Design, optimization and control in systems and synthetic biology

Gregory Batt

UNIVERSITÉ PARIS DIDEROT - PARIS 7
INRIA Paris-Rocquencourt

Habilitation à Diriger les Recherches
Spécialité : Biologie des systèmes

Mémoire présenté par
Grégory BATT

Design, Optimization and Control in Systems and Synthetic Biology

---

Soutenue le 7 mars 2013 devant un jury composé de

Reiner Veitia	Président	Université Paris Diderot
Vincent Danos	Rapporteur	Edinburgh University/CNRS Paris
Frédéric Devaux	Rapporteur	Université Pierre et Marie Curie
Olivier Gandrillon	Rapporteur	CNRS Lyon
Hidde de Jong	Examinateur	INRIA Grenoble - Rhône-Alpes
Mustafa Khammash	Examinateur	ETH Zürich

"Restons ce que nous avons toujours été : des gens libres.
C'est devenu suffisamment rare pour qu'on s'accroche ne serait-ce qu'à l'idée."
Enki Bilal

À Céline

Acknowledgments

First I would like to thank the members of my HDR committee: Frédéric Devaux and Reiner Veitia for being active promoters of quantitative biology amongst biological communities that are not (yet!) fully supportive, Vincent Danos and Olivier Gandrillon for sharing excitement of interdisciplinary research, and Mustafa Khammash for his inspirational work at the crossroad of control, computational sciences and quantitative biology. In the context of an *habilitation à diriger les recherches*, the presence of the former PhD adviser has a special significance. I'm very pleased to have here the opportunity to express all my gratitude to Hidde de Jong. In addition to all the scientific and technical skills I learned from him, he constantly demonstrated an exemplary rigorous and ethical attitude. In highly complex and competitive scientific fields, these two qualities are essential. He also set the standard for PhD supervision quality. Quite high actually. I simply hope I will be able with time to reach his level of guidance.

Regarding PhD supervision, I consider myself extremely lucky to be the (co-)supervisor of outstanding students: Francois Bertaux, Xavier Duportet, Artemis Llamosi, Jean-Baptiste Lugagne and Jannis Uhlendorf. What makes supervision really enjoyable is that knowledge flows in both ways. I hope they learned from me, but I also learned a lot from them. Close collaborators have also played a critical role regarding the type of research I have done and the pleasure I had to do it. I am particularly grateful to Valentina Peschetola and Szymon Stoma, and Dirk Drasdo, Pascal Hersen, and Ron Weiss, with whom I co-supervise students. It is striking to see how collaborations can develop throughout the years when people share scientific interests and mutual understanding. A significant part of my research is now carried out in tandem with Pascal. I am convinced that on the long-term this collaboration will have a major impact of the scientific value of my work. I'm grateful to INRIA that provided support at a time when it was needed to start collaborations, and to the Contraintes/Lifeware group that provides, under the aegis of Francois Fages, a very pleasant working environment where the pleasure to do research reigns.

The research material presented in this manuscript is the result of a large collaborative effort. In addition to the people I mentioned above, many others contributed directly or indirectly to this research and the one to come. Therefore I would like to thank Alessandro Abate, Kirill Batmanov, Calin Belta, Claudine Chaouyia, Eugenio Cinquemani, Alexandre Donzé, Samuel Druhle, Giancarlo Ferrari-Trecate, Hubert Garavel, Hans Geiselmann, Blaise Genest, Andres Gonzalez, Jean-Luc Gouzé, Jean Krivine, Cédric Lhoussaine, Ariel Lindner, Oded Maler, Radu Mateescu, Clément Nizak, Michel Page, Delphine Ropers, Brian Teague, P.S. Thiagarajan, Denis Thieffry, Ilya Tkachev, Cristian Versari, Jake Wintermute, Boyan Yordanov, and all members of Contraintes and Lifeware groups.

Research is not just pure science. Therefore I would like to thank Francoise de Connick, Nadia Mesrar, Stephanie Aubin, and Assia Saadi for their help (and patience!) on all administrative aspects throughout the years.

... and life is not just research. I am extremely grateful to my family and close friends for their constant support. And above all, none of this would have been possible without the love and understanding of Céline. Her importance in my life cannot be expressed in words. I dedicate this thesis to her.

Contents

1 Introduction

"Le véritable voyage de découverte ne consiste pas à chercher de nouveaux paysages, mais à avoir de nouveaux yeux"
Marcel Proust

Since 50 years, biology is undergoing a revolution. Driven by technological progress, the possibility to investigate the functioning of biological processes at the cellular level are expanding at an ever-increasing pace. Over the second half of the last century, the expansion of molecular and cellular biology has been stunning. It is now possible to sequence whole genomes for a few thousands of dollars [81], to get access to the mRNA content of cells in a routine manner [64], to get access to the protein content of whole tissues [58]. One can also observe protein and mRNA levels and locations in single cells for extended durations [54, 70]. Besides observing biological processes, one can also manipulate them with unprecedented capabilities. One can construct and integrate large genetic circuits [43, 56], synthesize entire genomes [40], hijack metabolism to efficiently produce biomolecules [69]. These techniques offer enormous opportunities [52]. In biotechnology major research efforts are invested for the production of biofuel or of high-value biomolecules, and for the development of biosensors or of bioremediation systems. In medicine, virus-based or cell-based therapies are envisioned using reprogrammed biological systems that exploit and expand their natural capabilities to create artificial tissues or even organs. The development of biosensors for drug screening and design is another application of high interest.

The technological push in modern biology is so strong that it has cleaved the historical domains of biology, ranging from biochemistry to physiology, into two broad areas: small-scale and large-scale biology [16]. In a nutshell, small-scale biology -also known as bottom-up systems biology- focuses on gathering detailed information on the specific components of a particular dynamical process. Data is often acquired with high temporal resolution and at the single cell or even single molecule level. Large-scale biology -also known as top-down systems biology- aims at getting a snapshot of the state of the cell for all biological processes. Because not everything can be observed at the same time, one distinguishes genomics, transcriptomics, proteomics, metabolomics, etc, according to the focus on the particular method employed [98, 45]. One of the main issues of large-scale approaches is that they often offer extensive but disconnected views of biological processes. The integration of these different views is often extremely challenging at such a scale [98, 45]. Actually, rather than bringing an increased understanding, these methods have revealed that the complexity of the functioning of cellular systems was even greater than originally assumed. This currently limits the usefulness of the "omics" approach. On the contrary, small-scale biology has been more successful in providing explanations on the functioning of biological processes thanks to a more integrated view [82, 65]. Naturally, it lacks extent. The integration of individual processes in the context of the functioning of the whole cell is still drastically missing. Therefore, it appears that any form of "mid-scale" biology that offers a better compromise in depth and breadth of information than existing methods has a significant potential to contribute to systems and synthetic biology.

Extending the depth/breadth frontier is precisely what modeling can do. Indeed, constructing a model that involves observed inputs and outputs, and unobserved variables is precisely a way to test assumptions on unobserved quantities. A striking example is the work of Suter and colleagues in which promoter properties are deduced from the observation of protein levels [90]. The main conclusion of the paper is that mammalian genes are transcribed with widely different bursting kinetics, yet transcription is never directly observed. In this case, modeling and carefully crafted experimental design were the critical elements that enabled linking an observed quantity, protein levels, with other, unobserved quantities, transcription rate and promoter bursting. A second example is the work of Spencer et al [86]. Here the authors relate observed variability in protein concentrations and in time of cell death via modeling apoptotic pathways and conclude that the naturally occurring differences in the levels of proteins regulating receptor-mediated apoptosis are the primary causes of cell-to-cell variability in the timing of death. Here again, the joint use of an appropriate model and of question-driven experiments has been instrumental to draw the proposed conclusions. Because models can push the depth/breadth frontier beyong what is currently possible by direct observation, modeling

has a potentially critical role in biology. One should note however, that genuine contributions of modeling to biology are still rare. One possible explanation is that it necessitates the combination of a well-defined biological question, of a question-driven experimental approach, and of the use of an appropriate modeling framework. Although this seems obvious, in practice, putting together these three aspects so that they fit perfectly together is an exquisitely delicate work.

Even if it is still rarely the case that modeling can provide solid evidence to draw new conclusions, it is often the case that modeling is effective to detect inconsistencies between existing data and current understanding. For example data has been accumulated for many years on bacterial adaptation to a variety of environmental stresses. Many papers explained adaptation to nutritional stress and the subsequent changes in expression of many genes by the combined effects of positive and negative transcription regulators. Yet, careful modeling of the various genetic regulatory interactions that were assumed to underlie the observed global response lead to inconsistencies and motivated the systematic analysis of the expression levels of all key genes involved in the nutritional stress response. This work lead to the striking conclusion that the large majority of expression changes were due to global changes of the gene expression machinery, rather than to specific control by transcription factors [15]. Therefore models have a significant role to play in testing the consistency of current understanding with actual experimental data. it is a critical tool to ensure that novel information can be aggregated with previous ones on a solid ground.

