INRIA

Project-Team Contraintes

Constraint Programming

Paris - Rocquencourt

THEME SYM

Activity
Report

2007

Table of contents

1. Team

Head of project-team

François Fages [Research Director (DR) INRIA, HdR]

Vice-head of project-team

Sylvain Soliman [Research Associate (CR) INRIA]

Administrative assistant

Nadia Mesrar [Secretary (SAR) INRIA]

Staff member

Grégory Batt [Research Associate (CR), INRIA, from October 07]

Pierre Deransart [Research Director (DR) INRIA, HdR]

Post-Doctoral fellow

Andras Kovacs [PostDoc ERCIM, up to June 07]

Sriram Krishnamachari [PostDoc ARC MOCA and Strep TEMPO]

Research scientist (partner)

Nicolas Beldiceanu [Professor, Ecole des Mines de Nantes]

Emmanuel Coquery [Associate Professor, University of Lyon 1]

Monika Heiner [Professor, on leave from the University of Cottbus, Germany, from March to September 07]

Denis Thieffry [Professor, on leave from the University of Marseilles, from October 2007]

Ph.D. students

Elisabetta De Maria [Univ. Udine, Italy, from October to December 07]

Rémy Haemmerlé [INRIA scholarship, up to November 07]

Domitille Heitzler [INRA scholarship]

Julien Martin [INRIA scholarship]

Thierry Martinez [INRIA scholarship, from October 07]

Aurélien Rizk [INRIA scholarship, from October 07]

Aurélie Strobbe [CIFRE TOTAL, up to June 07]

Student interns

Chi-Chia Huang [Univ. Taiwan]

Rishi Kumar [IIT Kanpur]

Vikas Marda [IIT Kanpur]

Thierry Martinez [ENS Ulm]

Jérôme Puissegur [ENS Lyon]

Aurélien Rizk [ENS Cachan]

2. Overall Objectives

2.1. Introduction

Constraint Logic Programming supports a great ambition for programming: the one of making of programming essentially a modeling task, with equations, constraints and logical formulas [11].

Constraint Programming is a field born during the mid 80s from Logic Programming, Linear Programming coming from Operations Research, and Constraint Propagation techniques coming from Artificial Intelligence. Its foundation is the use of relations on mathematical variables to compute with partial information. The successes of Constraint Programming for solving combinatorial optimization problems in industry or commerce are related to the bringing of new local consistency techniques and of *declarative languages* which allow control on the mixing of heterogeneous resolution techniques: numerical, symbolic, deductive and heuristic.

The "Contraintes" group investigates the logical foundations, design, implementation, programming environments and applications of constraint programming languages. The study of Concurrent Constraint languages is a core aspect of the project as they provide a conceptual framework for analyzing different issues of constraint programming, like constraint resolution techniques, concurrent modeling, reactive applications, etc.

The main application domains investigated are combinatorial optimization problems and computational system biology. In bioinformatics [10], our objective is not to work on structural biology problems, which has been the main trend up to now, but to attack the great challenge of systems biology, namely to model the function, activity and interaction of molecular systems in living cells, with logic programming concepts and program verification technologies.

2.2. Highlight: First Prize for the French Team at MIT's iGEM competition on Synthetic Biology using BIOCHAM

The international Genetically Engineered Machine Competition (iGEM) is a synthetic biology competition organized by the MIT. For its second year, iGEM gathered 52 teams from May to November 2007. For the first time a French team, composed of students in biology, engineering and computer science, including Aurélien Rizk and, in the last period, Grégory Batt both from *Contraintes*, participated to iGEM.

The French team goal was to design a synthetic multicellular bacterium to propose a new way of dealing with the complexity of synthetic systems by assigning different tasks to different cell lines. BIOCHAM has been used to model this system. More precisely BIOCHAM has been used to evaluate the robustness and optimization capabilities of two slightly different designs with regard to the system's specification thus providing valuable information to biologists for the construction process of the system.

This project was finalist for the grand prize (i.e. was one of the six best projects) and won the award of best Foundational Research.

3. Scientific Foundations

3.1. Concurrent constraint programming

The class of Concurrent Constraint programming languages (CC) was introduced a decade ago by Vijay Saraswat as a unifying framework for constraint logic programming and concurrent logic programming. The CC paradigm constitutes a representative abstraction of constraint programming languages, and thus allows a fine grained study of their fundamental properties.

CC generalizes the Constraint Logic Programming framework (CLP) by introducing a synchronization primitive, based on constraint entailment. It is a model of concurrent computation, where agents communicate through a shared store, represented by a constraint, which expresses some *partial information* on the values of the variables involved in the computation. The variables play the role of transmissible dynamically created communication channels.

One of the big successes of CC has been the simple and elegant reconstruction of finite domain constraint solvers, and the cooperation of several models to solve a single combinatorial problem. On the other hand, to use CC for programming reactive applications forces one to abandon the hypothesis of monotonic evolution of the constraint store; this is a strong motivation for new extensions of CC languages.

There are strong completeness theorems relating the execution of a CLP program and its translation in classical logic, which provide smooth reasoning techniques for such programs. However these theorems are broken by the synchronization operation of CC. Looking for a logical semantics of CC programs in the general paradigm of logic programming,

$$program = logical\ formula,$$
$$execution = proof\ search,$$

leads to a translation in Jean-Yves Girard's linear logic. This allows the recovery of some completeness results about successes and stores; even suspensions may be characterized with the non-commutative logic of Ruet and Abrusci.

