INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Modelling molecular networks: relationships between different formalisms and levels of details

Sylvain Soliman — Claudine Chaouiya — Grégory Batt — François Fages — Élisabeth Remy — Franck Pommereau — Laurence Calzone

N° 7221

February 2010

ISSN 0249-6399 ISRN INRIA/RR--7221--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
PARIS - ROCQUENCOURT

Modelling molecular networks: relationships between different formalisms and levels of details

Sylvain Soliman* , Claudine Chaouiya† , Grégory Batt , François Fages , Élisabeth Remy‡ , Franck Pommereau§ , Laurence Calzone¶

Thème : Programmation, vérification et preuves
Équipe-Projet Contraintes

Rapport de recherche n° 7221 — February 2010 — 23 pages

Abstract: This document is the deliverable 1.3 of French ANR CALAMAR. It presents a study of different formalisms used for modelling and analyzing large molecular regulation networks, their formal links, in terms of mutual encodings and of abstractions, and the corresponding levels of detail captured.

Key-words: regulation networks, expressivity, abstraction, formalisms, modelling

ANR CALAMAR ANR-08-SYSC-003

Compositional modelling and Analysis of LArge MoleculAr Regulatory networks

Deliverable 1.3

* Corresponding author, contact Sylvain.Soliman@inria.fr
† TAGC U928 Inserm, Campus de Luminy, Marseille
‡ IML, Campus de Luminy, Marseille
§ LACL, Université de Paris XII
¶ Insitut Marie Curie, Paris

Centre de recherche INRIA Paris – Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

Modélisation de réseaux moléculaires : relations entre différents formalismes et niveaux de détail

Résumé : Ce document est le livrable 1.3 de l'ANR CALAMAR. Il présente une étude de différents formalismes utilisés pour la modélisation et l'analyse de grands réseaux de régulation, leurs liens formels, aussi bien en termes de conversion mutuelle que d'abstraction, et les niveaux de détails correspondants ainsi traités.

Mots-clés : réseaux de régulation, expressivité, abstraction, formalismes, modélisation

Contents

1 Introduction

With the recent advent of Systems Biology has come a wide variety of modelling formalisms for molecular and gene regulation networks. In CALAMAR we tackle the difficult task of reasoning on very large models (i.e., genome-scale) and thus require powerful and efficient analysis tools relying on some of these formalisms. Moreover, we need to be able to gather the results of different analyses and to relate the information obtained from different versions of the same model. It is thus a natural preliminary step to try and review the main formalisms used in the project and to relate them through formal links when possible.

As a first remark, we would like to point out that a formalism, or formal language, defines a syntax and a set of semantics that in our context are related to dynamics. This is thus how we shall introduce each of the following modelling languages. Moreover, each formalism brings some analysis algorithms and tools.

Note however, that it is not an easy task to characterize these formalisms into a precise classification. Indeed a single formalism often allows different interpretations, different views of a single model. Take the example of a piecewise affine differential equation system, one can either see it as an assembly of of specific ODE models and use corresponding analyses (e.g., bifurcation), or take into account the discrete nature implied by its specific shape, and resort to model-checking for instance. Thus, this report aims also at reviewing the different possible views of each formalism.

Reactions vs. regulations As far as the core syntax is concerned, there is a first split between models centered around the notion of reactions and those focused on regulations (or influences).

In the first category are all process calculi that emerged from the pioneering use of π-calculus in [41]. They focus on (possible) interaction and are compound-centered. None of the partners in CALAMAR uses such formalisms. They currently only provide very few specific analysis tools (mostly stochastic simulation, for which they are quite efficient, and stochastic model-checking, which remains computationally very expensive), but can be mapped to ODE systems. We will not present them in any more details, but see [27] for a review. The second group of formalisms of the first category (reaction-centered) is that of rule-based languages. It includes SBML [32] and BIOCHAM [5]. We will show in section 2.4.1 that these formalisms relate to a bipartite multigraph, i.e., a Petri Net (PN), with compounds and reactions, the latter corresponding to consumption and production of the former. See also [31] for a review of rule-based modelling frameworks for Systems Biology.

The other category of formalisms is the one focusing on Gene Regulation Networks (GRNs). We will detail later the differences between purely Boolean and multi-level approaches, but the contrast with the previous group comes from the fact that the underlying graph is usually not bipartite anymore (though transformations do exist): all vertices are compounds, with an associated local change function, and edges represent regulations (usually classified as either positive or negative).

Discrete vs. continuous The second broad classification lies in the semantics of those different formalisms.

In general the main categories of semantics are either discrete: compounds have a limited number of possible levels, like active/inactive, present/absent, possibly an integer number of molecules when handling a stochastic model; or continuous, representing the idealized case of concentrations in a well-stirred environment.

Note that a discrete semantics for compounds does not necessarily imply a logical time: Continuous Time Markov Chains (CTMCs) are a well known, and correct with respect to the Chemical Master Equation [24], way to model the evolution of finite number of molecules involved in biochemical reactions.

Discrete semantics provide qualitative information when quantitative experimental data is missing, whereas continuous (or in some cases stochastic) semantics are better suited to quantitative analyses. In CALAMAR, qualitative models are the only meaningful ones when large networks are involved, but quantitative models are studied for reduced versions of these networks.

This report is organized as follows. Section 2 reviews the main formalisms we deal with in the context of the CALAMAR project. Section 3 intends to clarify the possible links between these formalisms, providing, in particular, a table (Table 1) and a figure (Figure 8) that summarize their discrete/continuous aspects. Finally, we end with some conclusions and prospects.

2 Modelling formalisms

2.1 Reaction models

2.1.1 SBML

SMBL [32] is an XML-based exchange format that has become the de-facto standard since its adoption by the Nature Publishing Group. As its web page shows, the format is constantly evolving, however its core lies in representing a model by a list of reactions with compounds (reactants, products and modifiers) given with their stoichiometry and an optional kinetic law providing a reaction rate.

As detailed in section 2.4.1, SBML basic models can easily be seen as Petri Nets (that might associate also a kinetic expression to transitions, allowing stochastic or continuous simulations).

One of the starting models for CALAMAR has actually been developed as an SBML model, using the CellDesigner tool. It represents a comprehensive map of the RB/E2F pathway [6] shown in Figure 1.

2.1.2 BIOCHAM

BIOCHAM (the BIOCHemical Abstract Machine) [21, 5, 43] is a software environment for modelling biochemical systems. It is based on two aspects:

1. the formalization of biological properties in temporal logic.
2. the analysis and simulation of rule-based Boolean, kinetic and stochastic models

The first aspect relies on the formalism of Temporal Logics described in section 3.1. We will here focus on the second aspect, which is quite close to SBML and related to PNs.

BIOCHAM provides tools and languages for describing protein networks with a simple and straightforward syntax, and for integrating biological properties into the model. It then becomes possible to analyze, query, verify, and maintain the model with respect to those properties. For kinetic models, BIOCHAM can use an evolutionary algorithm for searching the values of (several tenths of) parameters in order to reproduce a specific behavior observed in experiments and formalized in temporal logic. Coupled with other methods such as bifurcation diagrams, this search assists the modeller/biologist in the modelling process [35].