I started my PhD in 2002 and since then my research deals with computational systems biology. I addressed a number of diverse problems, ranging from model validation, hypothesis testing, robustness assessment, system design, optimization and control. My contributions include the development of novel mathematical methods, their implementation in publicly-available tools, the application of advanced modeling techniques, developed by myself or others, and the joint development of experimental plans and computational procedures to obtain a good integration of wet lab and dry lab data. Even if the overall objective of my work is to better understand the functioning of cellular processes, there is a marked trend towards problems that allow for a tight integration between experiment and modeling. In a sense my research has followed over the years the global trend of systems biology of being more and more quantitative. But more specifically, I focused on problems that enabled a good match between modeling predictions and experimental data. If modeling is expected to play an important role, the experimental setup and the computational framework should be jointly selected by the wet lab and dry lab biologists.

A second evolution of my research is to gradually evolve from method-driven research to problem-driven research. In the first case, the object of the research is a methodological tool, such as a theorem that applies to a particular class of dynamical systems, that is then applied to the most promising problems. In the second case, the object of the research is a particular biological process that one wants to understand, hijack for application purposes or control. This problem is solved using any methods that are appropriate. Naturally, both type of research are of value and both directions should be pursued. In my experiment, the second direction is superior on a critical aspect: it guarantees biological relevance of the work. Its potential drawback is that it does not guarantee the generality of the solution. On a personal level, I found that the first aspect is more important than the second. Both aspects can of course be combined and it could be argued that the main task of the person in charge of directing research is to select the most interesting problems, namely those that originate from a genuine biological question and whose resolution can be generalized to other related problems. Those problems are probably the most challenging to solve, but also will guarantee that the research will be original and innovative. They also require broad expertise to appreciate the specific difficulties on all aspects. On the biological side, the most critical domains of expertise are molecular and cellular biology. A good knowledge of existing experimental methods is required as well. On the methodological side, the important domains of expertise are dynamical systems and control, and computer science. One of the main objective of my research work was to gain some expertise in these various domains, that strengthen and complement my initial background in biology and computer science.

In the remainder of this manuscript, I will present and discuss a selection of my works that I consider representative of my contribution to the domain of computational systems biology and of my research path. As it will appear I started by considering rather abstract models of gene networks,

adapted to the level of information that is generally available in systems biology (Chapter 2), then considered better characterized systems such as synthetic gene networks and extended the previous framework to a more quantitative setting (Chapter 3). When one assumes that an even more complete knowledge is available, one can work with distributions of parameters and investigate the robustness of various properties (Chapter 4). In all the previous works, the work was done in tight collaboration with biologists, but the biological relevance of the work was assessed based on existing biological data. The application of modeling to practical control problems offered me the opportunity to co-supervise, in collaboration with Pascal Hersen (MSC lab CNRS/Paris 7) interdisciplinary work combining wet and dry lab biology (Chapter 5). I am now importantly involved in the continuation of this work in several directions and in two other collaborative projects with the Weiss lab for synthetic biology (MIT) and with the Drasdo group on multicellular system modeling (Bang, INRIA). In the near future, I plan to continue the development of these research directions that combine modeling and experiment in an intricate manner (Chapter 6).

2 Testing the consistency of regulatory interactions in medium-scale gene networks

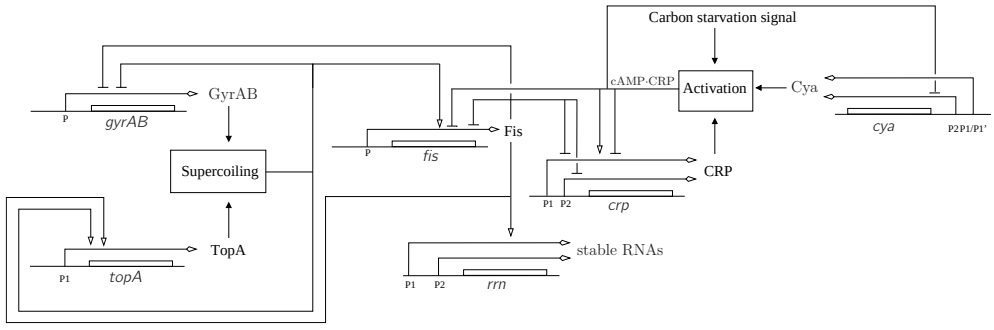
"All models are wrong but some are useful"
George Edward Pelham Box

2.1 Qualitative models of large gene networks

During my PhD I worked on a method developed for the modeling of relatively large genetic regulatory networks (up to 10-15 genes). These networks are made of genes coding for proteins, mostly transcription factors, and of regulatory influences of the proteins on gene expression. This framework is appropriate to describe a number of processes that mostly rely on intricate gene regulations. A number of developmental processes falls into this realm. On can notably mention the responses to environmental changes, like the sporulation decision in *B. subtilis* [26] or the adaptation to nutritional stress in *E. coli* [78] (Figure 1(a)), developmental processes, like flower morphogenesis [61] or drosophila development [79], and oscillatory behaviors, like cell cycle [35, 25].

For such complex cell processes, the information needed to develop quantitative models is seldom available. This seriously challenges the biological relevance of the standard quantitative framework used to model such networks: Hill-type models. These models are differential equation models where sigmoidal functions (Hill functions) are used to describe regulatory influences of proteins on gene expression, and more precisely on promoter activity. In particular, when genes are regulated by several transcription factors, obtaining a quantitative description of the regulation function is particularly challenging [80]. In most cases however, qualitative information is available on the "logic" of gene regulation.

(a)



(b)

$$\dot{x}_{topA} = \kappa^1_{topA} + \kappa^2_{topA}\, s^+(x_{gyrAB}, \theta^3_{gyrAB})\, s^-(x_{topA}, \theta^1_{topA})\, s^+(x_{fis}, \theta^4_{fis}) - \gamma_{topA}\, x_{topA}$$

$$0 < \kappa^1_{topA}/\gamma_{topA} < \theta^1_{topA} < \theta^2_{topA} < \theta^3_{topA} < (\kappa^1_{topA} + \kappa^2_{topA})/\gamma_{topA} < max_{topA}$$

Figure 1: **Gene regulatory networks and PADE models**. Network of key genes, proteins, and regulatory interactions involved in the nutritional stress network in *E. coli* [78]. (b) Piecewise affine differential equation and parameter inequality constraints for one protein in the network, the topoisomerase TopA. This model uses step functions s^+.

2.2 Reasoning with qualitative constraints on parameters

Several modeling frameworks have been proposed to deal with this problem, most notably logical regulatory models [91] and qualitative piecewise-affine differential equations (PADE) systems [28] (Figure 1(b)). The key idea of PADE system modeling is to abstract Hill functions appearing in Hill-type models by step functions. Step functions do capture the non-linearity and the saturating aspects of Hill functions, which are two critical features of Hill functions for biological relevance. However, the specific shape of the regulatory function, which is generally not known, is abstracted

away. This framework leads to switched affine systems. In each region of the switched system the dynamics can be described and solved very easily. More precisely, transitions from one region to another region can be inferred from qualitative information on parameters. Stated differently, the globally complex problem is recast in a set of locally simple problems. The analysis of such systems relies on the computation of a state transition graph that represents the dynamics of the system in the state space. States represent a partition of the state space into a set of hyperrectangular regions and transitions represent local reachability between regions.

One of the main issues with this and related formalisms arises from the simplifications that are employed. The dynamics of the system changes when protein concentrations cross threshold concentrations. Within threshold hyperplanes the dynamics is not defined. Even if the set of all regions that are in one or more threshold hyperplanes, called singular regions, is of measure zero in the whole state space, these regions cannot be neglected because they often contain attractors of the dynamics. Several attempts to define the dynamics in these regions have been made in PADE and related formalisms [62, 73]. A mathematically satisfying solution has been proposed by Gouzé and Sari using Filippov regularizations [42]. In short, the dynamics is defined in singular regions as the convex combination of the dynamics in neighboring regular regions. This extension leads to differential inclusions, defined everywhere, instead of differential equations, defined only almost everywhere [42]. Unfortunately, the analysis of this class of systems necessitated quantitative information on parameters. Gouzé and de Jong found that using rectangular combinations in place of Filippov's convex combinations makes it possible to compute the state transition graph representing in an abstract manner the dynamics solely by reasoning with the relative order of the model parameters [28]. This is critical to solve the original problem in a mathematically well grounded manner with qualitative reasoning only.