It is thus possible to address important issues for Constraint Programming:

- verifying CC programs;
- combining CLP and state-based programming;
- dealing with local search inside a global constraint solving procedure.

The last two cases rely on a natural extension of CC languages, called Linear Concurrent Constraint languages (LCC), which simply replaces constraint systems built onto classical logic by constraint systems built onto linear logic. This allows us to represent state changes thanks to the consumption of resources during the synchronization action, modeled by the linear implication.

3.2. Constraint solvers

Our domains of application use quite different constraint systems:

- finite domains (bounded natural numbers): primitive constraint of some finite domain membership, numerical, symbolic, higher order and global constraints;
- reals: Simplex algorithm for linear constraints and interval methods otherwise;
- terms: subtyping constraints and ontologies;
- temporal constraints: CTL and LTL formulae, either propositional or with numerical constraints.

The project works on constraint resolution methods and their cooperation. The main focus is the declarativeness of the constraint solver (e.g. implemented by CHR rules), the efficiency of constraint propagation methods, the design of global constraints and the combination of constraint propagation with heuristic search.

3.3. Computational systems biology

Systems biology is a cross-disciplinary domain involving biology, computer science, logics, mathematics, and physics to elucidate the high-level functions of the cell from their biochemical bases at the molecular level.

At the end of the Nineties, research in Bioinformatics evolved, passing from the analysis of the genomic sequence to the analysis of post-genomic interaction networks (expression of RNA and proteins, protein-protein interactions, etc). The complexity of these networks requires a large research effort to develop symbolic notation and analysis tools for biological processes and data. In order to scale-up, and get over the complexity walls to reason about biological systems, there is a general feeling that beyond providing tools to biologists, computer science has much to offer in terms of concepts and methods.

We are interested in the modeling and analysis of complex molecular processes in the cell, at different levels of abstraction, qualitative and quantitative. The most original aspect of our research can be summarized by the following identifications:

biological model = transition system,

biological property = temporal logic formula,

biological validation = model-checking.

Our main research axis is thus the application of logic programming concepts and circuit or program verification techniques to the analysis of complex biochemical processes in the cell.

4. Application Domains

4.1. Combinatorial optimization problems

The number and economic impact of combinatorial optimization problems found in the industrial world are constantly increasing. They cover:

- resource allocation;
- placement;
- scheduling;
- planing;
- transport;
- etc.

The last forty years have brought many improvements in Operations Research resolution techniques. In this context, Constraint Programming can be seen as providing, on the one hand, local consistency techniques that can be applied to various numerical or symbolic constraints, and on the other hand, declarative languages. This last point is crucial for quickly developing complex combinations of algorithms, which is not possible without a language with a high level of abstraction. It allowed for better results, for instance in scheduling problems, than traditional methods, and is promised to an even better future when thinking about the cooperation of global resolution, local consistency techniques and search methods.

The project builds upon its knowledge of CC languages, constraint solvers and their implementation to work in these directions. The LCC paradigm offers at the same time a theoretical framework for analysis, and a valuable guide for practical language design and implementation. The work on programming environments helps to integrate the Constraint Programming tools into this application domain.

4.2. In-silico cell

In 2002 , we started a Collaborative Research Initiative ARC CPBIO on "Process Calculi and Biology of Molecular Networks". By working on well understood biological models, we sought:

- to identify in the family of competitive models coming from the Theory of Concurrency[1] and from Logic Programming [2], the ingredients of a language for the modular and multi-scale representation of biological processes;
- to provide a series of examples of biomolecular processes transcribed in formal languages, and a set of biological questions of interest about these models;
- to design and apply to these examples formal computational reasoning tools for the simulation, the analysis and the querying of the models.

This work lead us to the design and implementation of the Biochemical Abstract Machine BIOCHAM that has the unique feature of providing formal languages corresponding to different qualitative and quantitative levels of abstraction, for, on the one hand, modeling biomolecular interaction diagrams with reaction rules, and on the other hand, modeling the biological properties of the system in temporal logic. This double formalization of both the model and the biological properties of the system at hand opens several new research avenues on the design and systematic validation of biological models.

In the framework of the 6th PCRD STREP project APrIL II (2004-2007) the main objective was to apply probabilistic inductive logic programming techniques to applications in bioinformatics. Our focus was on regulatory and metabolic networks in the cell, and on the semi-automatic completion or revision of models [13] from the temporal properties of the system as observed in biological experiments [19].

---

[1] π-calculus, Join-calculus and their derivatives

[2] Constraint Logic Programming, Concurrent Constraint languages and their extensions to discrete and continuous time, TCC, HCC

In the Network of Excellence REWERSE (2004-2008), the focus is on the application of the new Semantic Web technologies based on rules and constraints to bioinformatics. In this context we use the biological knowledge stored in online ontologies like GO in order to improve the re-use and composition of models, as well as the semi-automatic correction/completion of models mentionned above. In this respect, we developed type inference and abstract interpretation techniques to relate information at different levels of abstraction and use them reaction models [12].

We coordinated the ARC MOCA (2006-2007) on "MOdularity, Compositionality and Abstraction in gene and protein networks", and collaborate in this framework with Denis Thieffry, Claudine Chaouyia, Univ. Marseille, Anne Siegel and Ovidiu Radulescu, IRISA SYMBIOSE and IHES, Ralf Blossey IRI and Vincent Danos CNRS Paris, on the exploration of different techniques and methods allowing semi-automatic composition, decomposition, simplification and refinement of biological models [20]. More precisely, we studied the formal links between logical and numerical models of some parts of the cell cycle control, and its decomposition into modules based on control theory considerations.