A model is defined by a set of reaction rules, possibly equipped with kinetic expressions, a list of parameter values and initial conditions. A specification that accounts for the relevant biological properties can also be added to the model as a list of temporal logic formulae. A single BIOCHAM file can be used for Boolean, continuous or stochastic analyses. According to the type of study chosen by the user, the model receives different interpretations, e.g. the kinetic expressions are respectively ignored, seen as reaction rates or interaction probabilities.

A network is thus modelled by a list of biochemical reaction rules such as: `CycB + CDK => CycB-CDK` where `CycB` and `CDK` are two proteins and `CycB-CDK`

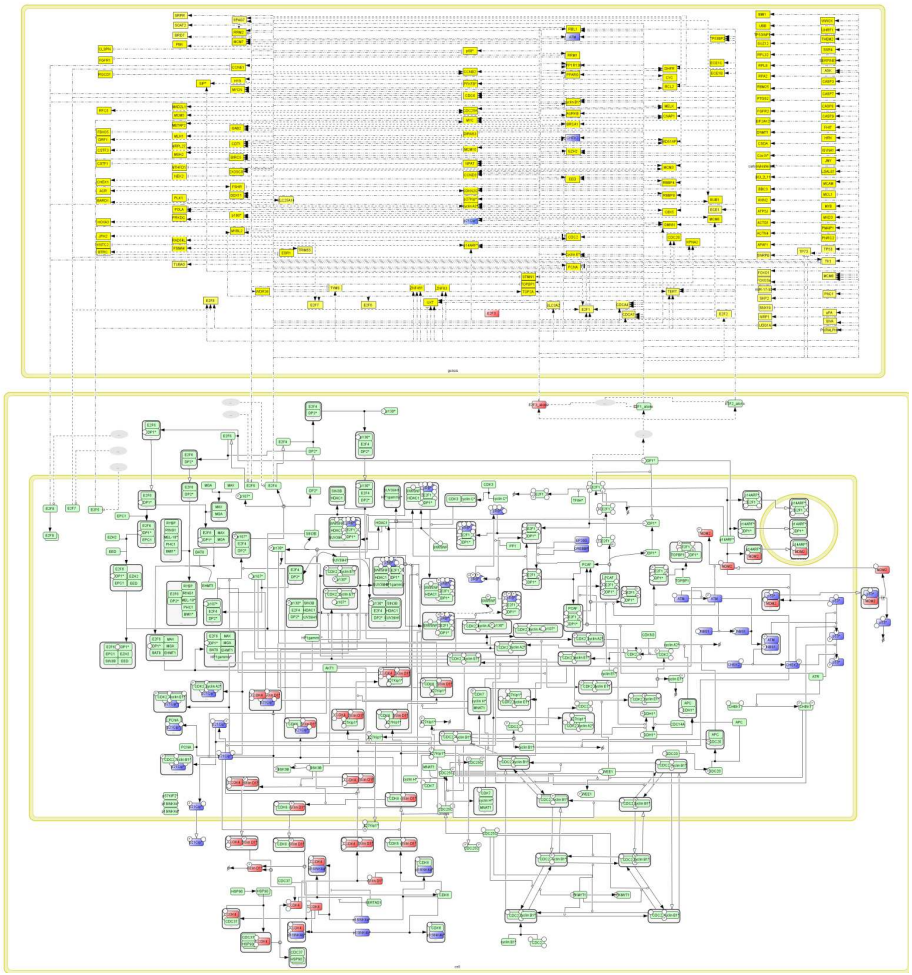


Figure 1: Comprehensive map of the RB/E2F pathway of [6]

is their complex. The locations of the interactions can also be explicitly specified by compartment names such as the nucleus, the cytoplasm, etc.:

`CycB::cyto + CDK::cyto => CycB-CDK::cyto`,

or in a transport rule: `CycB-CDK::cyto => CycB-CDK::nucleus`.
A kinetic expression can be attached to a reaction rule, as follows:

```
k*[CycB]*[CDK] for CycB + CDK => CycB-CDK
```

As mentioned above, this expression is ignored in the Boolean view of the model, while in the continuous interpretation, it is derived as a term in the differential equations of the reactants and products, such as: $d(\text{CycB-CDK})/dt = k * [\text{CycB}] * [\text{CDK}]$. The whole system of ODEs is thus automatically generated from the set of reaction rules. In the stochastic view, the kinetic expressions are interpreted as transition probabilities.

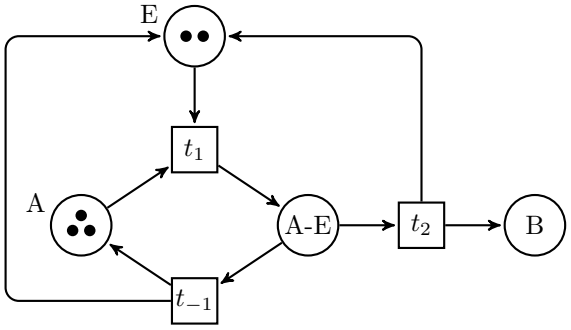


Figure 2: Biochemical model of example 1, represented as a PN with a marking enabling t_1

Example 1 *For instance the enzymatic reaction written (in BIOCHAM-like syntax),* `A + E <=> A-E => B + E` *corresponds to the Petri Net depicted in Figure 2.*

BIOCHAM has several possible semantics, Boolean, discrete/stochastic and continuous. Section 3.2 describes their relationships. In a discrete view of this model, starting from a marking with at least one token in A and in E, one can remove one of each to produce one token in A-E (firing of t_1) and then either remove it to add again one token to A and one to E (firing of t_{-1}), or to add one to B and one to E (firing of t_2).

The different semantics allow simulation of a model's behavior, analysis through Model-Checking but also structural study of the underlying PN (semi-flow computation providing elementary modes or conservation laws for instance). The discrete and Boolean semantics are especially important for large models where the kinetic data of reactions is only partially available.

It is worth noting that BIOCHAM can export its continuous semantics to ODE models, but that under restrictions the reverse is also true (see Section 3.3), and that an influence network related to the (symbolic) Jacobian of the ODE system can also be extracted in linear time (see Section 3.2), providing a link from ODEs to reaction models and then to regulation networks.

2.2 Logical regulatory networks

The logical formalism applies to Gene Regulatory Networks, in that it describes influences between regulatory products in an abstract, discrete way. Already in the sixties, Sugita [52], Kauffman [33] and others proposed a qualitative representation of regulatory networks. Here, we rely on the asynchronous, generalized logical formalization introduced by R. Thomas and colleagues [55, 57, 54]. In this framework, regulatory networks are represented by Logical Regulatory Graphs (LRG) which include:

- nodes that represent regulatory components (be it genes, proteins or even phenomenological components such as cell mass involved in the cell cycle control) are associated with discrete variables accounting for their functional levels;

- arcs that represent regulatory interactions between components, each arc being associated to the threshold above which the source node exerts its influence on the target node. These arcs are often labelled with a sign, positive in the case of an activation, negative in the case of an inhibition;
- logical parameters that specify the target levels of the nodes when subject to regulatory interactions (for each node, as many parameters as the number of possible interaction combinations). These sets of logical parameters can equivalently be defined by truth tables, logical or evolution functions, decision diagrams (e.g. [10, 38]).