2.3 Model validation

This qualitative framework is ideally suited to test the consistency of our understanding of the functioning of complex biological processes and on the often heterogeneous and qualitative available experimental information. Indeed, because one reasons for large sets of parameter values, simply defined in terms of ordering relations, the predictions obtained by this method are by essence very robust. Therefore, if an observed behavior is not accounted for by a qualitative model, it seriously calls for model revision. Developing an approach for testing the validity of qualitative models of genetic regulatory networks was precisely the topic of my PhD work. The objective is to encode the observed property in some formal framework and test whether the model satisfies this property in an automated and efficient manner. This is precisely the objective of model checking [23]. Over the years, the formal verification community has developed extremely efficient methods for testing whether discrete transition systems are satisfying dynamical properties, often encoded in temporal logic [22, 17]. Temporal logics are flexible languages able to express a broad range of dynamical properties [72]. Therefore this framework is well suited for our problem. However, because the available information is almost exclusively available in arbitrary units, it is important to be able to exploit the information on the direction of the variations. Stated differently, although the exact value of the measures we have is generally meaningless, the information on their change in time contains valuable information. Unfortunately, the level of abstraction of the classical analysis made on PADE models, and the one used by similar formalisms, are too coarse to develop discriminant model validation approaches: often, the sign of the derivative of the variables is unknown, and so could be matched with any observation. The main technical contribution of my thesis is therefore to show that with the existing information on parameters, one can make a finer-grained analysis that leads to a more detailed representation of the dynamics in which derivative signs are known. This lead me to reimplement the core of the qualitative simulator *Genetic Network Analyzer*[1] (GNA) developed by the Helix (now Ibis) group at INRIA [27, 8]. This approach has been applied to the validation of two models of bacterial stress adaptation responses: sporulation in *B. subtilis* [26] and nutritional stress response in *E. coli* [78]. In the first case, the comparison with experimental data [71] revealed

1. GNA is freely available for academic research at the address: http://ibis.inrialpes.fr/article122.html

that the observed expression pattern of a protein, Hpr, is incompatible with the model. Further analysis showed that this observation is also in contradiction with the role generally assumed for Hpr [89], calling for the experimental validation of Hpr expression pattern. In the second case, we obtained different discrepancies between model and observations during the entry of *E. coli* cells into stationary phase [73]. This motivated the group to initiate an ambitious research program that lead to the finding that the global regulation of gene expression, neglected in our model as in the vast majority of the experimental studies, has a major role in the adaptation to nutritional stress [15].

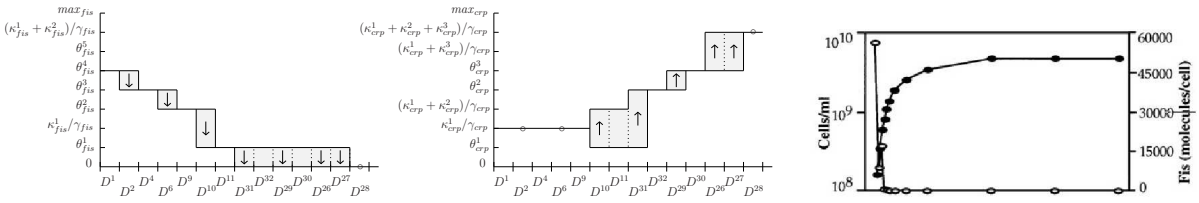


Figure 2: **Model validation.** Temporal evolution of the concentration of proteins in the nutritional stress response network during the transition to stationary phase. (a) Predictions for Fis and CRP in a path in the state transition graph generated by qualitative simulation. (b) Observation of Fis concentration (open circles) during the growth-phase transition, as indicated by cell density (closed circles) [4].

2.4 Scaling up

The proposed approach is based on the automated analysis of a class of differential inclusion systems obtained by regularizing the dynamics of differential equation systems with discontinuities. Recently, we found that huge computational gains are possible by using extended step functions [11]. Stated simply, the idea is to define the system directly using differential inclusions, instead of defining the dynamics with differential equations and regularizing it afterwards. Although, this comes at the price of a second (modest) over-approximation, it greatly simplifies the presentation of the approach and, most importantly, the computations to be done. This step has been critical to propose a fully symbolic representation of the dynamics that enabled the use of highly-efficient symbolic analysis tools [21]. The computational gains have been obtained this way significantly extended the class of problems that are solvable via this approach [11]. As often with theory, it takes a lot of efforts to find a simple solution to a complex problem. But then, thanks to its simplicity, or more precisely to its high regularity, this solution then offers key advantage for the resolution of the problem.

To illustrate the effectiveness of the approach, we considered the problem of parameter search for qualitative models (Figure 3(a)-(b)). By considering all possible parameter orderings, it is possible to explore exhaustively the parameter space. The challenge naturally comes from the combinatorial explosion of the number of parameter orderings -that is of models to analyze- with the increase of the size of the system, that is with the number of genes. The efficiency of the approach has been demonstrated on the redesign of one of the largest synthetic genetic regulatory network constructed so far [18] (Figure 3(c)-(d)). We have been able to find among thousands of possible qualitative parametrizations a handful of parameter orderings that guarantee (in theory!) the robust control of the behavior of the synthetic system.

2.5 Significance and perspectives

Proposed in 2003, the formal verification approach I developed during my PhD was one of the very first work proposing the application of model checking to systems biology problems [9, 10]. Since then model checking approaches for systems biology problems became quite popular (for reviews, see e.g. [30, 46, 36]). To the best of my knowledge, the approach I developed during my PhD

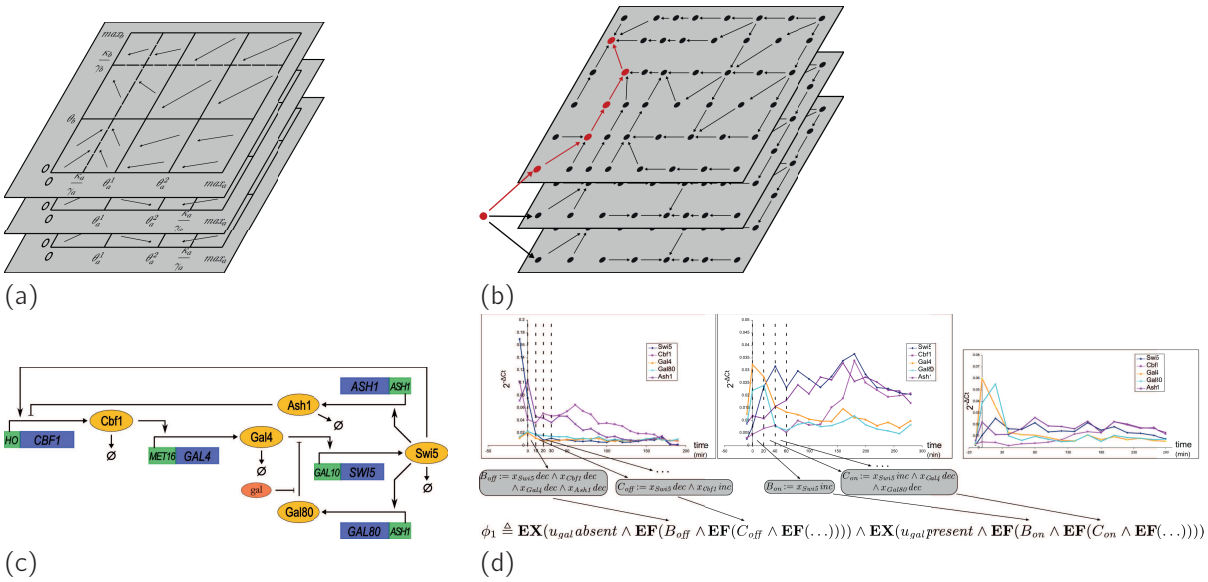


Figure 3: **Network optimization.** (a) Representation of the dynamics of a simple gene network, represented in the state space, for different parameter orderings. (b) Abstract representation of the dynamics of the system, represented by a state transition graph, for different parameter orderings. (c) The IRMA network in yeast: a network for in vivo assessment of reverse-engineering and modeling approaches [18]. (d) Representation in temporal logic formulas of the experimentally-observed gene expression profiles in IRMA.

for the validation of genetic regulatory network models is still the sole approach able to generate a fine grained representation of the dynamics adapted to model validation against real-life experimental data. In retrospect, I explain this by the fact that the approach aiming at defining the problem at the continuous level, solving the various issues that arise at this level, and then abstracting is more effective for solving the issues raised by the discretization of the dynamics than those aiming at defining the dynamics directly at the abstract level. This approach is still lacking the capability to reason in a compositional manner, so as to exploit the modularity in biological systems, as done in [60].

Even if the proposed approach is still one of the most attractive approach to make value of the (essentially qualitative) data that has been produced so far on biological networks, it is not in phase with a strong global trend in molecular and cellular biology: being more and more quantitative. Even if the biological relevance of quantitative precision offered by the novel experimental methods is still often largely questionable (lack of reproducibility, imperfect experimental designs, ...), in terms of perspective it is of high interest to develop approaches that can account for all the available information. In this respect, synthetic biology can be considered as a niche for developing modeling approaches for quantitative systems biology since systems are constructed to be manipulated, observed and optimized. Therefore, during my postdoctoral stay at Boston University I worked on adapting the idea of PADE system analysis to the quantitative problems found in synthetic biology. This line of research is described in the following chapter.

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

The modeling and simulation of genetic regulatory networks have created the need for tools for model validation. The main challenges of model validation are the achievement of a match between the precision of model predictions and experimental data, as well as the efficient and reliable comparison of the predictions and observations. We present an approach towards the validation of models of genetic regulatory networks addressing the above challenges. It combines a method for qualitative modeling and simulation with techniques for model checking, and is supported by a new version of the computer tool Genetic Network Analyzer (GNA). The model-validation approach has been applied to the analysis of the network controlling the nutritional stress response in Escherichia coli.

