paris France", S. R. D. ROCCA (editor), vol. 4583, Springer, 06 2007, p. 290-304, http://hal.archives-ouvertes.fr/hal-00175412/en/.

[25] J.-Y. MARION, R. PÉCHOUX. *Resource control of object-oriented programs*, in "The International Workshop on Logic and Computational Complexity - LCC 2007, Wroclaw/Poland", 07 2007, http://hal.inria.fr/inria-00155288/en/.

Internal Reports

[26] G. BONFANTE. *Monotone interpretations*, Research Report, 2007, http://hal.inria.fr/inria-00145060/en/.

[27] I. GNAEDIG. *Induction for Positive Almost Sure Termination - Extended version*, Research Report, 2007, http://hal.inria.fr/inria-00147450/en/.

[28] J.-Y. MARION, R. PÉCHOUX. *A characterization of polynomial complexity classes using dependency pairs*, Research Report, 2007, http://hal.inria.fr/inria-00155287/en/.

References in notes

[29] A. ROBINSON, A. VORONKOV (editors). *Handbook of Automated Reasoning*, vol. 1, North-Holland, 2001.

[30] L. ADLEMAN. *An Abstract Theory of Computer Viruses*, in "Advances in Cryptology — CRYPTO'88", vol. 403, Lecture Notes in Computer Science, 1988.

[31] E. ANSHELEVICH, A. DASGUPTA, E. TARDOS, T. WEXLER. *Near-optimal network design with selfish agents*, in "Proceedings of the thirty-fifth annual ACM symposium on Theory of computing", ACM Press, 2003, p. 511–520.

[32] E. ASARIN, O. MALER, A. PNUELI. *Reachability analysis of dynamical systems having piecewise-constant derivatives*, in "Theoretical Computer Science", vol. 138, n° 1, February 1995, p. 35–65.

[33] F. BAADER, T. NIPKOW. *Term rewriting and all that*, Cambridge University Press, New York, NY, USA, 1998.

[34] L. BLUM, M. SHUB, S. SMALE. *On a theory of computation and complexity over the real numbers; NP completeness, recursive functions and universal machines*, in "Bulletin of the American Mathematical Society", vol. 21, n° 1, July 1989, p. 1–46.

[35] G. BONFANTE, M. KACZMAREK, J.-Y. MARION. *On abstract computer virology: from a recursion-theoretic perspective*, in "Journal in Computer Virology", vol. 1, n° 3-4, 2006.

[36] O. BOURNEZ. *Modèles Continus. Calculs. Algorithmique Distribuée*, HDR, Institut National Polytechnique de Lorraine, 7 Décembre 2006.

[37] O. BOURNEZ, M. CAMPAGNOLO. *A Survey on Continuous Time Computation*, in "New Computational Paradigms", B. COOPER (editor), To appear., Springer-Verlag, 2007.

[38] O. BOURNEZ, F. GARNIER. *Proving Positive Almost Sure Termination*, in "16th International Conference on Rewriting Techniques and Applications (RTA'2005), Nara, Japan", J. GIESL (editor), Lecture Notes in Computer Science, vol. 3467, Springer, 2005, 323.

[39] O. BOURNEZ, C. KIRCHNER. *Probabilistic rewrite strategies: Applications to ELAN*, in "Proceedings of the 13th International Conference on Rewriting Techniques and Applications", S. TISON (editor), Lecture Notes in Computer Science, vol. 2378, Springer, 2002, p. 252–266.

[40] M.G.J. VAN DEN. BRAND, A. VAN. DEURSEN, J. HEERING, H.A. DE. JONG, M. DE. JONGE, T. KUIPERS, P. KLINT, L. MOONEN, P. OLIVIER, J. SCHEERDER, J. VINJU, E. VISSER, J. VISSER. *The ASF+SDF Meta-Environment: a Component-Based Language Development Environment*, in "Compiler Construction (CC '01)", R. WILHELM (editor), Lecture Notes in Computer Science, vol. 2027, Springer, 2001, p. 365–370.

[41] M. S. BRANICKY. *Universal computation and other capabilities of hybrid and continuous dynamical systems*, in "Theoretical Computer Science", vol. 138, n^{o} 1, February 1995, p. 67–100.

[42] M. CLAVEL, F. DURÁN, S. EKER, P. LINCOLN, N. MARTÍ-OLIET, J. MESEGUER, C. TALCOTT. *The Maude 2.0 System*, in "Proceedings of the 14th International Conference on Rewriting Techniques and Applications", R. NIEUWENHUIS (editor), Lecture Notes in Computer Science, vol. 2706, Springer, June 2003, p. 76-87.