This qualitative level of description is very well adapted to regulatory networks for which precise, quantitative data is scarcely available.

Given a state of an LRG, that is a vector of node levels, the logical parameters determine which nodes are called to change their levels. Here, the choice of the updating scheme is crucial since it will lead to different behaviours. The synchronous updating performs all changes simultaneously, hence a state has at most one subsequent state. In contrast, in the asynchronous updating, a unique change is performed at each step, hence each state has as many potentially subsequent states as the number of nodes that are called to change [56]. These discrete dynamics are generally represented by a State Transition Graph (STG).

Since STG sizes exponentially increase with the number of regulatory components, methods relating structural properties of the regulatory network to its dynamical properties are of interest. Regulatory circuits are known to play a crucial dynamical role: positive circuits (encompassing an even number of inhibitions) are necessary for multistationarity, whereas negative circuits (encompassing an odd number of inhibitions) are required for oscillatory behaviours [53, 57]. Moreover, circuit analysis provides a valuable tool for conducting model reductions [37] and could be useful as well for identifying functional modules in regulatory networks.

2.2.1 Boolean framework

The approximation of genetic regulation by such discrete modelling was justified by the well-known threshold-effect which occurs in the genetic regulation (the curve representing the effect of a regulator on its target is a sigmoid of high degree).

In the Boolean case, each regulatory node is associated to a Boolean variable, which can be either 1 (meaning, e.g., that the gene is expressed or the produced protein is active) or 0 (meaning, e.g., that the gene is not expressed or the protein is inactive). Another way of interpreting these values is that the component's current level (of expression, concentration or activity) is above (for value 1) or below (for value 0) its functional threshold (associated to the interactions it exerts on its targets). Notably, there is a wealth of literature on dynamical properties of Random Boolean networks as introduced by S. Kauffman [33]. In these models, the wiring is random (though fixing the number of incoming interactions for all nodes), the logical functions are also randomly chosen and the updating is performed synchronously (see e.g. [19] for a review). Studies of these models mainly relate to the number and lengths of the attractors.

In the context of Boolean Regulatory Graph as defined by R. Thomas, a number of formal results concerning regulatory circuit roles has been published,

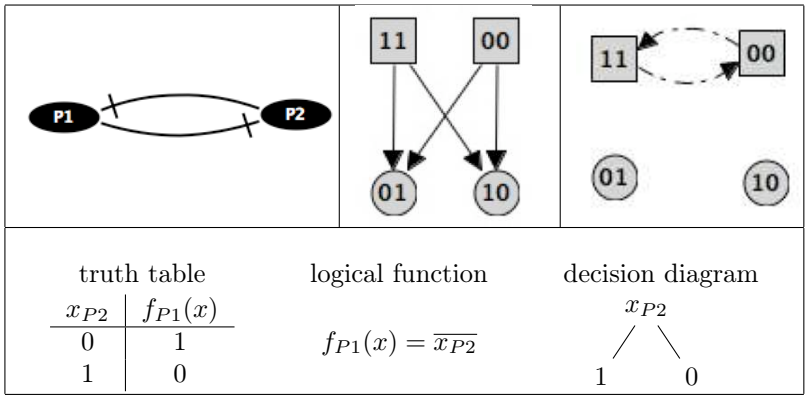


Figure 3: The logical model of the toggle-switch. The regulatory graph is displayed on top-left box. The blunt arrows denote inhibitory interactions. The asynchronous (resp. synchronous) state transition graph is given in the top-middle (resp. top-right) box. Plain arcs denote transitions encompassing a unique change, whereas dotted arcs denote multiple simultaneous changes. States show the current levels of P1 and P2. Note that the attractors are different for these updating schemes. In the asynchronous case, there are two stable states, while in the synchronous dynamics there is, in addition to the stable states, a stable cycle. The bottom box gives the logical rule attached to the node P1, in different equivalent forms. The current level of P2 is denoted x_{P2}.

proving the necessary conditions evoked above. The converse of these rules is not true in general (see [16]). Anyway, we can provide sufficient conditions for multistationarity or sustained oscillations in some specific cases, typically when we enforce the system to remain in the context of functionality of the circuit [42].

2.2.2 Multi-level logical framework

In many cases, Boolean variables are sufficient to convey the role of regulatory components, but this all-or-none description must be extended when, for example, a component does not act at similar levels on various targets, or when its effect on a target differs depending on its functional level. To take into account such situations, multivalued variables were introduced [56]. It should be noted however that increasing the range of values taken by the component levels increases the complexity of the model and hence its analysis.

2.3 Ordinary Differential Equations (ODEs)

ODEs are a fundamental modelling tool for mathematical biology and chemistry [23]. When precise kinetic data is available, it allows simulation via numerical integration, and bifurcation analysis in low-dimensional models when one or two parameters are unknown.

In the context of CALAMAR, large models are not amenable to ODE models, since most of the quantitative data is lacking. However, since time courses are a very valuable means of interaction with biologists, it is crucial to relate formally

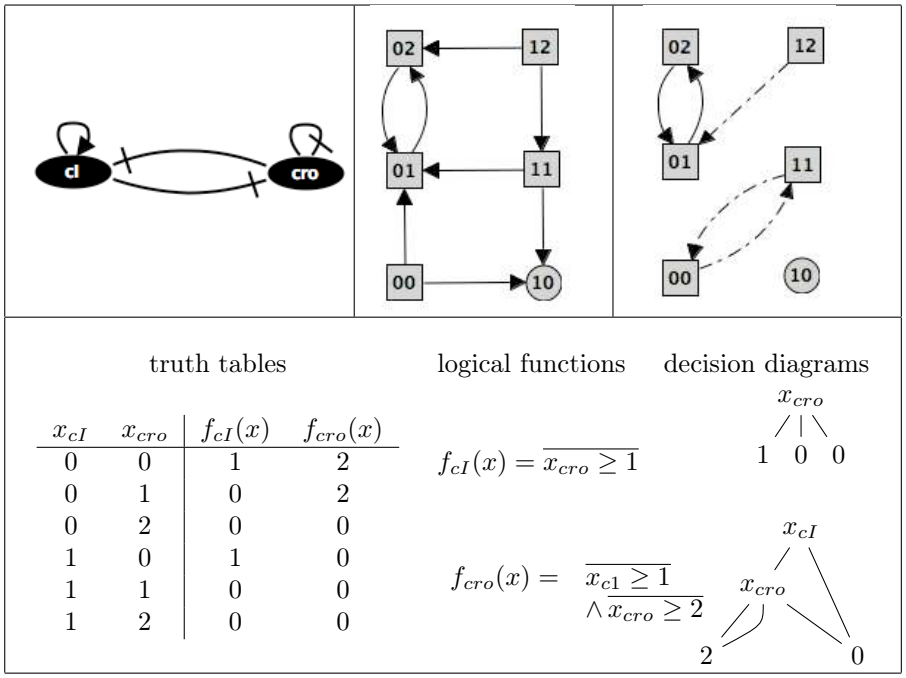


Figure 4: The logical model of the bacteriophage lambda decision switch as defined in [54]. The core of the lambda regulatory network displayed in the top-left box involves the two main cross-regulating genes: cI coding for lambda repressor, the only gene expressed in the lysogenic state and cro, coding for another repressor, expressed in the lytic cycle. The asynchronous (resp. synchronous) state transition graph is given in the top-middle (resp. top-right) box. States display the current levels of cI and cro, in this order. In the asynchronous case, there are two attractors: one stable lysogenic stable and a cycle denoting and homeostatic level of cro, corresponding to the lysis. The bottom box gives the logical functions for cI and cro in different equivalent forms.

big qualitative models to small quantitative models - as shown in section 3 - and ODEs to structured models - as shown in section 3.3.