Efficient parameter search for qualitative models of regulatory networks using symbolic model checking (2010)
G. Batt, M. Page, I. Cantone, G. Goessler, P. Monteiro and H. de Jong
Bioinformatics, 26(18):i603-i610

Investigating the relation between the structure and behavior of complex biological networks often involves posing the question if the hypothesized structure of a regulatory network is consistent with the observed behavior, or if a proposed structure can generate a desired behavior. The above questions can be cast into a parameter search problem for qualitative models of regulatory networks. We develop a method based on symbolic model checking that avoids enumerating all possible parametrizations, and show that this method performs well on real biological problems, using the IRMA synthetic network and benchmark datasets. We test the consistency between IRMA and time-series expression profiles, and search for parameter modifications that would make the external control of the system behavior more robust.

3 Optimization of simple synthetic genetic circuits

"It doesn't matter how beautiful your theory is, it doesn't matter how smart you are. If it doesn't agree with experiment, it's wrong."
Richard P. Feynman

3.1 Quantitative yet robust analysis of synthetic genetic circuits

The number of biological processes that have been quantitatively studied with an accuracy that enables the development of biologically relevant quantitative models is still limited. Nevertheless their number is steadily increasing and it is likely that this trend will become a major direction in systems biology in the coming years. This is particularly true in the field of synthetic biology. The objective of synthetic biology is to develop methods that facilitates the engineering of biological systems implementing useful functions [3, 84]. Important potential application domains include the production or biofuels or of other types of biomolecules of technological or medical interest [69], or the development of novel therapeutic strategies like cell or tissue based therapies [38].

The correct functioning of such systems often involves quantitative aspects, constraining for example the minimal output amplitude or the maximal response time of synthetic systems. Therefore the qualitative approaches presented in the previous chapter are not appropriate for such applications. However, because of the intrinsic noisiness of the functioning of biological systems and the fact that they should accomplish their functions despite fluctuating environments, traditional quantitative engineering methods, based on numerical simulation of ordinary differential equation systems, are not appropriate either. One should design systems that behave robustly for sets of parameters or perturbations.

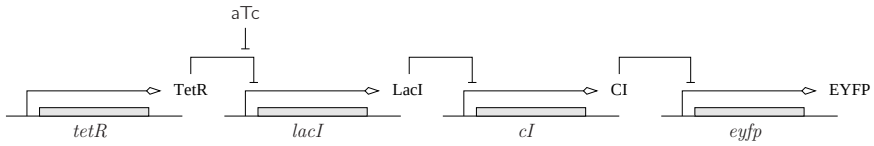
3.2 Reasoning with quantitative constraints on parameters

This objective can be met by using an extension of the previously-presented PADE method. In PADE systems the nonlinear responses of promoter activity, often represented via Hill functions, is abstracted by step functions. The idea here is to use a less drastic abstraction and use instead so-called ramp functions, a class of piecewise affine function. By using many segments, Hill functions can be approximated to any degree of accuracy. As previously, regulation functions describing the promoter activity as a function of the concentrations of its regulators are constructed by combining these elementary functions. This leads to a class of piecewise multiaffine (PWMA) models [6, 13]. This class of models has a nice mathematical property: in entire regions of the state and parameter space, the flow of the system is a convex combination of the flow at extreme points (vertices) of the region [14, 44]. Therefore, it is easy to identify that the flow is monotonic in some regions: the flow at all vertices point in the same direction. A more quantitative version of this intuition states that, given an appropriate partition of the state and parameter spaces, the derivative everywhere in each region is included in the convex hull of the derivatives at the vertices of this region. It is then possible to use the same idea as with qualitative PADE models: defining the state transition graph in which states represent regions of the partition and transitions represent the possibility for the system to go from one region to another. This graph can again be efficiently analyzed by model checking. This allows to identify parameter sets for which one can guarantee that a given behavior is necessarily present or impossible. Because of the approximations that are made, the approach is conservative. When parameter sets are identified as valid, one has the guarantee that this holds. But valid parameters might be missed. This happens when because of the approximations, the approach cannot prove their validity.

To improve the efficiency of the (exhaustive) exploration of the parameter space, one can extend the approach previously sketched as follows. Instead of fully partitioning the parameter space and testing each parameter region, one can partition the parameter space in a dynamic manner. Parameter constraints are added only when needed, leading to a hierarchical approach in which parameter space partitioning and model verification alternate. This approach has been shown to be much more efficient [6]. However, it necessitated to work with parameter sets that are not equivalence classes

(but are unions of equivalent classes) and non-trivial theoretical extensions have been needed [12]. This approach has been implemented in the tool RoVerGeNe (standing for robust verification of gene newtorks) [2].

(a)



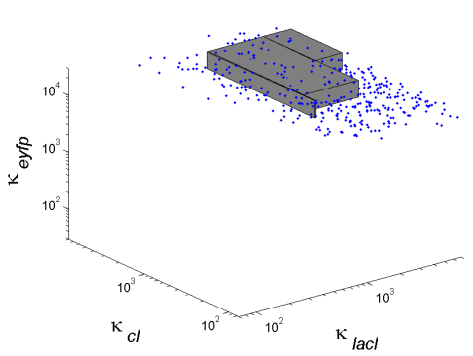
(b)

$$
\begin{aligned}
\dot{x}_{tetR} &= \kappa_{tetR} - \gamma_{tetR}\, x_{tetR},\\
\dot{x}_{lacI} &= \kappa_{lacI,0} + \kappa_{lacI}(r_{lacI,1}(x_{tetR}) + r_{lacI,2}(u_{aTc}) - r_{lacI,1}(x_{tetR})\, r_{lacI,2}(u_{aTc})) - \gamma_{lacI}\, x_{lacI},\\
\dot{x}_{cI} &= \kappa_{cI,0} + \kappa_{cI}\; r_{cI}(x_{lacI}) - \gamma_{cI}\, x_{cI},\\
\dot{x}_{eyfp} &= \kappa_{eyfp,0} + \kappa_{eyfp}\; r_{eyfp}(x_{cI}) - \gamma_{eyfp}\, x_{eyfp},\\
\dot{u}_{aTc} &= 0.
\end{aligned}
$$

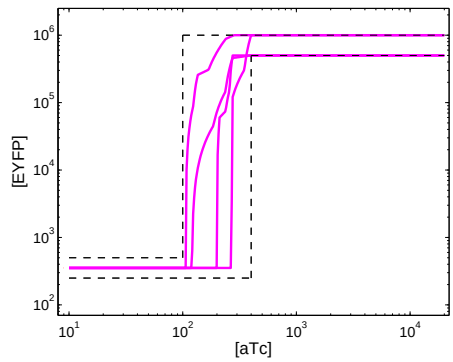
Figure 4: (a) Synthetic transcriptional cascade constructed and characterized in the Weiss lab [48]. (b) Piecewise-multiaffine model of the cascade in (a). This model uses so-called ramp functions r.

3.3 Optimization of synthetic circuits: in silico studies

This approach has been applied to the analysis of a genetic circuit made of a cascade of repressing transcriptional factors [48] (Figure 4). We first generated constraints on protein synthesis parameters so as to optimize the input/output response of the circuit. We found a set of constraints on protein production parameters (Figure 5(a)), suggesting biological modifications of the network, such as tuning ribosome binding sites. Then, we tested the robustness of the newly parametrized system with respect to variations of all its parameters. We found that the systems is guaranteed to satisfy the desired behavior for a relatively large set of parameters: the expected property is satisfied by the system for any parameter fluctuations in $\pm 10\%$ ranges around their nominal values.



(a)



(b)

Figure 5: (a) Valid parameters in the parameter space as identified by RoVerGeNe (rectangular regions) or by brute-force sampling (dots). κ_{lacI}, κ_{cI}, and κ_{eyfp} are production rate parameters for three proteins of the transcriptional cascade. (b) Steady-state input/output behavior of the cascade for extreme parameter values in the valid parameter sets represented (a) showing that relevant parameter constraints have been identified by the approach. The output is expected to remain between the bounds represented by dotted lines.

2. RoVerGeNe is freely available for academic research. It can be downloaded from the address: http://iasi.bu.edu/∼batt/rovergene/rovergene.htm

The property of interest stated that at steady state the output of the system should remain between given bounds, these bounds depending on the input (Figure 5(b)). Therefore, to express this property in temporal logic, one typically states that eventually some constraints hold and that they will then remain always true. To test whether a property eventually hold, one should consider trajectories of the system where the time diverges (ie is unbounded). Because time is abstracted away in the state transition graph that is analyzed by model checking, a specific analysis of the divergence of time in the regions where the attractors lie need to be done before model checking [7]. An extension to capture more quantitative constraints on the time spent in regions of the state space has been carried out subsequently using timed automata and timed logics instead of discrete transitions systems and temporal logics [59].

3.4 Significance and perspectives

One should insist here on the computational difficulty of the problem. Indeed one tries to prove that a global dynamical property holds for a 4 dimensional nonlinear system and for all parameters in an 11 dimensional parameter space. We have demonstrated that the approach is computationally feasible for systems of reasonable complexity as typically encountered in the field (the typical size of synthetic gene network is 2 to 6 genes). Yet, scalability to larger systems would be an issue, since the complexity of the approach scales exponentially with the dimensionality of the system.