These technologies are applied to new biological questions that we investigate in partnerships with biologists in two more recent projects. First, the EU STREP project TEMPO (2006-2009) on "temporal genomics for patient tailored chornotherapeutics", coordinated by Francis Lévi INSERM Villejuif, where, in partnership with Jean Clairambault of the BANG project-team, we develop in BIOCHAM coupled models of the cell cycle, the circadian cycle and the effect of cytotoxic drugs in cancer therapies [9].

Second, the INRA AgroBi project INSIGHT, coordinated by Eric Reiter INRA Tours, where, in partnership with Frédérique Clément of the SISYPHE project-team, we develop models of FSH signaling networks in mammalian cells.

5. Software

5.1. BIOCHAM

Participants: François Fages, Sylvain Soliman.

The Biochemical Abstract Machine BIOCHAM is a modeling and validation environment for molecular systems biology. BIOCHAM provides precise semantics to biomolecular interaction maps at three abstraction levels:

1. the boolean semantics of molecules presence and absence,
2. the continuous semantics of molecular concentrations,
3. the stochastic semantics of molecule populations.

Based on this formal framework, BIOCHAM offers:

- a compositional rule-based language for modeling biochemical systems, allowing patterns and kinetic expressions when numerical data are available;
- numerical and boolean simulators;
- a temporal logic language (CTL for qualitative models and LTL with numerical constraints for quantitative models) for formalizing biological properties such as reachability, checkpoints, oscillations or stability, and checking them automatically with model-checking techniques;
- automatic search procedures to infer interaction rules and parameter values from known temporal properties of the system.

BIOCHAM is fully implemented in GNU-Prolog and interfaced to the state-of-the-art symbolic model checker NuSMV for evaluating boolean CTL queries in large models over several hundreds of variables. BIOCHAM models can be imported from, and exported to, the standard Systems Biology Markup Language SBML.

5.2. TCLP

Participant: Emmanuel Coquery.

TCLP is a prescriptive type system for Constraint Logic Programming, currently: ISO-Prolog, GNU-Prolog, SICStus Prolog and the constraint programming libraries of SICStus Prolog. TCLP can also type check constraint solvers written in the *Constraint Handling Rules* (CHR) language.

The flexibility of the TCLP type system is obtained by the combination of three kinds of polymorphism: parametric polymorphism (e.g. *list(A)*), subtyping (e.g. *list(A)<term*) and overloading (e.g. *−:num∗num→num* and *−:A∗B→pair(A,B)*). No type declaration are required, thanks to a type inference algorithm for predicates and to a default *term* type for function symbols.

5.3. GNU-Prolog-RH

Participant: Rémy Haemmerlé.

GNU Prolog is a free Prolog compiler with constraint solving over finite domains developed by Daniel Diaz. GNU Prolog accepts Prolog extended with primitives for constraint programming and produces native binaries (like gcc does from a C source). The Prolog part conforms to the ISO standard for Prolog with many practical extensions (global variables, OS interface, sockets,...). GNU Prolog also includes an efficient constraint solver over Finite Domains (FD), giving the user the combined power of constraint programming and the declarativity of logic programming.

An experimental version, called GNU-Prolog-RH, is developed as an extension introducing concurrency, attributed variables, and constraint solving over the reals. This version greatly extends the expressive and modeling power of GNU-Prolog. It is also used to prototype the design and implementation of our new SiLCC language.

5.4. CLPGUI

Participant: François Fages.

CLPGUI is a generic graphical user interface written in Java for constraint logic programming. It is available for GNU-Prolog and SICStus Prolog. CLPGUI has been developed both for teaching purposes and for debugging complex programs. The graphical user interface is composed of several windows: one main console and several dynamic 2D and 3D viewers of the search tree and of finite domain variables. With CLPGUI it is possible to execute incrementally any goal, backtrack or recompute any state represented as a node in the search tree. The level of granularity of the search tree is defined by annotations in the CLP program.

6. New Results

6.1. Implementation of SiLCC

Participants: François Fages, Rémy Haemmerlé, Thierry Martinez, Sylvain Soliman.

We are developing SiLCC an imperative and concurrent constraint programming language based on a single paradigm: the one of Vijay Saraswat's concurrent constraint programming extended with constraint systems based on Jean-Yves Girard's Linear Logic. In the late 90's we developed the theory of this extension and we have now begun its implementation.

From a constraint programming point of view, the unique combination of constraint programming with imperative features opens many new possibilities, among which:

- the capability of programming constraint solvers in the language, making them extensible by the user,

- making a fully bootstrapped implementation of a constraint programming language (for the first time since Prolog)

- combining constraint reasoning with state change;

- embedding program declarations, modules and closures as agents;

- proving program correctness using Linear Logic.

The main step realized this year has been the use of GNU-Prolog-RH as temporary kernel language to develop a first bootstrapped prototype. Its module system relies upon the theoretical description of [5]. Our prototype implementation was used to participate in the ICFP programming contest and obtain the 59th place over 357 teams submitting a solution.