Piecewise Affine Differential Equation (PADE) models have been proposed by Glass and Kauffman in the seventies to model genetic regulatory networks [25]. They are specific types of ODE models in which step functions are used to describe the influence of transcription factors on gene expression. Step functions are introduced as a simplification of the Hill functions often seen in ODE models of gene networks.

One major attractive feature of this formalism is that it is possible to obtain a description of the dynamics of gene networks in the form of a state transition graph, simply using qualitative information on biological parameters [15]. In this respect, this formalism is closely related to the (generalized) logical formalism of Section 2.2. A significant difference however is that the dynamics of the generalized logical models is defined at the discrete level, whereas that of PADE models is defined at the continuous level.

2.4 Petri Nets

The use of Petri nets to model biochemical networks is quite old [40] but was mostly limited to metabolic pathway representation and analysis. In recent years however, they have become a more and more widely used formalism in the Systems Biology community and for a variety of biological systems since they provide, for instance, a natural framework both for reaction models and for logical regulatory graphs (see below). This allows, amongst other things, common analysis through model-checking as detailed in Section 3.1.

2.4.1 Standard PNs

A Petri net is a bipartite oriented multigraph of transitions, usually represented as square boxes, and places, usually represented as circles, that defines a transition relation on *markings* of the net, i.e., multisets of tokens associated to places. The relation is defined by *firings* of transitions, i.e., when there are tokens (as many as the weights of the incoming arcs) in all pre-places of a transition, they can be consumed and as many tokens as the weights on the outgoing arcs are added to each post-place.

Reaction models can usually be easily represented as PNs by mapping compounds to places and reactions to transitions (stoichiometry corresponding to the weights of the arc between the places and the transition), see Figure 1.

2.4.2 Colored and High-level PNs

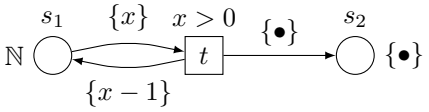
A colored Petri net involves values, variables and expressions. These objects are defined by a *color domain* that provides data values, variables, operators, a syntax for expressions, possibly typing rules, etc. For instance, one may use integer arithmetic or Boolean logic as color domains. Usually, more elaborated color domains are useful to ease modelling. In particular, one may consider a functional programming language or the functional fragment (expressions) of an imperative programming language.

A *colored Petri net* is a tuple (S, T, ℓ) where:

- S is the finite set of *places*;
- T, disjoint from S, is the finite set of *transitions*;
- ℓ is a *labelling* function such that:
  - for all $s \in S$, $\ell(s) \subseteq \mathbb{D}$ is the *type* of s, i.e., the set of values that s is allowed to carry,
  - for all $t \in T$, $\ell(t) \in \mathbb{E}$ is the *guard* of t, i.e., a condition for its execution,
  - for all $(x, y) \in (S \times T) \cup (T \times S)$, $\ell(x, y)$ is a multiset over $\mathbb{E}$ and defines the *arc* from x toward y.

Conventions to depict colored PN are similar to those adopted for standard PN (see Figure 5 for a colored PN represented in both textual and graphical notations). +

Let $N \stackrel{\text{df}}{=} (S, T, \ell)$ be a colored PN. A *marking* M of N is a function on S that maps each place s to a finite multiset over $\ell(s)$ representing the tokens held



$$
\begin{array}{lcl}
S & \stackrel{\text{df}}{=} & \{s_1, s_2\} \\
T & \stackrel{\text{df}}{=} & \{t\} \\
\ell & \stackrel{\text{df}}{=} & \{s_1 \mapsto \mathbb{N}, s_2 \mapsto \{\bullet\}, t \mapsto x > 0, (s_1, t) \mapsto \{x\}, \\
 & & (s_2, t) \mapsto \emptyset, (t, s_1) \mapsto \{x-1\}, (t, s_2) \mapsto \{\bullet\}\}
\end{array}
$$

Figure 5: A simple colored PN.

by s. A transition $t \in T$ is *enabled* at a marking M and a binding β, which is denoted by $M[t, \beta\rangle$, iff the following conditions hold:

- M has enough tokens, i.e., for all $s \in S$, $\beta(\ell(s,t)) \leq M(s)$;
- the guard is satisfied, i.e., $\beta(\ell(t)) = \mathit{true}$;
- place types are respected, i.e., for all $s \in S$, $\beta(\ell(t,s))$ is a multiset over $\ell(s)$.

We make no assumption about the typing or syntactical correctness of values or expressions; instead, we assume that any expression can be evaluated, possibly to $\perp$ (undefined). More precisely, a *binding* is a partial function $\beta : \mathbb{V} \to \mathbb{D}$. Let $e \in \mathbb{E}$ and β be a binding, we denote by $\beta(e)$ the evaluation of e under β; if the domain of β does not include all the variables involved in e, denoted by $\mathit{vars}(e)$, then $\beta(e) \stackrel{\text{df}}{=} \perp$. The application of a binding to evaluate an expression is naturally extended to sets and multisets of expressions.

If $t \in T$ is enabled at marking M and binding β, then t may *fire* and yield a marking M' defined for all $s \in S$ as $M'(s) \stackrel{\text{df}}{=} M(s) - \beta(\ell(s,t)) + \beta(\ell(t,s))$.

We have recently defined *High-level Petri Nets* as colored PNs equipped with place names and an associated composition operation. This last feature is of a particular interest in the context of the CALAMAR project to handle the compositional building of large biochemical models [4].

2.4.3 Further Petri Nets extensions

A number of extensions have been designed to increase the expressive power of Petri nets. It is not our purpose here to review all these extensions but rather to mention those that are actually used to model biochemical networks.

In continuous Petri Nets [34], places are assigned a continuous marking (real positive number). Such a marking typically carries the concentrations of the compounds (denoted by the places). Each transition is associated with a firing rate function, accounting for concentration-dependent chemical reaction rates, which depends only on its pre-places (and possibly their current markings). Note that this restriction might forbid some reaction models with (quite specific and not biochemically meaningful) kinetic expressions to be mapped to continuous PNs. This is the case of kinetics with explicit inhibitors like $1/(K + x^n)$. With

this semantics, CPNs can easily be mapped to ODEs. We develop the reverse mapping from ODEs to CPNs in section 3.3.

In stochastic Petri nets, the marking of the places remains discrete (natural numbers of tokens representing molecules) and the transitions have an associated delay, which is a random variable. There are different classes of stochastic PNs, depending on the type of the involved distributions. Most models of biochemical networks are equivalent to Continuous Time Markov Chains and thus can be simulated using Gillespie's algorithm (see e.g. [51]).