However, the main drawback of this and the previous approach on PADE models is that the discrete abstractions suffer from the well-known transitivity problem. In short, the problem is encountered when solutions can go from a region A to a region B, and from a region B to a region C, but no solution traverses the A, B, and C regions. Such information will be lost in the abstractions in which transitions from A to B and from B to C will exists. There is no information to infer that the sequence of transition A$\rightarrow$B $\rightarrow$C is not valid. As a consequence, one might not be able to prove that some property hold based on the state transition graph. For large classes of properties of interest the conservativeness of the approach might prevent from performing informative analyses. This is an intrinsic limitation of any approaches using discrete abstractions. A second limitation comes from the treatment of parameter uncertainties. In our models, parameters are assumed to lie in sets, however, they can in principle vary in a unbounded manner in these sets across time. While it is true that parameters are likely to fluctuate in time, assuming that all temporal variations are admissible within the given bounds leads to overly conservative results. An alternative approach is to work with parameter distributions. That is, instead of set-valued uncertainties we consider probabilistic uncertainties. This second approach is employed in the next chapter.

Robustness analysis and tuning of synthetic gene networks (2007)
G. Batt, B. Yordanov, C. Belta and R. Weiss
Bioinformatics, 3(18):2415-2422

> The goal of synthetic biology is to design and construct biological systems that present a desired behavior. The construction of synthetic gene networks implementing simple functions has demonstrated the feasibility of this approach. However, the design of these networks is difficult, notably because existing techniques and tools are not adapted to deal with uncertainties on molecular concentrations and parameter values. We propose an approach for the analysis of a class of uncertain piecewise-multiaffine differential equation models. This modeling framework is well adapted to the experimental data currently available. Moreover, these models present interesting mathematical properties that allow the development of efficient algorithms for solving robustness analyses and tuning problems. These algorithms are implemented in the tool RoVerGeNe, and their practical applicability and biological relevance are demonstrated on the analysis of the tuning of a synthetic transcriptional cascade built in Escherichia coli.

4 Investigating dynamical properties of complex networks

"Un état dangereux : croire comprendre"
Paul Valéry

4.1 Quantitative analysis of large biological networks

In my previous works, I considered methods that enabled to reason for sets of parameters. The preeminent advantage of these approaches is that they provide robust predictions of the possible behaviors of the biomolecular system under study. Given the usually high level of uncertainty on precise mechanisms of biomolecular reactions, initial conditions and parameter values, this aspect is of utmost importance for systems and synthetic biology applications. By using highly efficient tools from formal methods, one can exhaustively test dynamical properties of interest in the whole state and parameter spaces. The exhaustiveness of the search is important for model (in)validation. We have been able to identify that a previously-proposed qualitative model of nutritional stress response in *E. coli* cannot account for the observed protein variations during and after the transition to stationary phase or to propose robust network modifications to improve the all-or-none response of a synthetic transcriptional cascade in *E. coli*.

However, these approaches suffer from discrete abstraction problems. In both cases, the idea is to partition the state space, compute local reachability properties (ie reachability between regions of the partition), and define a state transition graph where nodes are regions and transitions represent the possibility to go from one region to another. Therefore this state transition graph is an abstract representation of the dynamics: to each trajectory of the original system corresponds a path in the graph. The problem comes from the fact that the converse does not necessarily hold: some paths in the graphs correspond to no real trajectory. The fact that one trajectory can reach region B from A and that another con reach region C from B will result in the presence of a path reaching C from A in the graph irrespectively of the existence of trajectories of the original system reaching C from A: discrete abstraction creates spurious trajectories. They often severely limit the prediction capabilities of discrete-abstraction-based approaches.

A well-known alternative to model parameter uncertainty is to work with the set of trajectories that one obtains by considering dynamical systems with parameter distributions. The behavior of such systems can then be seen as an (infinite) number of trajectories forming "tubes" of various densities in the state space. Like discrete abstraction methods, scalability is an issue. When the size of the system increases, the volume of the space to analyze or to sample increases exponentially. Moreover, because biologically-relevant distributions for parameters and initial conditions have generally an unbounded support (e.g., normal and log-normal distributions), a very broad diversity of behaviors are possible with low probability. Therefore one is then confronted to the analysis of a large number of trajectories to systematically investigate and visual inspection rapidly becomes impractical. The strategy we developed is to use temporal logic to define the properties of interest and (a quantitative version of) model checking to test whether (or more precisely how well) the trajectories satisfy the expected properties.

4.2 Reasoning with large sets of trajectories

Typical properties that interest systems and synthetic biologists include verifying that the output of a three-stage transcriptional cascade in *E. coli* has reached the desired value in at most 7 hours, or that caspase-3 is never activated before caspase-8 during apoptosis in certain mammalian cell lines. Then, how can one relate these biological properties with the (infinite) set of trajectories generated by the corresponding models? Because parameter distributions are generally unbounded, asking that such properties always hold does not make much sense. In fact, one would like to get a score on how well trajectories satisfy the properties of interest so that statistics or optimization can be made, typically for robustness evaluation or optimization purposes.

The solution that we proposed is to use temporal logic to encode the desired dynamical properties and a quantitative interpretation of their satisfaction or violation. The notion of satisfaction degree

of a temporal logic property captures whether a specific behavior satisfies (or violates) robustly the property. With this tool, one can then represent graphically in the space of parameters or of initial conditions the satisfaction degree of various properties, use them to perform statistics or optimization, or even apply global sensitivity analysis methods to identify the parameters that are most influential for the property [76, 88].

In what follows, this idea is applied to two problems. In the first case, we propose to use temporal logic as a property specification language and present a computational framework that enables the computation of the robustness of many properties in a unique setting. In the second case, we show that temporal logics can be used to encode precisely observed system behaviors and that this enable to systematically challenge model predictions for various experimental observations and cell types.

4.3 Application to robust timing of a synthetic transcriptional cascade

In a seminal paper Kitano defined the robustness of a property a of a system s with respect to a set of perturbations P as the average of an evaluation function D_a^s of the system over all perturbations $p \in P$, weighted by the perturbation probabilities $prob(p)$ [53]: $R_{a,P}^s = \int_{p \in P} prob(p)\ D_a^s\ dp$

Unfortunately, Kitano does not provide much information on how to define the so-called *evaluation function* D_a^s of the system. This function should determine if the system still maintains its function under a perturbation and to what degree. The evaluation function needs to be defined for each specific problem in an ad-hoc manner. In [74, 75] we introduce the notion of satisfaction degree $sd(T_p, \phi)$ of a trajectory T_p of the system under perturbation p with respect to the temporal logic property ϕ and show that one can then provide a generic computational for the robustess simply by using $sd(T_p, \phi)$ in place of D_a^s in Kitano's definition.

In [75] we investigate whether the transcriptional cascade constructed in [48] and presented in the previous chapter can robustly be used as a biological timer. The response of the cascade to the addition of an inducer is characterized by a rapid increase of the fluorescence preceded by a significant lag-phase. This system could therefore be used as a timer for synthetic biology applications, for example for developmental programs. Unfortunately, the heterogeneity of the cell responses may prevent its robust use as a timer. Indeed, having even a low proportion of cells sending a signal too early or too late might compromise the correct functioning of the whole system.

To investigate the robustness of the "well-timed" behavior of the cascade, we developed a Hill-type model and searched for parameter distributions that fitted the observed mean and variance. Then simulation and model checking showed that the property was not fully robustly satisfied and global optimization was used to optimize the robustness of the behavior of the cascade. Interestingly, with the proposed parameter modifications, the variability is reduced at moments that are important for the specification as expected, but increased at less constrained times, suggesting the existence of a robustness/fragility trade-off.

Finally, we used global sensitivity analysis to study how parameter changes affect the robustness of the property. Our analysis suggested that heterogeneities in growth rates have a strong influence on the robustness of the property and that the performance of the cascade is limited by the fact that one repressor is not fully able to robustly repress its target gene.

4.4 Application to the comparison of different apoptotic responses in different cell types

In higher eukaryotic cells, a given environmental signal is processed in different manners in different cell types. Although the identification of the exact origins of such differences is still an open problem of major importance, experimental evidence indicates that subtle differences in the concentrations of signal transduction proteins may have an important impact. In [1], Sorger and colleagues provide detailed experimental data on extrinsic apoptosis in three cell types and propose a model of this pathway for these three different cell types. In agreement with the current understanding, cell type models have the same set of reactions but different initial protein concentration distributions. However, in [1] model predictions have not been systematically compared with the produced data.

In [88], we encode in temporal logic a set of observation dealing with the relative order of caspases activation, the necessity of mitochondria outer membrane permeabilization (MOMP) for effector caspase activation, and the survival of cell lines overexpressing Bcl2. All these properties deal with the role of mitochondria in cell death and are interchangeably used to classify cell lines in two types: type I (mitochondria independent death) and type II (mitochondria dependent death). Then, using simulation and model checking, we systematically tested the consistency of all observed behaviors with respect to all cell lines. This systematic procedure illustrated that these three properties are not equivalent and therefore result in inconsistent type I/II cell line classifications.