The objective is now to define a small kernel language as an instance of LCC over a simple constraint domain, on top of which the complete SiLCC language will be built. Bootstrapping is a fundamental step for getting over the current limits of today constraint programming tools, concerning their extensibility, robustness, and teaching. A new prototype, not using GNU-Prolog at all, is being worked on to match closely the theoretical expressiveness of LCC and to circumscribe the kernel constraint system more precisely.

6.2. Module System for SiLCC

Participants: François Fages, Rémy Haemmerlé, Rishi Kumar, Thierry Martinez, Sylvain Soliman.

There are two somewhat contradictory ways of looking at modules in a given programming language. On the one hand, module systems are largely independent of the particulars of programming languages. On the other hand, the module constructs may interfere with the programming constructs, and may be redundant with the other scope mechanisms of a specific programming language, such as closures for instance. There is therefore a need to unify the programming concepts that are similar, and retain a minimum number of essential constructs to avoid arbitrary programming choices.

In [22], [5], we realize this aim in the framework of linear logic concurrent constraint programming (LCC) languages. We first show how declarations and closures can be internalized as agents in a variant of LCC for which we provide precise operational and logical semantics in linear logic. Then, we show how a complete module system can be represented within LCC, and prove for it a general code protection property. Finally we study the instanciation of this scheme to the implementation of a safe module system for constraint logic programs, and conclude on the generality of this approach to programming languages with logical variables.

This approach serves as a design principle for the module system of SiLCC and its bootstrapped implementation [30].

We apply it also to the design and implementation of a modular version of the Constaint Handling Rules language CHR, where the primitive constraints at one level can be defined by CHR programs at lower levels [28].

6.3. Confluence of Arbitrary Binary Relations

Participants: François Fages, Rémy Haemmerlé.

In a seminal paper, Huet introduced abstract properties of term rewriting systems, and the confluence analysis of terminating term rewriting systems by critical pairs computation. However, to date, there was no abstract notion of critical pairs. In [21], we introduce such an abstract notion for arbitrary binary relations and arbitrary context operators. We show how this framework applies to the confluence analysis of various transition systems, ranging from classical term rewriting systems to production rules with constraints and partial control strategies, such as the Constraint Handling Rules language CHR. Interestingly, we show in all these cases that some classical critical pairs can be disregarded. The crux of these analyses is the ability to compute critical pairs between states built with general context operators, on which a bounded, not necessarily well-founded, ordering is assumed.

6.4. Traces

Participant: Pierre Deransart.

Semantics based debugging, static and dynamic typing, and more generally validation of CC programs are also studied in the project.

We study theoretical questions issued from the former OADymPPaC project (RNTL ended in 2004) by trying to clarify the notions of trace exchanged between observed and observing processees and their semantics called "observational semantics". We wanted also to better formalize the interactions between these processes in order to be able to optimize their communications in particular by formalizing the introduction of Tracer Driver and Analyser Manager respectively added to the oberserved and observing processes. Trace definition, generation and analysis play an important role in sofware verification. We thus develop a methodology to relate the interpretation which can be given to a trace, with the observational semantics, using two notions of traces [15], [32]: the "actual trace", the data stream generated by (the tracer of) a process, and the "full virtual trace", an abstracted stream, sequence of virtual states which may be related with the observational semantics. We have illustrated this methodology with the case of the classical trace of Prolog, implemented in most Prolog solvers [25], [16], [17].

6.5. Bin Packing and Placement Constraints

Participants: François Fages, Andras Kovacs, Julien Martin.

In the framework of the Net-WMS European project, we study higher-dimensional bin packing problems and placement constraints [26]. In [24], we describe a knowledge representation language for Packing problems allowing to state higher-dimensional placement constraints with business rules, and to translate them into constraint programs. This paper presents a three-level hierarchy of knowledge representation languages for Packing problems: geometrical constraints, placement constraints and business rules. The challenge is to translate business rules into efficient constraint programs in this domain. A first implementation in SICStus prolog, Constraint Handling Rules and CLPGUI has been developed and demonstrated.

6.6. Global constraint of scheduling with concentration constraints

Participants: François Fages, Aurélie Strobbe.

In collaboration with TOTAL, we have studied an optimization problem of crude oil blending in refineries. The problem was to find a plan of transfer and mixing of crude oil, from the boats to the distillation units through storage and mixing tanks, under various constraints of product concentrations, continuous distillation, schedule, and optimization criteria. This problem is NP-hard and is the combination of a planning problem (for choosing the transfer tasks and quantities) with a scheduling problem.

Previous approaches based on Mixed-Integer Linear Programming (MILP) relied on a discretization of time that may give infeasible or suboptimal solutions. On the other hand, a constraint programming approach was appealing for its capability to treat efficiently the scheduling part of the problem without time discretization, We have developed such a solution with ILOG tools, by introducing a global constraint of scheduling with concentration constraints, and using heuristics for the planning part of the problem [6]. A filtering algorithm for this global constraint has also been investigated as a generalization of the multi-resource cumulative constraint.

6.7. Abstract Interpretation for Systems Biology I: Hierarchy of Semantics

Participants: François Fages, Sylvain Soliman.

Abstract interpretation is a theory of abstraction that has been introduced for the analysis of programs. In particular, it has proved useful for organizing the multiple semantics of a given programming language in a hierarchy of semantics corresponding to different detail levels, and for defining type systems for programming languages and program analyzers in software engineering. We have investigated the application of these concepts to systems biology formalisms.