In hybrid functional Petri nets, there are both discrete and continuous places and transitions. Moreover, the arc weights might vary with the marking of the places. This most extended version of PN can be very convenient to represent some biological features (see e.g. [18]). This high level of expressiveness comes at the cost of few analytical means.

For an overview of the various classes of PN employed in the field of biochemical network modelling, see [29, 8].

2.5 Others

Between purely qualitative and highly non-linear ODE systems lie many different formalisms. We refer to [39] for a comparison of such formalisms, from Hill-based equations, to piecewise-affine with or without quasi-steady-state assumptions. On the other side, models taking into account the stochasticity implied by a low number of molecules also exist. They can be simple stochastic views of a continuous model, or involve multiplicative or additive stochastic noise. This level of detail, however, is out of the scope of CALAMAR, since it does not allow to cope with large models.

3 Links between formalisms

We recall here some of the links already described in the previous sections and add some specific relationships between all discrete formalisms in 3.1, discrete and continuous reaction models or continuous reaction models and regulatory graphs in 3.2 and finally pure ODEs and reaction models in 3.3.

3.1 Specification formalisms

In order to apply some of the analyses described above, and mainly those centered around Model-Checking, another formalism comes into the picture: the one that allows the encoding of the specification to be verified, i.e., the experimental data or hypothesized behavior that the model should reproduce. Note that this is quite independent of the formalism chosen to model the system, and thus provides yet another link between the different formalisms.

The main formalism used in this domain is that of Temporal Logics.

The *Computation Tree Logic* CTL* [12] is an extension of classical logic that allows reasoning about an infinite tree of state transitions. It uses operators about branches (non-deterministic choices) and time (state transitions). Two path quantifiers **A** and **E** are thus introduced to handle non-determinism: $\mathbf{A}\phi$ meaning that ϕ is true on all branches, and $\mathbf{E}\phi$ that it is true on at least one branch. The time operators are $\mathbf{F}, \mathbf{G}, \mathbf{X}, \mathbf{U}$ and $\mathbf{W}$; $\mathbf{X}\phi$ meaning ϕ is true at

the next transition, $\mathbf{G}\phi$ that ϕ is always true, $F\phi$ that ϕ is eventually true, $\phi\mathbf{U}\psi$ meaning ϕ is always true until ψ becomes true, and $\phi\mathbf{W}\psi$ meaning ϕ is always true until ψ might become true. In this logic, $\mathbf{F}\phi$ is equivalent to $true\mathbf{U}\phi$, $\phi\mathbf{W}\psi$ to $(\phi\mathbf{U}\psi)\vee\mathbf{G}\phi$, and the following duality properties hold: $\neg(\mathbf{EF}(\phi)) = \mathbf{AG}(\neg\phi)$, $\neg(\mathbf{E}\ \phi\mathbf{U}\psi) = \mathbf{A}(\neg\psi\mathbf{W}\neg\phi)$ and $\neg(\mathbf{E}\phi\mathbf{W}\psi) = \mathbf{A}(\neg\psi\mathbf{U}\neg\phi)$, where $\neg$ denotes negation.

We refer to [7, 22, 1, 2, 3, 5] for examples of biological properties of a system expressed by Temporal Logic formulae.

In BIOCHAM, the CTL fragment of CTL* is used for the (non-deterministic) Boolean semantics: each temporal operator must be preceded by a path operator, and each path operator has to be immediately followed by a temporal operator. In that case, atomic formulae α denote usually presence/absence or activity/inactivity of some compounds. The LTL fragment is used for linear Kripke structures corresponding to continuous or stochastic semantics: only time operators are allowed. In this case, atomic formulae are extended to linear inequalities between (continuous or discrete) values of compounds, i.e., concentrations (and their derivatives) or numbers of molecules; on the other hand, since theses values come from experimental or in-silico experiments, reasoning about finite traces might call for an adaptation of usual LTL semantics, see [43] for a discussion.

In most of the cases, if there is an underlying discrete state-space, CTL* will allow the modeller to state some properties about the experimental/observed/expected behavior of the system, whatever the formalism used for it.

This is related to the fact that, as explained in Section 2.4, Petri Nets provide a natural framework for both reaction models and logical regulatory graphs, which allows their use as a common language that can gather different kinds of analyses, like model-checking, within a single formalism (see for instance [30]).

3.2 Hierarchy of abstractions

In [20] we exhibited a formal hierarchy of abstractions, based on the framework of Abstract Interpretation [13], that relates Boolean, stochastic, discrete and continuous semantics.

We first show how these different semantics can be formally related by simple Galois connections, as required in the theory of abstract interpretation, with the noticeable exception of the differential semantics that are linked through limit operations instead of simple algebraic operators.

We also describe a type system allowing the automatic derivation of a regulation network corresponding to that of the Jacobian matrix (as used for instance in [26, 48, 50]) even when precise quantitative information is lacking, provided that the kinetic laws satisfy some general properties satisfied by the usual kinetics (Mass Action, Michaelis Menten, Hill, etc.).

As explained in Section 2.1, these links provide a way to relate ODE systems to structured model, (using the results of Section 3.3) and then to regulation networks.

3.3 Structure out of ODEs

The continuous PNs as mentioned in section 2.4.3 defines a system of ODEs. Since there is structural information in the continuous PNs that disappears in

ODEs, in general the reverse mapping is not unique. However we already obtained results providing conditions and algorithms to map back an ODE system to a CPN, such that the mapping is unique and biologically relevant.

A system of ODEs is an unstructured model, even if it usually represents information that was graphical in the mind of the biologist/modeller.

Since there is structural information in the CPNs that disappears in ODEs, in general the reverse mapping is not unique. Conditions and algorithms have been proposed about how to map back an ODE system to a CPN, so that the mapping is unique and biologically relevant. In [49] we provide sufficient conditions to recover in a unique way a structured model, like a continuous PN, from a given system of ODEs. These conditions mostly restrict the type of kinetics allowed for each reaction to either pure Mass-Action Law or Michaelis-Menten. Actually for most of mechanistic biochemical models, these (mathematical) restrictions make sense even for modellers: for instance, they also correspond to what is usually needed to allow stochastic simulation.

Under these conditions, the structure of the model, which can be represented by a Petri net, can be uncovered from a system of ODEs, thus making equivalent the two formalisms. Going one step further, this allows the link to regulation networks through the results of Section 3.2.

3.4 Logical regulatory graphs and Petri nets

Logical Regulatory Graphs can also be represented by means of Petri nets, although the translation is rather tricky compared to the intuitive PN representation of reaction networks [9]. Briefly, in this PN representation, pairs of complementary places account for regulatory nodes and transitions account for regulatory effects by specific interaction combinations.