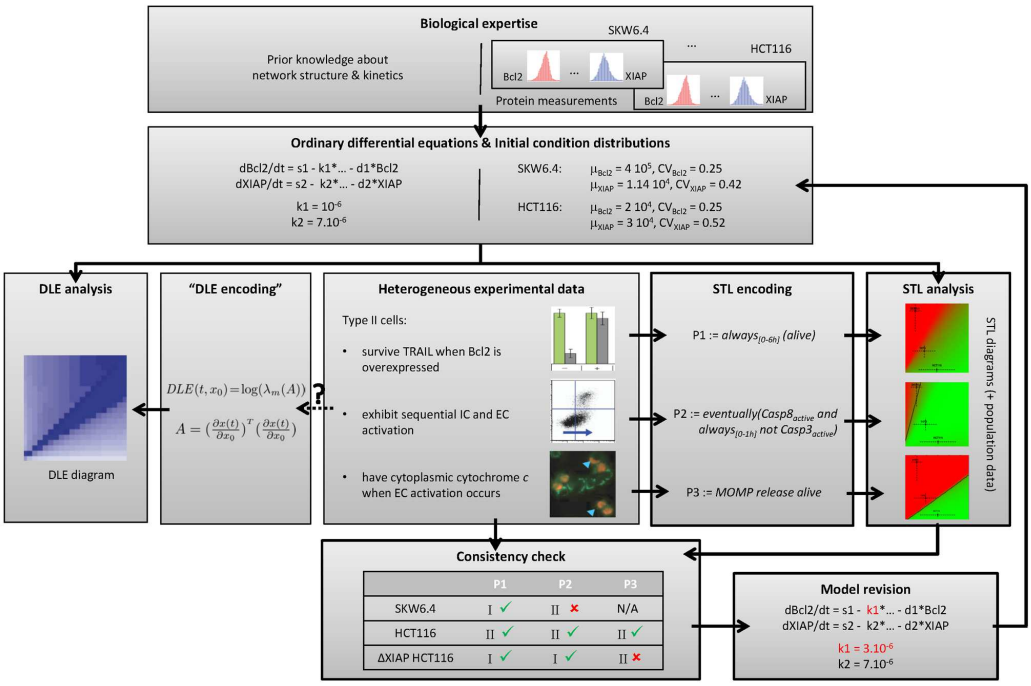


Figure 6: Property-based model analysis framework. Heterogeneous observations on the system are formalized as STL properties. Consistency between model and experimental observations is tested via STL diagrams and population data. Inconsistencies can be resolved via property-guided model revision. In contrast to the DLE-based approach proposed by Aldridge and colleagues [1], STL properties explicitly encode specific aspects of cell's response, in our case, of the role of mitochondria in type I/II apoptosis.

4.5 Significance and perspectives

The analysis of large ODE models is challenging. To obtain a reasonable picture of the dynamics, one needs to simulate many trajectories for many different initial conditions and parameter values. Visual inspection of the resulting set of trajectory then becomes impractical.

Temporal logics are property specification languages. They allow to express a broad class of observed or expected behaviors in a precise manner. Moreover recent theoretical developments introduced the notions of satisfaction degree for temporal logic formula evaluated on numerical traces. This is a significant contribution since the usual tools for the analysis of high dimensional systems (global sensitivity analysis, global optimization, etc) operate on real-valued quantities. Using the natural Boolean interpretation of temporal logic formulas would be inappropriate. A last advantage of temporal logics is their simplicity. A number of interesting properties can be expressed in an intuitive manner. On the negative side, the apparent simplicity to express properties can be misleading. It is

sometimes difficult to identify that what one has written is not exactly what one intended to write. Still, one should hope that these formal methods will be widely adopted by computational biologists working in systems and synthetic biology communities.

In comparison to the set-based approached presented in the previous chapters, deterministic models with parameter distributions adopt an other extreme view on parameter values. As seen previously, in set-based approaches, parameters can vary in an arbitrary manner across time within given bounds. That is, the model does not exclude behaviors in which parameters would jump at each time instant between two extreme values. This is obviously not realistic. In the probabilistic models that we presented here, parameters are initially sampled according to distributions, but then keep their values in time. Given that the physiology of the cell necessarily changes in time, let alone because it ages, this assumption is certainly not valid either. Therefore efforts should be made to develop a modeling framework in which parameters could slowly change in time. Experimental efforts should accompany these modeling developments to provide data to constrain the novel parameter fluctuation models. This will be all the more important that the duration of the experimentation increases.

A general computational method for robustness analysis with applications to synthetic gene networks (2009)
A. Rizk, G. Batt, F. Fages and S. Soliman
Bioinformatics, 25(12):i169-i178

> Robustness is the capacity of a system to maintain a function in the face of perturbations. It is essential for the correct functioning of natural and engineered biological systems. Robustness is generally defined in an ad-hoc, problem-dependent manner, thus hampering the fruitful development of a theory of biological robustness, advocated by Kitano [Mol Syst Biol, 3:137, 2007]. In this paper, we propose a general definition of robustness that applies to any biological function expressible in temporal logic LTL, and to broad model classes and perturbation types. Moreover, we propose a computational approach and an implementation in BIOCHAM 2.8 for the automated estimation of the robustness of a given behavior with respect to a given set of perturbations. The applicability and biological relevance of our approach is demonstrated by testing and improving the robustness of the timed behavior of a synthetic transcriptional cascade that could be used as a biological timer for synthetic biology applications.

STL-based analysis of TRAIL-induced apoptosis challenges the notion of type I/type II cell line classification (2013)
S. Stoma, A. Donzé, F. Bertaux, O. Maler, G. Batt
PLoS Computational Biology, 9(5):e1003056

> Extrinsic apoptosis is a programmed cell death triggered by external ligands, such as the TNF-related apoptosis inducing ligand (TRAIL). Depending on the cell line, the specific molecular mechanisms leading to cell death may significantly differ. Precise characterization of these differences is crucial for understanding and exploiting extrinsic apoptosis. Cells show distinct behaviors on several aspects of apoptosis, including (i) the relative order of caspases activation, (ii) the necessity of mitochondria outer membrane permeabilization (MOMP) for effector caspase activation, and (iii) the survival of cell lines overexpressing Bcl2. These differences are attributed to the activation of one of two pathways, leading to classification of cell lines into two groups: type I and type II. In this work we challenge this type I/type II cell line classification. We encode the three aforementioned distinguishing behaviors in a formal language, called signal temporal logic (STL), and use it to extensively test the validity of a previously-proposed model of TRAIL-induced apoptosis with respect to experimental observations made on different cell lines. After having solved a few inconsistencies using STL-guided parameter search, we show that these three criteria do not define consistent cell line classifications in type I or type II, and suggest mutants that are predicted to exhibit ambivalent behaviors. In particular, this finding sheds light on the role of a feedback loop between caspases, and reconciliates two apparently-conflicting views regarding the importance of either upstream or downstream processes for cell-type determination.

More generally, our work suggests that these three distinguishing behaviors should be merely considered as type I/II features rather than cell-type defining criteria. On the methodological side, this work illustrates the biological relevance of STL-diagrams, STL population data, and STL-guided parameter search implemented in the tool Breach. Such tools are well-adapted to the ever-increasing availability of heterogeneous knowledge on complex signal transduction pathways.

5 Computer-assisted control of biocellular processes

"What I cannot control, I do not understand."
Freely adapted from Richard P. Feynman

5.1 Real-time control in systems and synthetic biology

How predictive can models be? This question is critical for quantitative systems biology. The objective of this field is to propose explanation to observed phenomena in terms of well defined biological processes. Adaptation to an hyper-osmotic stress in yeast results from the activation of enzymes via a signal transduction pathway and subsequent synthesis of glycerol. To test whether our understanding can quantitatively explain observations, models are proposed that should at the very least account for the observations, and in the best cases, should predict new situations.

Yet, it is apparent that models have a very limited predictive power. It is rarely the case that a model developed to account for observation made in one specific context extends easily to observations made in a slightly different context. Does it indicate that our understanding is inaccurate or is it something that is to be expected given the importance of the cellular context for biological processes? This is a very fundamental question at the core of systems biology research.

To address this problem, I proposed to consider real-time model predictive control (MPC) problems. Given a target temporal profile for the output and a model of the system, the problem is to find how to play with the input so that the desired output behavior is obtained. In close collaboration with Pascal Hersen, we developed an automated experimental platform that in real time observes the current state of a biological process (outputs) and acts on it (inputs) based on algorithms for state estimation, parameter inference and active control.

With this platform, one can investigate model predictive power from different perspectives. How performance degrades when observation times are more distant in time? Answering this question will give us valuable information on the predictive horizon of our models. Can we get better performance with more detailed models? Answering this question enables us to compare the predictive power of different models.