More specifically, we consider the Systems Biology Markup Language SBML, and the Biochemical Abstract Machine BIOCHAM with its differential, stochastic, discrete and boolean semantics [8]. In [12], we show how all of these different semantics, except the differential one, can be formally related by simple Galois connections. This provides a simple algebraic setting for relating the different interpretations of biochemical networks, and a formal ground conducting precise analyses.

6.8. Abstract Interpretation for Systems Biology II: Types

Participants: François Fages, Jérôme Puissegur, Sylvain Soliman, Vikas Marda.

Still in the framework of abstract interpretation, we have developed three type systems for biochemical reaction models, showing how the problem of relating different models corresponding to different abstraction levels could be formalized as type checking problems by analogy to programming languages [12]. We show that the framework of abstract interpretation elegantly applies to the formalization of these further abstractions, and to the implementation of linear or quadratic time type checking as well as type inference algorithms. Through some examples, we show that the analysis of biochemical models by type inference provides accurate and useful information. Interestingly, such a mathematical formalization of the abstractions commonly used in systems biology already provides some guidelines for the extensions of biochemical reaction rule languages.

The first type system is for checking or inferring the functions of proteins in a reaction model. This type system provides a way to establish connections between reaction models and gene and protein ontologies on the web. A prototype implementation has been realized to query the ontologies on the web from the molecule names in BIOCHAM [29].

The second type system is for checking or inferring the activation and inhibition effects of proteins in a reaction model, called the influence graph. In [12], we show a theorem of equivalence between the definition of the graph obtained from the differential semantics (by the signs of Jacobian matrix coefficients) with the one obtained from the syntax of the reactions (by the differences between the left and right hand sides of the rules). This theorem holds under general assumptions of monotonicity of the kinetic expressions satisfied by the usual kinetic laws (mass action, Michaelis-Menten, Hill). It shows that the influence graph can be computed very efficiently by type inference from the syntax of the rules, instead of by sign analysis in the Jacobian matrix.

The third type system is for checking or inferring the topology of compartments or locations in a reaction model.

Related to these studies, we have also designed a constraint system for describing combinatorics of molecular species in a concise fashion [31].

6.9. Stochastic Semantics of Reaction Models

Participants: François Fages, Chi-Chia Huang, Sylvain Soliman.

The stochastic semantics of BIOCHAM [12] has been implemented using Gillespie's stochastic simulation algorithm (SSA) and the tau-leaping methods [27]. To make the simulation compatible with the checking of temporal logic formulas, special methods for filtering excrescent data and smoothing the curve have been developed. Similarly due to the characteristic of stochastic simulation, some query methods of BIOCHAM are no longer appropriate and have been adapted.

6.10. Analysis of Numerical Data Time Series in Temporal Logic

Participants: François Fages, Sylvain Soliman, Aurélien Rizk.

We have shown how temporal logics and model-checking techniques can be used to respectively express biological properties of complex biochemical systems, and automatically verify their satisfaction in both qualitative and quantitative models [34], [33]. In [19], we propose a finite time horizon model-checking algorithm for the existential fragment of LTL with numerical constraints over the reals, with the ability to compute the range of values of the real variables occurring in a formula that makes it true in a model. We illustrate this approach for the analysis of biological data time series, provide a set of biologically relevant patterns of formulas, and evaluate them on models of the cell cycle control and MAPK signal transduction.

6.11. Temporal Logic Analysis of Gene Networks under Parameter Uncertainty

Participant: Grégory Batt.

The lack of precise numerical information for the values of biological parameters severely limits the development and analysis of models of genetic regulatory networks. To deal with this problem, we propose in [7], a method for the analysis of genetic regulatory networks under parameter uncertainty. We consider models based on piecewise-multiaffine differential equations, dynamical properties expressed in temporal logic, and intervals for the values of uncertain parameters. The problem is then either to guarantee that the system satisfies the expected properties for every possible parameter value – the corresponding parameter set is then called valid – or to find valid subsets of a given parameter set. The proposed method uses discrete abstractions and model checking, and allows for efficient search of the parameter space. However, the abstraction process creates spurious behaviors in the abstract systems, along which time does not progress. Consequently, the verification of liveness properties, expressing that something will eventually happen, and implicitly assuming progress of time, often fails. A solution to this second problem is proposed using the notion of transient regions. This approach has been implemented in a tool for robust verification of gene networks (RoVerGeNe) and applied to the tuning of a synthetic network built in *E. coli*.

6.12. Coupled Model of the Cell Cycle and the Circadian Cycle in Mammalian Cells

Participants: Elisabetta De Maria, François Fages, Sriram Krishnamachari, Sylvain Soliman, Aurélien Rizk.

Recent advances in cancer chronotherapy techniques support the evidence that there exist some links between the cell and the circadian cycles. Both cycles have been successfully modeled, however, as of today, there are no precise models describing the coupling of the two cycles. One purpose of a coupled model is to better understand how to efficiently target malignant cells depending on the phase of the day. This is at the heart of our participation in the EU STREP project TEMPO.

Our model is built from two models, a model of the circadian cycle from Leloup and Goldbeter and a generic model of the cell cycle model focusing on a crucial event, the entry into mitosis. Currently, the model is focused on the coupling through the WEE1 kinase. This protein plays an important role in the regulation of the activity of MPF, a complex crucial in cell division. The WEE1 kinase seems to be regulated positively by CLOCK/BMAL1 and negatively by PER/CRY, these genes being well-known circadian core genes.