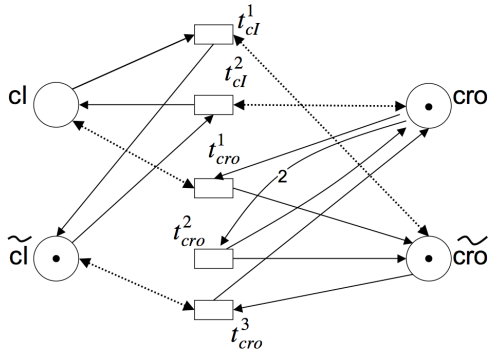


Figure 6: Standard PN representation of the bacteriophage lambda logical regulatory graph of Figure 4. For each component, there are two complementary places: cI (whose marking represents the current level x_{cI} of cI) and the complementary place $\widetilde{cI}$ (whose marking is $Max_{cI} - x_{cI}$). The current marking $M(cI, \widetilde{cI}, cro, \widetilde{cro}) = (0, 1, 1, 1)$ depicted here corresponds to the state $x_{cI} = 0$ and $x_{cro} = 1$; it enables transition t^3_{cro} whose firing leads to the new marking $(0, 1, 2, 0)$, which in turn enables t^2_{cro}.

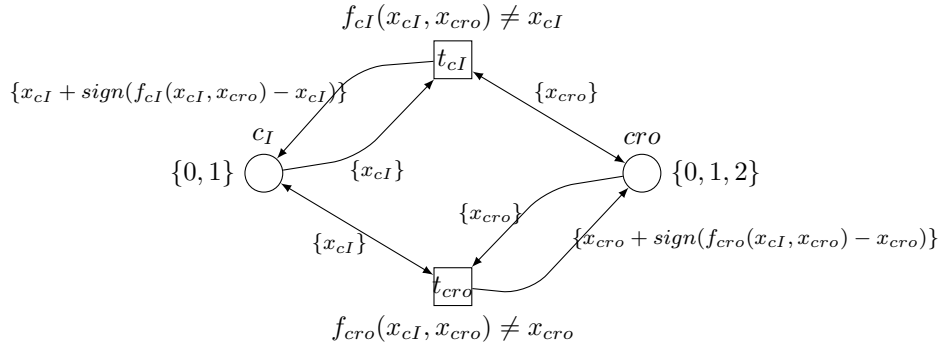


Figure 7: A colored Petri net representation of the phage lambda switch model as defined in Figure 4 Figure 5.

4 General overview

A fine understanding of the links between the different types of formalisms is important for different reasons, besides the intrinsic mathematical interest. The transition between discrete and continuous - through all intermediate - formalisms should allow the "transfer of information" from one formalism to another, and then help to refine the models. The Figure 8 and Table 1 provide a summary.

Formalism	Variables		Time	
	Discrete	Continuous	Discrete	Continuous
ODE		yes	(steps of integration)	yes
Logical	bool. or multi-level		logical	
Petri Nets	standard PNs	CPNs	logical	CPNs
BIOCHAM	bool. or multi (stoch.)	ODEs	logical (bool. only)	ODEs

Table 1: A summary of the characteristics of different formalisms

So far, most biological data were of qualitative nature, justifying the construction of models based on discrete formalisms that allow qualitative analyses of the considered networks. Since it is preferable to handle models which are coherent with the nature of the data, the recent influx of quantitative data encourages the modeller to consider continuous formalisms. The difficulties then are to select the Ordinary Differential Equations which better fit the system under consideration, and to estimate their parameters. Having access to the corresponding discrete models, and having clearly established the links between discrete and continuous formalisms would permit us to consider discrete modelling as a helpful intermediate step towards, for example, the identification of the corresponding class of ODE.

The transition from continuous to discrete can also be informative. For example, as it has been emphasized above in this report, the choice of the update

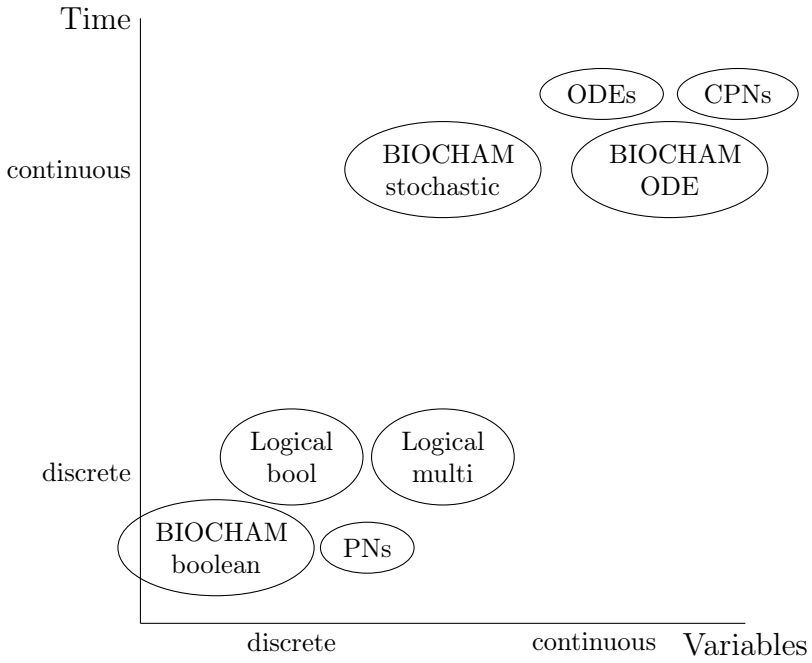


Figure 8: Plot of the discrete/continuous characteristics of different formalisms. PADEs can be seen either as ODEs or as Logical models

scheme in discrete models has an important effect on the dynamics and can be a difficult point of the model setup. A precise relation with continuous-time models would allow us to select the most realistic class of update schemes. On the same topic, there is more and more research on the influence of the structure of reaction models on their dynamics, either based on Petri Nets (invariant-based properties for instance) or on other tools (e.g., Chemical Reaction Network Theory, Abstract Interpretation as in Section 3.2, etc.).

It is not our aim here to detail these comparisons. Different papers deal with the relationships between formalisms; in general they focus on a particular transition from a formalism to another, and on the conservation of some dynamical properties. Figure 9 provides a non-exhaustive list of such works on gene regulatory models. Although this scheme might appear almost complete, it should be noted here that there is still a lot to do, as a number of results are either not formally proved, or only apply in restricted conditions.

Recently we have formally proved that the multivalued to Boolean mapping proposed years ago by Van Ham, Snoussi and Thomas is indeed the sole mapping that could preserve both the regulatory structures and the dynamical behaviours [28, 47, 17]. With this preliminary work, existing methods that apply so far on Boolean models will be extended to the multi-level case.

The picture is both simpler if one considers simple reaction models and more tricky if one wants to include ODE models, which are quite often what the modellers have in mind even when they use a structured model. Once again, we have already given some pointers to precise articles since an exhaustive

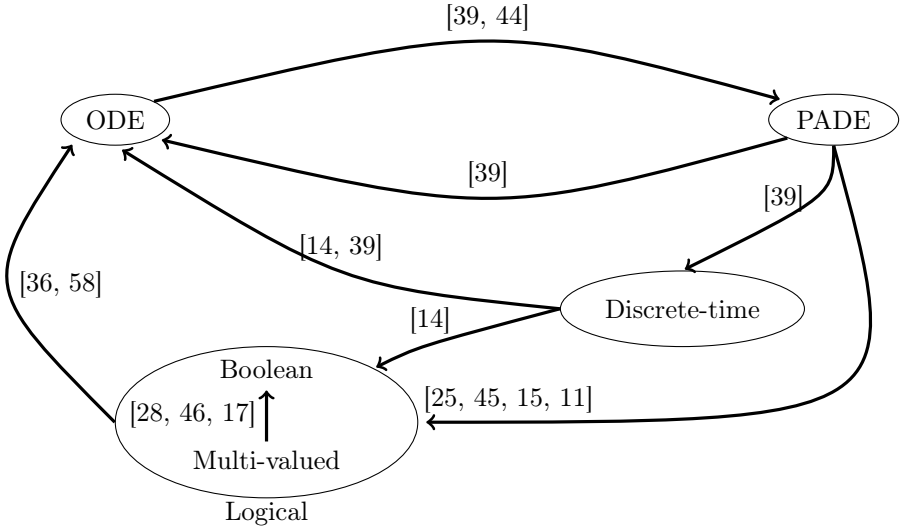


Figure 9: A tentative scheme illustrating existing links between the different formalisms employed for the modelling of regulatory networks. The arrows indicate studies, which compare dynamical properties, propose specific convenient mappings or even formally define mappings. It should be noted that most of these studies often apply to restricted classes of models, within the broader class defined by the formalism.

description is out of the scope of this report, but Figure 10 recapitulates some of the known abstractions/embeddings/etc.