[43] F. COHEN. *Computer Viruses*, Ph. D. Thesis, University of Southern California, January 1986.

[44] N. DERSHOWITZ, D. PLAISTED. *Rewriting*, in "Handbook of Automated Reasoning", A. ROBINSON, A. VORONKOV (editors), vol. I, chap. 9, Elsevier Science, 2001, p. 535-610.

[45] J. FEIGENBAUM, C. H. PAPADIMITRIOU, S. SHENKER. *Sharing the Cost of Multicast Transmissions*, in "Journal of Computer and System Sciences", vol. 63, n^{o} 1, 2001, p. 21–41, http://citeseer.nj.nec.com/feigenbaum01sharing.html.

[46] E. FILIOL. *Les virus informatiques: théorie, pratique et applications*, Springer-Verlag France, 2004.

[47] E. FILIOL. *Computer Viruses: from Theory to Applications*, Springer-Verlag, 2005.

[48] E. FILIOL. *Malware Pattern Scanning Schemes Secure Against Black-box Analysis*, in "Journal in Computer Virology", vol. 2, n^{o} 1, 2006, p. 35-50.

[49] E. FILIOL. *Techniques virales avancées*, Springer, 2007.

[50] E. FILIOL, G. JACOB, M. LE LIARD. *Evaluation methodology and theoretical model for antiviral behavioural detection strategies*, in "Journal in Computer Virology", vol. 3, n^{o} 1, 2007, p. 23-37.

[51] O. FISSORE, I. GNAEDIG, H. KIRCHNER. *Termination of rewriting with local strategies*, in "Selected papers of the 4th International Workshop on Strategies in Automated Deduction", M. P. BONACINA, B. GRAMLICH (editors), Electronic Notes in Theoretical Computer Science, vol. 58, esp, 2001.

[52] O. FISSORE, I. GNAEDIG, H. KIRCHNER. *CARIBOO : An induction based proof tool for termination with strategies*, in "Proceedings of the Fourth International Conference on Principles and Practice of Declarative Programming, Pittsburgh (USA)", ACM Press, October 2002, p. 62–73.

[53] O. FISSORE, I. GNAEDIG, H. KIRCHNER. *Outermost ground termination*, in "Proceedings of the Fourth International Workshop on Rewriting Logic and Its Applications, Pisa, Italy", Electronic Notes in Theoretical Computer Science, vol. 71, esp, September 2002.

[54] O. FISSORE, I. GNAEDIG, H. KIRCHNER. *A proof of weak termination providing the right way to terminate*, in "First International Colloquium on Theoretical Aspect of Computing, Guiyang, China", LECTURES NOTES IN COMPUTER SCIENCE, vol. 3407, Springer Verlag, September 2004, p. 356-371.

[55] O. FISSORE, I. GNAEDIG, H. KIRCHNER. *CARIBOO, a termination proof tool for rewriting-based programming languages with strategies, Version 1.0*, August 2004, Free GPL Licence, APP registration IDDN.FR.001.170013.000.R.P.2005.000.10600.

[56] O. FISSORE, I. GNAEDIG, H. KIRCHNER, L. MOUSSA. *CARIBOO, a termination proof tool for rewriting-based programming languages with strategies, Version 1.1*, December 2005, http://protheo.loria.fr/softwares/cariboo/, Free GPL Licence, APP registration IDDN.FR.001.170013.000.S.P.2005.000.10600.

[57] I. GNAEDIG, H. KIRCHNER. *Computing Constructor Forms with Non Terminating Rewrite Programs*, in "Proceedings of the Eighth ACM-SIGPLAN International Symposium on Principles and Practice of Declarative Programming, Venice, Italy", ACM Press, July 2006, p. 121–132.

[58] M. W. HIRSCH, S. SMALE, R. DEVANEY. *Differential Equations, Dynamical Systems, and an Introduction to Chaos*, Elsevier Academic Press, 2003.

[59] S. KLEENE. *Introduction to Metamathematics*, Van Nostrand, 1952.

[60] M. LUDWIG. *The Giant Black Book of Computer Viruses*, American Eagle Publications, 1998.

[61] J.-Y. MARION. *Complexité implicite des calculs, de la théorie à la pratique*, Habilitation à diriger les recherches, Université Nancy 2, 2000.