The BIOCHAM feature for learning parameter values under period constraints was introduced for this study. This allowed us to compute the domain of entrainment of the cell cycle by the circadian cycle. This domain mainly depends on the strength of the WEE1 kinase, regulated by circadian genes, on the activity of MPF. We now intend to incorporate other links, based on evidence from the literature: through p21 or c-MYC for instance; and to make the generic cell cycle model more complete in order to see the effect of these other links, especially on the phase of the cell cycle.

6.13. Connectivity and Dynamics of the FSH Signalling Network in Granulosa Cells

Participants: François Fages, Domitille Heitzler.

In collaboration with Frédérique Clément (SISYPHE) and Eric Reiter (UMR CNRS-INRA 6175), and in the framework of the INRA AgroBI INSIGHT project, we analyse the connectivity and dynamics of the FSH signalling network in its target cells, and embedding the network within a multi-scale representation, from the molecular up to the organic level. We are examining the relative contributions of different pathways to the cell response to FSH signal, in order to determine how each pathway controls downstream cascades and which mechanisms are involved in the transition between different cellular states. We have used the BIOCHAM (BIOCHemical Abstract Machine) environment, first at the boolean level, to formalise the network of interactions corresponding to the FSH-induced signalling events on the cellular scale. This network has then be enriched with kinetic information coming from experimental data, which allows us to use the ordinary differential equation level of BIOCHAM. In order to find and fine-tune the structure of the network and the values of the kinetic parameters, model-checking techniques are applied to undertake a systematic comparison between the model behaviour and the results of experiments.

7. Contracts and Grants with Industry

7.1. TOTAL

Industrial contract with TOTAL (July 2004- July 2007) on the optimization of crude oil blending, and PhD Thesis of Aurélie Strobbe [6] under a CIFRE contract.

8. Other Grants and Activities

8.1. National contracts

- INRA project AgroBi (2006-2010) on the modeling of FSH signaling, coordinated by Eric Reiter, INRA Tours;
- ARC MOCA (2006-2007) MOdularité, Compositionalité et Abstraction dans les réseaux géniques et protéiques, coordinated by Sylvain Soliman;
- ACI IMPbio VICANNE (2004-2007) Modélisation dynamique et simulation des systèmes biologiques;

8.2. European contracts

- 6th PCRD STREP Net-WMS (2006-2010) on constraint optimization in Wharehouse Management Systems, ERCIM coord, F. Fages scientific coordinator, Ecole des Mines de Nantes, SICS, KLS optim, CEA, mindbiz, Widescope, PSA, Fiat;
- 6th PCRD STREP TEMPO (2006-2010) on temporal genomics for tailored chronotherapeutics, coordinated by Francis Lévi at INSERM Villejuif;
- 6th PCRD STREP APrIL II (2004-2007) "Applications of probabilistic inductive logic programming", coord. Prof. L. de Raedt, University of Freiburg;
- 6th PCRD Network of Excellence REWERSE (2004-2008) "Reasoning with rules and semantics", coord. Prof. F. Bry, Ludwig Maximillian's University in Munich;
- ERCIM Working Group on Constraints, coord. F. Fages, INRIA Rocquencourt.

8.3. International contracts

Participants: Pierre Deransart, François Fages.

We launched a new project C4RBCP in the framework of the bilateral cooperation with Brazil, with Jacques Robin from the UFPE.

In this project, we investigate the synergy between two hitherto unrelated areas of computer science: Rule-Based Constraint Programming (RBCP) and Component-Based Software Engineering (CBSE). These two areas bring complementary principles to the general issue of software reuse for Automated Reasoning (AR). We thus investigate how these CBSE concepts can be adapted to the RBPC language CHRÚ (Constraint Handling Rules with Disjunctive bodies, a simple yet powerful extension of CHR) and implemented in the adaptive CHRÚ engine CHROME (CHR Online Model-driven Engine) currently under development at CIn-UFPE.

8.4. Invitations

Have been invited for short visits :

- Ludovic Langevine, Mission Critical, Belgium,
- Jacques Robin, Universidade Federal de Pernambuco, Brazil,
- Samuel Bernard, Humboldt University, Berlin, Germany,
- Gregory Batt, Boston University.

9. Dissemination

9.1. Teaching

Contraintes is affiliated to the Doctoral school of Mathematical Science of the University of Paris 7.

The following post-graduate courses have been given:

- 24h Course on *Constraint Programming*, Master Parisien de Recherche en Informatique (MPRI) by Sylvain Soliman (15h) and François Fages (9h).
- 48h Course on *Computational Systems Biology*, Master Parisien de Recherche en Informatique (MPRI) by François Fages (15h), Vincent Danos (15h) and Vincent Schächter (18h).
- 6h Course on Constraint Programming at Ecole Jeunes Chercheurs en Programmation, Rennes, by François Fages (6h).
- 6h Course on Formal Verification of Dynamical Systems and Application to Cell Cycle Control, Spring School on Dynamical Modelling of Biological Regulatory Networks, Les Houches, by François Fages (6h).