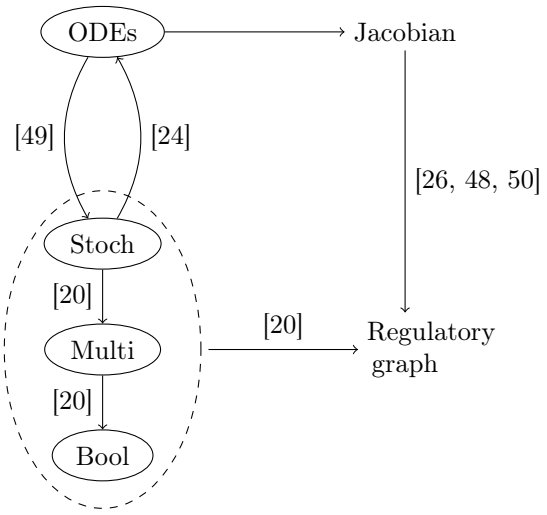


Figure 10: Some of the links for reaction-based modelling formalisms

5 Conclusion

The various graph-theoretic formalisms we have considered in this report reveal the rich structures of biochemical interaction systems. They provide a variety of graph-based analysis tools that are complementary to the classical mathematical tools used for analysing the ODE models in which these structures vanish.

We have given a brief review of different modelling formalisms used in Systems Biology, and more precisely in the framework of the ANR project CALAMAR. Several links have been outlined and will be exploited in our concrete work on the E2F/RB network and in some other models at the core of our study. This report might thus be updated with the forthcoming illustrative examples.

Interestingly, in section 4, we have noticed that some links have been established by some authors in an *empirical* way, meaning that formalization is still needed, e.g. proving the conserved vs lost dynamical properties, or with some strong restrictions that could probably be relaxed to some extent.

In this respect, we believe that the graph-theoretic study of biochemical interaction systems should reconcile the mathematician and the biologist on a common formal ground.

References

[1] M. Antoniotti, A. Policriti, N. Ugel, and B. Mishra. Model building and model checking for biochemical processes. *Cell Biochemistry and Biophysics*, 38:271–286, 2003.

[2] G. Batt, D. Ropers, H. de Jong, J. Geiselmann, R. Mateescu, M. Page, and D. Schneider. Validation of qualitative models of genetic regulatory networks by model checking : Analysis of the nutritional stress response in *Escherichia coli*. *Bioinformatics*, 21(Suppl.1):i19–i28, 2005.

[3] G. Bernot, J.-P. Comet, A. Richard, and J. Guespin. A fruitful application of formal methods to biological regulatory networks: Extending thomas' asynchronous logical approach with temporal logic. *Journal of Theoretical Biology*, 229(3):339–347, 2004.

[4] C. Chaouiya, H. Klaudel, and F. Pommereau. A modular, qualitative modelling of regulatory networks using petri nets. in press.

[5] L. Calzone, F. Fages, and S. Soliman. BIOCHAM: An environment for modeling biological systems and formalizing experimental knowledge. *Bioinformatics*, 22(14):1805–1807, 2006.

[6] L. Calzone, A. Gelay, A. Zinovyev, F. Radvanyi, and E. Barillot. A comprehensive imodular map of molecular interactions in RB/E2F pathway. *Molecular Systems Biology*, 4(173), 2008.

[7] N. Chabrier and F. Fages. Symbolic model checking of biochemical networks. In C. Priami, editor, *CMSB'03: Proceedings of the first workshop on Computational Methods in Systems Biology*, volume 2602 of *Lecture Notes in Computer Science*, pages 149–162, Rovereto, Italy, Mar. 2003. Springer-Verlag.

[8] C. Chaouiya. Petri net modelling of biological networks. *Briefings in Bioinformatics*, 8:210–9, 2007.

[9] C. Chaouiya, A. Naldi, E. Remy, and D. Thieffry. Petri net representation of multi-valued logical regulatory graphs. *Natural Computing*, to appear.

[10] C. Chaouiya, E. Remy, B. Mossé, and D. Thieffry. Qualitative analysis of regulatory graphs: a computational tool based on a discrete formal framework. In *POSTA'03*, volume 294 of *Lecture Notes in Control and Information Sciences (LNCIS)*, pages 119–126, 2003.

[11] M. Chaves, L. Tournier, and J.-L. Gouzé. Comparison between boolean and piecewise affine differential models for genetic networks. Technical report, INRIA, http://hal.archives-ouvertes.fr/inria-00426414/en/, 2009.

[12] E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. MIT Press, 1999.

[13] P. Cousot and R. Cousot. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *POPL'77: Proceedings of the 6th ACM Symposium on Principles of Programming Languages*, pages 238–252, New York, 1977. ACM Press.

[14] R. Coutinho, B. Fernandez, R. Lima, and A. Meyroneinc. Discrete time piecewise affine models of genetic regulatory networks. *J Math Biol*, 52(4):524–70, Apr 2006.

[15] H. De Jong, J.-L. Gouzé, C. Hernandez, M. Page, T. Sari, and J. Geiselmann. Qualitative simulation of genetic regulatory networks using piecewise-linear models. *Bull Math Biol*, 66(2):301–40, Mar 2004.

[16] G. Didier and E. Remy. Relations between gene regulatory networks and cell dynamics in boolean models. Submitted.

[17] G. Didier, E. Remy, and C. Chaouiya. Mapping multivalued onto boolean dynamics. Submitted.

[18] A. Doi, S. Fujita, H. Matsuno, M. Nagasaki, and S. Miyano. Constructing biological pathway models with hybrid functional petri nets. *In Silico Biology*, 4(0013), 2004.

[19] B. Drossel. Random boolean networks. arXiv.org:0706.3351, 2008.

[20] F. Fages and S. Soliman. Abstract interpretation and types for systems biology. *Theoretical Computer Science*, 403(1):52–70, 2008.

[21] F. Fages and S. Soliman. Formal cell biology in BIOCHAM. In M. Bernardo, P. Degano, and G. Zavattaro, editors, *8th Int. School on Formal Methods for the Design of Computer, Communication and Software Systems: Computational Systems Biology SFM'08*, volume 5016 of *Lecture Notes in Computer Science*, pages 54–80, Bertinoro, Italy, Feb. 2008. Springer-Verlag.