5.2 Proof of principle: controlling gene expression in yeast

To demonstrate the potential of the approach, we considered the problem of controlling gene expression in yeast cells by using a osmoresponsive promoter and applying osmotic stresses to cells. The platform integrates microscopy for monitoring gene expression at the cell level, microfluidics to manipulate the cells environment, and original software for automated imaging, quantification, and online learning or control (Figure 7). The challenges reside in the tight integration of all the platform components and the real time constraint. In particular, the image processing step should be robust enough to be able to track single cells over the full course of the experiment (15hrs) without human assistance. Yet the image analysis process should not last longer than 2 to 3 minutes. All other elements of the platform should comply with the same robustness and efficiency criteria.

An extremely simple model of the osmostress response pathway has been used. We showed that this model alone is not able to propose temporal input profiles that lead to accurate results (Figure 8(E)-(F); open loop framework). However, when one uses observations on the current state of the cell to adapt the control policy, good control performances are obtained (Figure 8(A)-(D)) [95]. To appreciate the difficulty of the control problems that we addressed, one should keep in mind that the controlled system, a yeast cell, is an extremely complex and partially known dynamical system and that the controlled process, gene expression, is intrinsically stochastic. We are currently investigating the effects on control performance of decreasing the sampling rate (in the current framework, observations are made every 6 minutes) or of using more complex models in the controller [97]. This should provide valuable information on the current state of our understanding of the hyperosmotic stress response pathway in yeast.

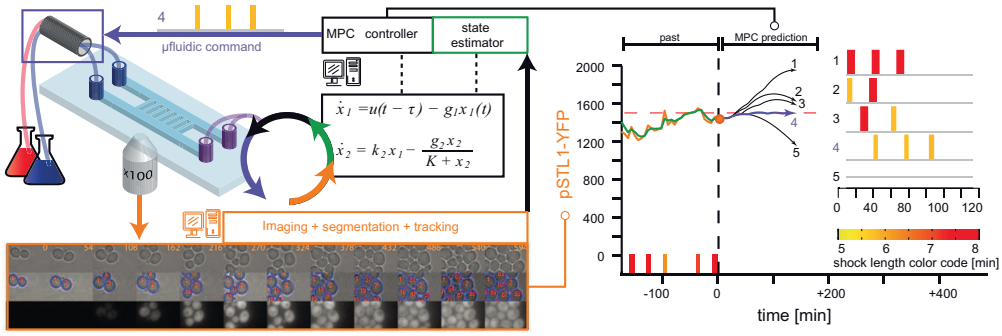


Figure 7: A platform for real-time control of gene expression in yeast. Yeast cells grow as a monolayer in a microfluidic device which enables to rapidly change the cells' osmotic environment (valve, blue frame) and to image their response. After image processing (orange frame) the measured yECitrine fluorescence, either of a single cell or of the mean of all cells, is sent to a state estimator connected to a MPC controller. A model (center, black frame) of pSTL1 induction is used to find the best possible series of osmotic pulses to apply in the future so that the predicted yECitrine level follows a target profile. At the present time point (orange disk), the system state is estimated (green) and the MPC searches for the best input profile for the next 2 hours (blue curve). The selected hyperosmotic profile is sent to the microfluidic command. This control loop is iterated every 6 minutes.

5.3 Significance and perspectives

In addition to be a valuable tool to investigate the importance of various factors on the predictivity of models, our real-time control platform is a unique tool for systems biologists to realize well-controlled physiological modifications of the level of proteins in live cells. Stated differently, this enables biologists to perturb cellular processes with an unprecedented accuracy. This can be a very important contribution to dissect the functionning of many biological processes, since identification theory clearly indicates, that the possibilities to understand (ie identify) a process is limited by the possibilities to perturb it (notion of practical identifiability) [96]. This platform also offer perspectives for synthetic biology applications. Indeed, one could separate the actual biological processes that are of interest for the biotechnology or medical application from its control. Indeed, so far, the goal of biologists is to engineer cells that implement a desired function in a fully automated way, meaning that process and control were implemented within cells. However, in many cases it turned out that implementing complex control functions in cells was challenging. Therefore, by offering the possibility to externalize part of the problem, we might offer important solutions to synthetic biology.

Despite the fact that the importance of control theory for systems and synthetic biology has been widely recognized for more than a decade [24, 50], the actual use of *in silico* feedback loops to control intracellular processes has only been proposed recently. In 2011, we were the first to show that the signaling activity in live yeast cells can be controlled by an in silico feedback loop [94]. Using a proportional-integral (PI) controller we controlled the output of a signal transduction pathway by modulating the osmotic environment of cells in real time. More recently, Toettcher et al. used elaborate microscopy techniques and opto-genetics to control in real time and at the single cell level

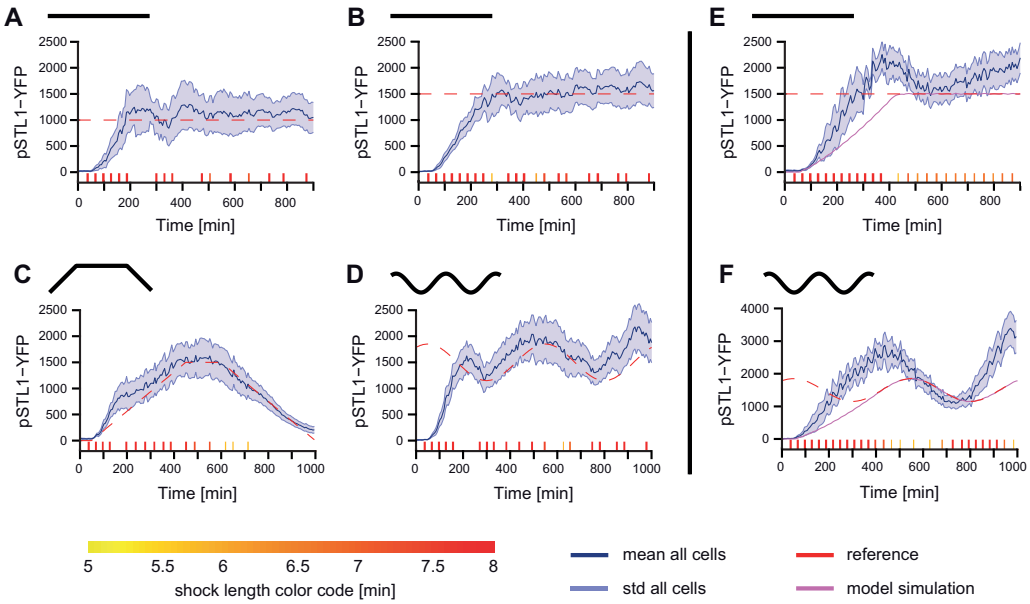


Figure 8: Real-time control of gene expression can be achieved at the population level. (A) and (B) Set-point control experiments with different target values (red dashed line). The timeline of osmotic events is shown at the bottom of each graph (see color code for shock durations, bottom). Shock starting times and durations are computed in real-time. The measured mean cell fluorescence is shown as solid blue lines. The enveloppes indicate standard deviation of the fluorescence distribution across the yeast population. (C) and (D) Tracking control experiments. In both cases, the mean level of fluorescence successfully follows the time-varying target profile. (E) and (F) Open-loop control experiments. Two examples of open-loop control (the osmotic inputs were computed using our model, before starting the experiments) showing poor control quality. Errors accumulate over time. The simulated behavior of the system is represented in violet.

the localization and activity of a signal transduction protein (PI3K) in eukaryotic cells [92]. Also using optogenetic techniques, Milias-Argeitis et al. managed to control the expression of a yeast gene to a constant target value over several hours [63]. Their approach is based on a chemostat culture and is therefore better adapted to biotechnological applications than to probing biological processes for single-cell quantitative biology applications. These works have been reviewed in Chen et al. [20].

Long-term model predictive control of gene expression at the population and single-cell levels (2012)
J. Uhlendorf, A. Miermont, T. Delaveau, G. Charvin, F. Fages, S. Bottani, G. Batt and P. Hersen
PNAS, 109(35):14271-14276

Gene expression plays a central role in the orchestration of cellular processes. The use of inducible promoters to change the expression level of a gene from its physiological level has significantly contributed to the understanding of the functioning of regulatory networks. However, from a quantitative point of view, their use is limited to short-term, population-scale studies to average out cell-to-cell variability and gene expression noise and limit the nonpredictable effects of internal feedback loops that may antagonize the inducer action. Here,we showthat, by implementing an external feedback loop, one can tightly control the expression of a gene over many cell generations with quantitative accuracy. To reach this goal, we developed a platform for real-time, closed-loop control of gene expression in yeast that integrates microscopy for monitoring gene expression at the cell level, microfluidics to manipulate the cells? environment, and original software for automated imaging, quantification, and model predictive control. By using an endogenous osmostress responsive promoter and playing with the osmolarity of the cells environment, we show that long-term control can, indeed, be achieved for both time-constant and time-varying target profiles at the population and even the single-cell levels. Importantly, we provide evidence that real-time control can dynamically limit the effects of gene expression stochasticity. We anticipate that our method will be useful to quantitatively probe the dynamic properties of cellular processes and drive complex, synthetically engineered networks.