9.2. Leadership within scientific community

- Pierre Deransart is the General Secretary, past Chairman, of the "Association Française pour la Programmation par Contraintes" AFPC. He is "Brazil Correspondent" in the DRI (INRIA international Direction) and organized several INRIA visits and activities related to INRIA-Brazil scientific cooperation. He was General Chair of JFPC'07 JFPC'07 which he organized at INRIA-Rocquencourt in June.
- François Fages is member of the Editorial Board of RAIRO Operations Research. programme chair of JFPC'07 [1], conference co-chair of CP-AI-OR'08, Chairman of the ERCIM Working Group on Constraints, past Chairman of the "Association Française pour la Programmation par Contraintes" AFPC, member of the Scientific Council of the Epigenomics project of the Genopole of Evry, and Chairman of the evaluation committee of the dept. "Conduite et Décision" ONERA.
- Sylvain Soliman is the Secretary of the ERCIM Working Group on Constraints, co-organizer of CSCLP'07 [2], and was invited to an open session of Académie des Sciences [38].

10. Bibliography

Year Publications

Books and Monographs

[1] F. FAGES, S. SOLIMAN (editors). *Actes des 3ièmes Journées Francophones de Programmation par Contraintes JFPC'07*, INRIA, Rocquencourt, France, 2007.

[2] F. FAGES, S. SOLIMAN, F. ROSSI (editors). *Proceedings of the 12th Annual ERCIM International Workshop on Constraint Solving and Constraint Logic Programming CSCLP'07*, INRIA, Rocquencourt, France, 2007.

[3] S. SPACCAPIETRA, P. ATZENI, F. FAGES, M.-S. HACID, M. KIFER, J. MYLOPOULOS, B. PERNICI, P. SHVAIKO, J. TRUJILLO, I. ZAIHRAYEU (editors). *Journal on Data Semantics IX*, Lecture Notes in Computer Science, vol. 4601, Springer-Verlag, 2007.

[4] S. SPACCAPIETRA, P. ATZENI, F. FAGES, M.-S. HACID, M. KIFER, J. MYLOPOULOS, B. PERNICI, P. SHVAIKO, J. TRUJILLO, I. ZAIHRAYEU (editors). *Journal on Data Semantics VIII*, Lecture Notes in Computer Science, vol. 4380, Springer-Verlag, 2007.

Doctoral dissertations and Habilitation theses

[5] R. HAEMMERLÉ. *Fermetures et Modules dans les Langages Concurrents avec Contraintes fondés sur la Logique Linéaire*, Ph. D. Thesis, Univ. Paris 7. Soutenance le 17 janvier 2008, December 2007.

[6] A. STROBBE. *Application de la programmation par contraintes a un problème d'optimisation des mélanges de bruts en raffinerie*, Ph. D. Thesis, Univ. Paris 6. Thèse achevée non soutenue, June 2007.

Articles in refereed journals and book chapters

[7] G. BATT, C. BELTA, R. WEISS. *Temporal Logic Analysis of Gene Networks under Parameter Uncertainty*, in "IEEE Transactions on Circuits and Systems and IEEE Transactions on Automatic Control", Joint special issue on Systems Biology, to appear, 2007.

[8] L. CALZONE, N. CHABRIER-RIVIER, F. FAGES, S. SOLIMAN. *Langages formels dans la machine abstraite biochimique BIOCHAM*, in "Techniques et Sciences Informatiques", vol. 26, n° 1–2, 2007, p. 47–72, http://contraintes.inria.fr/~fages/Papers/CCFS05tsi.pdf.

[9] J. CLAIRAMBAULT, F. FAGES, S. SOLIMAN. *Patient-tailored cancer therapeutics - The TEMPO Project*, in "ERCIM News", vol. 69, April 2007, p. 24–25.

[10] A. DAL PALU, A. DOVIER, F. FAGES, S. WILL. *Constraint-based Methods in Bioinformatics*, in "Future and trends of Constraint Programming: the next ten years", F. BENHAMOU, N. JUSSIEN, B. O'SULLIVAN (editors), Hermès, 2007.

[11] F. FAGES. *Des contraintes au programme*, in "Revue littéraire de l'Institut Universitaire de France, Actes de Savoirs", December 2007.

[12] F. FAGES, S. SOLIMAN. *Abstract Interpretation and Types for Systems Biology*, in "Submitted to Theoretical Computer Science", August 2007.

[13] F. FAGES, S. SOLIMAN. *Model Revision from Temporal Logic Properties in Systems Biology*, in "Applications of Probabilistic Inductive Logic Programming", To appear, 2007.

Publications in Conferences and Workshops

[14] *Recent Advances in Constraints, 11th Annual ERCIM International Workshop on Constraint Solving and Constraint Logic Programming CSCLP'06*F. AZEVEDO, P. BARAHONA, F. FAGES, F. ROSSI (editors)., Lecture Notes in Artificial Intelligence, vol. 4651, Springer-Verlag, 2007.

[15] P. DERANSART. *Comprendre les traces (une approche sémantique des traces)*, in "Actes du colloque de l'association pour la recherche cognitive, ARCo 07, Poster, Nancy, France", N. GREGORI (editor), November 2007.

[16] P. DERANSART, M. DUCASSÉ, G. FERRAND. *Observational Semantics of the Resolution Box Model*, in "Proceedings of the 17th Workshop on Logic-based Methods in Programming Environments (WLPE'07), a post-conference workshop of ICLP'07, Porto, Portugal", W. VANHOOF, P. HILL (editors), à paraître dans le Computing Research Repository (CoRR), September 2007.