[22] F. Fages, S. Soliman, and N. Chabrier-Rivier. Modelling and querying interaction networks in the biochemical abstract machine BIOCHAM. *Journal of Biological Physics and Chemistry*, 4(2):64–73, Oct. 2004.

[23] M. Feinberg. Mathematical aspects of mass action kinetics. In L. Lapidus and N. R. Amundson, editors, *Chemical Reactor Theory: A Review*, chapter 1, pages 1–78. Prentice-Hall, 1977.

[24] D. T. Gillespie. General method for numerically simulating stochastic time evolution of coupled chemical-reactions. *Journal of Computational Physics*, 22:403–434, 1976.

[25] L. Glass and S. Kauffman. The logical analysis of continuous, non-linear biochemical control networks. *J. Theor. Biol.*, 39(1):103–129, 1973.

[26] J.-L. Gouzé. Positive and negative circuits in dynamical systems. *Journal of Biological Systems*, 6:11–15, 1998.

[27] M. L. Guerriero, D. Prandi, C. Priami, and P. Quaglia. Process calculi abstractions for biology. *Algorithmic Bioprocesses*, pages 463–486, 2009.

[28] P. V. Ham. How to deal with variables with more than two levels. In R. Thomas, editor, *Kinetic logic: a Boolean approach to the analysis of complex regulatory systems.*, volume 29. Lecture notes in Biomathematics, 1979.

[29] S. Hardy and P. Robillard. Modeling and simulation of molecular biology using petri nets: modeling goals of various approaches. *Journal of Bioinformatics and Computational Biology*, 2(4):619–37, 2004.

[30] M. Heiner, D. Gilbert, and R. Donaldson. Petri nets for systems and synthetic biology. In *SFM*, pages 215–264, 2008.

[31] W. S. Hlavacek, J. R. Faeder, M. L. Blinov, R. G. Posner, M. Hucka, and W. Fontana. Rules for modeling signal-transduction systems. *Science STKE*, 344:re6, 2006.

[32] M. Hucka et al. The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[33] S. Kauffman. Metabolic stability and epigenesis in randomly constructed genetics nets. *J. Theor. Biol.*, 22:437–67., 1969.

[34] I. Koch and M. Heiner. *Biological Network Analysis*, chapter Petri nets, pages 139–179. Number 7 in Series on Bioinformatics. Wiley, 2008.

[35] E. D. Maria, F. Fages, and S. Soliman. On coupling models using model-checking: Effects of irinotecan injections on the mammalian cell cycle. In *CMSB'09: Proceedings of the seventh international conference on Computational Methods in Systems Biology*, volume 5688 of *Lecture Notes in BioInformatics*, pages 142–157. Springer-Verlag, 2009.

[36] L. Mendoza and I. Xenarios. A method for the generation of standardized qualitative dynamical systems of regulatory networks. *Theor Biol Med Model*, 3:13, 2006.

[37] A. Naldi, R. Remy, D. Thieffry, and C. Chaouiya. A reduction method for logical regulatory graphs preserving essential dynamical properties. In *CMSB'09: Proceedings of the seventh international conference on Computational Methods in Systems Biology*, volume 5688 of *Lecture Notes in BioInformatics*, pages 266–280. Springer-Verlag, 2009.

[38] A. Naldi, D. Thieffry, and C. Chaouiya. Decision diagrams for the representation of logical models of regulatory networks. In *CMSB'07*, volume 4695 of *Lecture Notes in Bioinformatics (LNBI)*, pages 233–247, 2007.

[39] A. Polynikis, S. J. Hogan, and M. di Bernardo. Comparing different ode modelling approaches for gene regulatory networks. *J Theor Biol*, 261(4):511–30, Dec 2009.

[40] V. N. Reddy, M. L. Mavrovouniotis, and M. N. Liebman. Petri net representations in metabolic pathways. In L. Hunter, D. B. Searls, and J. W. Shavlik, editors, *Proceedings of the 1st International Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 328–336. AAAI Press, 1993.

[41] A. Regev, W. Silverman, and E. Y. Shapiro. Representation and simulation of biochemical processes using the pi-calculus process algebra. In *Proceedings of the sixth Pacific Symposium of Biocomputing*, pages 459–470, 2001.

[42] E. Remy and P. Ruet. From minimal signed circuits to the dynamics of boolean regulatory networks. *Bioinformatics*, 24(16):i220–6, 2008.

[43] A. Rizk, G. Batt, F. Fages, and S. Soliman. A general computational method for robustness analysis with applications to synthetic gene networks. *Bioinformatics*, 12(25):il69–il78, June 2009.

[44] D. Ropers, V. Baldazzi, and H. de Jong. Model reduction using piecewise-linear approximations preserves dynamic properties of the carbon starvation response in escherichia coli. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 99(PrePrints), 2009.

[45] E. Snoussi. Qualitative dynamis of piecewise-linear differential equations: a discrete mapping approach. *Dyn Stab Syst*, 4:189–207, 1989.

[46] E. Snoussi and R. Thomas. Logical identification of all steady states: the concept of feedback loop characteristic states. *Bull. Math. Biol.*, 55(5):973–91, 1993.

[47] E. Snoussi, R. Thomas, and R. D'Ari. *Generalized kinetic logics*, chapter 7. CRC Press, 1990.

[48] E. H. Snoussi. Necessary conditions for multistationarity and stable periodicity. *Journal of Biological Systems*, 6:3–9, 1998.

[49] S. Soliman and M. Heiner. A unique transformation from ordinary differential equations to reaction networks. *BMC Bioinformatics*, submitted.

[50] C. Soulé. Graphic requirements for multistationarity. *ComplexUs*, 1:123–133, 2003.

[51] R. Srivastava, M. S. Peterson, and W. E. Bentley. Stochastic kinetic analysis of the escherichia coli stress circuit using σ^{32}-targeted antisense. *Biotechnol Bioeng*, 75(1):120–9, 2001.

[52] M. Sugita. Functional analysis of chemical systems in vivo using a logical circuit equivalent. *J Theor Biol*, 1:415–30, Oct 1961.

[53] D. Thieffry. Dynamical roles of biological regulatory circuits. *Brief Bioinform.*, 8(4):220–225, 2007.

[54] D. Thieffry and R. Thomas. Dynamical behaviour of biological regulatory networks, ii. immunity control in bacteriophage lambda. *Bul. Math. Biol.*, 57(2):277–297, 1995.

[55] R. Thomas. Boolean formalisation of genetic control circuits. *J. Theor. Biol.*, 42:565–583, 1973.

[56] R. Thomas. Regulatory networks seen as asynchronous automata: A logical description. *J. Theor. Biol.*, 153:1–23, 1991.

[57] R. Thomas and R. D'Ari. *Biological Feedback*. CRC Press, 1990.

[58] D. M. Wittmann, J. Krumsiek, J. Saez-Rodriguez, D. A. Lauffenburger, S. Klamt, and F. J. Theis. Transforming Boolean models to continuous models: methodology and application to T-cell receptor signaling. *BMC Syst Biol*, 3:98, 2009.

Centre de recherche INRIA Paris – Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399