[62] J.-Y. MARION, J.-Y. MOYEN. *Efficient first order functional program interpreter with time bound certifications*, in "Logic for Programming and Automated Reasoning, 7th International Conference, LPAR 2000, Reunion Island, France", M. PARIGOT, A. VORONKOV (editors), Lectures Notes in Computer Science, vol. 1955, Springer, Nov 2000, p. 25–42.

[63] J.-Y. MARION, R. PÉCHOUX. *Resource Analysis by Sup-interpretation*, in "FLOPS", Lectures Notes in Computer Science, vol. 3945, Springer, 2006, p. 163–176.

[64] R. MCKELVEY, A. MCLENNAN. *Computation of equilibria in finite games*, in "Handbook of Computational Economics", Elsevier, 1996, http://citeseer.ist.psu.edu/mckelvey96computation.html.

[65] J. W. MILLS, B. HIMEBAUGH, A. ALLRED, D. BULWINKLE, N. DECKARD, N. GOPALAKRISHNAN, J. MILLER, T. MILLER, K. NAGAI, J. NAKAMURA, B. OLOLOWEYE, R. VLAS, P. WHITENER, M. YE, C. ZHANG. *Extended Analog Computers: A Unifying Paradigm for VLSI, Plastic and Colloidal Computing Systems*, in "Workshop on Unique Chips and Systems (UCAS-1). Held in conjunction with IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS05), Austin, Texas", March 2005.

[66] C. MOORE. *Dynamical recognizers: real-time language recognition by analog computers*, in "Theoretical Computer Science", vol. 201, n° 1–2, 6 July 1998, p. 99–136.

[67] P.-E. MOREAU, C. RINGEISSEN, M. VITTEK. *A Pattern Matching Compiler for Multiple Target Languages*, in "12th Conference on Compiler Construction, Warsaw (Poland)", G. HEDIN (editor), Lectures Notes in Computer Science, vol. 2622, Springer-Verlag, May 2003, p. 61–76.

[68] B. MORIN, L. MÉ. *Intrusion detection and virology: an analysis of differences, similarities and complementariness*, in "Journal in Computer Virology", vol. 3, n° 1, 2007, p. 33-49.

[69] J. D. MURRAY. *Mathematical Biology. I: An Introduction*, in "Biomathematics", vol. 17, Springer Verlag, 2002.

[70] J. F. NASH. *Equilibrium points in n-person games*, in "Proc. of the National Academy of Sciences", vol. 36, 1950, p. 48-49.

[71] N. NISAN, A. RONEN. *Algorithmic mechanism design (extended abstract)*, in "Proceedings of the thirty-first annual ACM symposium on Theory of computing", ACM Press, 1999, p. 129–140.

[72] P. ORPONEN. *A Survey of Continuous-Time Computation Theory*, in "Advances in Algorithms, Languages, and Complexity", D.-Z. DU, K.-I. KO (editors), Kluwer Academic Publishers, 1997, p. 209-224, http://citeseer.ist.psu.edu/orponen97survey.html.

[73] M. J. OSBOURNE, A. RUBINSTEIN. *A Course in Game Theory*, MIT Press, 1994.

[74] C. E. SHANNON. *Mathematical Theory of the Differential Analyser*, in "Journal of Mathematics and Physics MIT", vol. 20, 1941, p. 337-354.

[75] H. SIMMONS. *Derivation and Computation*, Tracts in theoretical computer science, vol. 51, Cambridge, 2000.

[76] P. SZOR. *The Art of Computer Virus Research and Defense*, Addison-Wesley Professional, 2005.

[77] TERESE. *Term Rewriting Systems*, Cambridge Tracts in Theoretical Computer Science, n° 55, Cambridge University Press, 2003.

[78] K. THOMPSON. *Reflections on Trusting Trust*, in "Communication of the ACM", Also appears in ACM Turing Award Lectures: The First Twenty Years 1965-1985, vol. 27, august 1984, p. 761–763.

[79] K. WEIHRAUCH. *Computable Analysis*, Springer, 2000.

[80] J. VON NEUMANN. *Theory of Self-Reproducing Automata*, edited and completed by A.W.Burks, University of Illinois Press, Urbana, Illinois, 1966.

[81] J. VON NEUMANN, O. MORGENSTERN. *Theory of Games and Economic Behavior*, First, Princeton University Press, Princeton, New Jersey, 1944.