[17] P. DERANSART, M. DUCASSÉ, G. FERRAND. *Une sémantique Observationnelle du modèle des boîtes pour la résolution de programmes logiques*, in "Actes des troisièmes Journées Francophones de Programmation par Contraintes, JFPC 07, Rocquencourt, France", F. FAGES, S. SOLIMAN (editors), June 2007.

[18] F. FAGES. *Formal Verification of Biochemical Models and Application to Cell Cycle (invited talk)*, in "Integrative Post-Genomique IPG'07, Lyon, France", November 2007.

[19] F. FAGES, A. RIZK. *On the Analysis of Numerical Data Time Series in Temporal Logic*, in "CMSB'07: Proceedings of the fifth international conference on Computational Methods in Systems Biology", Lecture Notes in Computer Science, vol. 4695, Springer-Verlag, 2007, p. 48–63.

[20] D. GILBERT, M. HEINER, S. LEHRACK. *A Unifying Framework for Modelling and Analysing Biochemical Pathways Using Petri Nets*, in "CMSB'07: Proceedings of the fifth international conference on Computational Methods in Systems Biology", Lecture Notes in Computer Science, vol. 4695, Springer-Verlag, 2007.

[21] R. HAEMMERLÉ, F. FAGES. *Abstract Critical Pairs and Confluence of Arbitrary Binary Relations*, in "Proceedings of th 18th International Conference on Rewriting Techniques and Applications, RTA'07", Lecture Notes in Computer Science, n° 4533, Springer-Verlag, 2007, p. 214–228, http://contraintes.inria. fr/~haemmerl/pub/HF07rta.pdf.

[22] R. HAEMMERLÉ, F. FAGES, S. SOLIMAN. *Closures and Modules within Linear Logic Concurrent Constraint Programming*, in "Proceedings of FSTTCS 2007, IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science", V. ARVIND, S. PRASAD (editors), Lecture Notes in Computer Science, vol. 4855, Springer-Verlag, 2007, p. 544–556.

[23] A. KOVÁCS, J. C. BECK. *A Global Constraint for Total Weighted Completion Time*, in "Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems", Lecture Notes in Computer Science, vol. 4510, Springer-Verlag, 2007, p. 112–126.

[24] J. MARTIN, F. FAGES. *From Business Rules to Constraint Programs in Wharehouse Management Systems*, in "Doctoral programme of the 13th Conference on Principles and Practice of Constraint Programming CP'07, Providence, MA, USA", September 2007, http://voter.engr.uconn.edu/~ldm/CP2007/DP-proc2007.pdf.

Internal Reports

[25] P. DERANSART, M. DUCASSÉ, G. FERRAND. *Une sémantique observationnelle du modèle des boîtes pour la résolution de programmes logiques (version étendue)*, RR-6229, INRIA, May 2007, http://hal.inria.fr/inria-00151285.

[26] F. FAGES, A. AGGOUN, N. BELDICEANU, M. CARLSSON, F. CARVALHO, P. GRAVEZ, A. KOVÁCS, J. MARTIN. *State-of-the-art of enabling technologies for packing and planning in future WMS*, deliverable, n$^{\text{o}}$ D3.1, INRIA, 2007.

[27] C.-C. HUANG. *Stochastic Simulation in BIOCHAM*, Rapport de stage de Master de l'université de Taiwan, INRIA, July 2007.

[28] R. KUMAR. *Modularization of CHR*, Rapport de stage de Master de l'IIT Kanpur, INRIA, July 2007.

[29] V. MARDA. *Introduction of an Alias mechanism and Connecting BIOCHAM to the Web*, Rapport de stage de Master de l'IIT Kanpur, INRIA, July 2007.

[30] T. MARTINEZ. *Analyse syntaxique modulaire, dynamique et autogène en programmation logique*, Rapport de stage du MMFAI de l'ENS Ulm Paris, INRIA, September 2007.

[31] J. PUISSEGUR. *Un système de contraintes pour les motifs biochimiques*, Rapport de stage de mastère de l'ENS Lyon, INRIA, September 2007.

Miscellaneous

[32] P. DERANSART, M. DUCASSÉ, G. FERRAND. *Observing Traces: an Example*, Document interne, 2007.

[33] F. FAGES. *Formal Verification and Inference of Biochemical Reaction Models (invited talk)*, in "Workshop Towards Systems Biology, Verimag, Grenoble", October 2007.

[34] F. FAGES. *Formal Verification and Inference of Biochemical Reaction Models (invited talk)*, in "Bioinformatics and Systems Biology Workshop, co-located with FSTTCS'07, New Delhi, India", December 2007.

[35] F. FAGES. *Formal Verification of Biochemical Models and Application to Cell Cycle (invited talk)*, in "Premier Colloque IXXI: Modèles Mathématiques en Biologie et en Médecine, Lyon, France", November 2007.

[36] F. FAGES. *Formal Verification of Dynamical Models and Application to Cell Cycle Control (invited lecture)*, in "Spring School on Dynamical Modeling of Biological Regulatory Networks, Les Houches, France", April 2007.

[37] F. FAGES. *Semantical and algorithmic aspects of the living*, in "Séminaire Modèles et Algorithmes, INRIA Paris-Rocquencourt", September 2007.

[38] S. SOLIMAN. *Langages formels pour la biologie systémique dans la machine abstraite biochimique BIOCHAM*, October 2007, http://contraintes.inria.fr/~soliman/publi/AcademieDesSciences071009.pdf, Avancées en Sciences de l'Information.