6 Conclusion and future directions of research

"For success, attitude is equally as important as ability."
Walter Scott

6.1 Cells as members of a population

The investigation of the functioning of cells at the molecular level has so far mostly been addressed at the cell population level. Standard molecular biology techniques provide population-averaged measurements of cellular compounds and activities. Yet, recently, great progress has been been made in single-cell measurements, revealing a significant cell-to-cell variability [66, 87]. Although cell-to-cell variability at the molecular level does not always generate physiological differences [2], it has been shown in a few cases at least that variability must be accounted for to explain certain cellular processes [68, 34].

One can distinguish two different causes of "noisy" cell behaviors. The first and most obvious cause is the stochasticity of biological processes [66, 87]. Indeed at the molecular level, biological reactions rely on the stochastic encountering of individual molecules. The second cause is unobserved deterministic factors. Differences in initial molecular content, in cell size, in cell age, or in local cell density can cause heterogeneous cell responses to an homogeneous stimulation, and therefore an apparently noisy behavior [85, 99, 49]. In any observed biological process, biological variability originates from both sources, with one possibly dominating the other. It is important to be able to distinguish these two types of variability, since a lot of biological knowledge can be learned by identifying unknown deterministic causes. Modeling can help to disentangle those two types of noise.

However, from a modeling and system identification point of view distinguishing these noises is challenging [41, 93]. Not only it necessitates appropriate high quality data but also effective parameter estimation methods. Indeed the stochastic nature of reactions is generally captured by a class of stochastic models based on the chemical master equation, and the unknown deterministic influences are generally captured by models with parameter distributions. The problem of parameter estimation of stochastic models with multidimensional parameter distributions is open. Actually, approximate methods have been recently developed for the first extreme case, where all the variability is assumed to exclusively originate from molecular noise, with good performances [67, 97]. However, no efficient method has been proposed so far for the estimation of deterministic models with parameter distribution, the other extreme case, based on single cell data. One should note that the method used by Zechner and colleagues does support the identification of models mixing the two sources of variability mentioned above [97]. However, no analysis has been made to evaluate the capability of the proposed approach to appropriately proportionate the two influences on the overall variability.

The naive approach for the identification of deterministic models with parameter distributions would be to fit a model to the mean behavior and from this fit, refit the model to single cell data. This way, one would obtain a set of "individual-cell" parameters. One could then identify the multi-dimensional distributions that describe the identified set of parameters. The major drawback of such an approach is that one has no guaranty that this gives an acceptable model of the population. Indeed if one resamples parameters in the identified distributions, and computes the simulated population behaviors, large deviations are encountered. With the Ferari-Trecate group (Pavia Univ.), we investigate the use of mixed effect parameter identification methods to identify parameter distributions from single cell videomicroscopy data generated in the Hersen lab (CNRS/Paris 7). Importantly mixed effect methods capture parameter multidimensional correlations and search for distributions that fit the behaviors of the entire population [29]. This will be critical for the identification of biologically meaningful models.

This framework then enables to assign specific parameter values to specific cells. Therefore this directly addresses questions related to cell individuality. How effective are single cell models in predicting the cell behavior? For how long is the predictive power of the model better than the one of the mean model, or stated differently how long is this individuality preserved? What are the connections with protein mixing times as introduced by Sigal et al [83]? We investigate these

questions in the context of long-term prediction of a cell population subjected to repeated TRAIL applications as described in more details in the following section.

6.2 Cells within their environment

Environment matters. It is clear that growth conditions affect the cell physiology and hence all biomolecular processes. However, how and to which degree is a specific biological process affected by environmental changes? This question is too often neglected. In bacterial systems it has been shown that the probability of switching from growth on glucose to growth on lactose depends on cellular growth rate and is not purely stochastic as postulated earlier [77]. At a much larger level, it has been shown that gene expression is globally affected by growth conditions, and that this global influence plays a major regulatory role in the orchestration of the adaptation response [15]. This has consequences in systems and synthetic biology applications since biological systems are often analyzed or developed in conditions that are different from the standard, natural or operating, conditions. In eukaryotic cells, and most notably in mammalian cells, endogenous and ectopic genes are subject to epigenetic modifications and silencing [51]. Growth conditions, such as possible oxidative stresses, are known factors that influence the epigenetic status of genes. But the main determinants are still unknown. This is a major issue for the development of predictive functional systems in mammalian synthetic biology. Similarly, the contribution of environmental changes to the orchestration of cell responses in the human body is de facto neglected in most of the in vitro studies. This may severely hamper our capacity to understand and interfere with cell functioning for systems or pharmaceutical biology.

To detect the influence of environmental factors and assess their impact, quantitative approaches are needed. To obtain a quantitative understanding of the system in its changing environment, one needs to model the system *and* its environments, and to obtain the corresponding data. In what follows, I will describe two problems and for each of those, envision the approaches that can be developed.

The first problem that we considered is the creation of a patterning system. More precisely, we consider engineered yeast cells derived from [19] whose growth depends on a small diffusible molecule, IP, in a band pass manner. That is, growth is possible only when the IP concentration is within given bounds. In effect, two "killing modules", a low threshold and a high threshold, have been implemented to trigger cell death outside the desired IP range. Moreover, these cells have been engineered to produce IP in an inducible manner. Therefore, in principle this system could exhibit patterning capabilities on solid media, typically agar plates. Indeed different initial seedings will result in different non trivial configurations of the system in time. In this project, our objective is to develop models of the intracellular synthetic network and of cell growth in solid environments, tune these models based on data collected independently, and test the accuracy of our predictions for the full system. Deviations from model predictions will indicate differences in the functioning of the system.

The development of such systems is difficult. Indeed, one typically considers a dynamical system with a dynamical structure. Indeed, unlike standard problems, the structure of the system itself (ie cell number and cell locations) is changing with time. This cannot be neglected. Moreover, the scale of the phenomena spans several orders of magnitude, from the cell size to the Petri dish. Therefore one should employ multiscale methods. This work is done in collaboration with the groups of Ron Weiss (MIT) for the synthetic biology constructions and of Dirk Drado (INRIA) for the cell-based spatial simulations.

A second problem of interest is understand how the geometry of the cell population affects tissue response. In most studies the cell response is characterized in monolayer conditions. This notably gives access to single cell behaviors. However, in vivo, cells adopt more complex 3D organizations. For example, cancer cells form spheroids at the early tumor stages. Such spheroids can be recreated in vitro but cell observations is more difficult. This explains at least partly why drug testing and drug treatment optimization is mostly made in vitro on monolayer [55]. However, it is unclear how those results will transpose to spheroids or even to in vivo tumors. Indeed, the 3D structuration of the tissue affects the physical accessibility of cells in the tissue (molecular diffusion) and possibly the cell physiology as well (contact inhibition). Following the strategy employed in the previous project, the

idea here is to combine a model of cell death following anti-cancerous treatment calibrated based on monolayer cell cultures with a 3D model of spheroid growth and molecule diffusion and compare model predictions with experimental data. Deviations from predictions will indicate important differences between cell responses in different conditions. These findings will likely guide the development of novel therapeutic strategies.

Existing 3D models of tumors growth have already been developed by collaborators in the Bang research group (Dirk Drasdo Multi-cellular Systems group) [39, 32, 47]. A challenging missing piece is a model of long term cell response to drug treatments. For example, even if TRAIL is one of the most studied and best characterized death-inducing molecule [31], the quantitative understanding of the effect of repeated TRAIL additions on (monolayer) cells is still missing [37]. It has been shown that accounting for cell-to-cell variability was essential for explaining the observed variable delay in the times of death following Trail treatment [86]. However, the solution proposed to implement cell-to-cell variability does not allow for long term predictions: cell heterogeneity is modeled using distributions on protein concentrations, serving as initial conditions of the simulation. Clearly, this heterogeneity is cast into stone at the beginning and cannot be regenerated with time in the surviving population. Therefore, we extended this model with stochastic gene expression processes generating the same steady state distributions but able to dynamically capture protein fluctuations. Preliminary results suggest that our extended model is able to capture the observed reversible resistance of the population of surviving cells. Equipped with these two models one can test consistency between model predictions and recently published data on spheroids TRAIL treatments [57]. The development and validation of such tools will hopefully prove valuable for many therapeutic studies.

6.3 A platform for well-controlled physiological perturbations

To provide means to better control intracellular processes, we have developed a platform for real-time control of gene expression. As described in the previous chapter this platform enables to control the concentration of a protein in a time-varying manner at the single-cell level with unprecedented accuracy. At the same time a few other works have been published on this problem using different approaches and different focuses but with comparable accuracy. Collectively, these works have attracted quite some attention from the community and press[3]. Possible extensions can be classified in two groups: further methodological developments and novel applications. In what follows, I briefly present perspectives for each research direction.

Methodological developments are needed to produce data of even higher quality and develop models that make even better use of the available data. Getting quantitative readout, devoid of biological or optical artifacts, for each cell along the entire experiment in an automated manner necessitates in fact non-trivial image processing techniques and tools. Excellent image segmentation, tracking, and whenever possible, lineage reconstruction are needed is one wants to get biologically relevant conclusions. We are working with the BioComputing group in Lille in this direction. The use of the novel optogenetics methods in place of osmotic stresses to trigger gene expression would also be beneficial since it would limit the influence of the input on the cell under investigation (better orthogonality) [5].

To make better use of available data during control experiments one needs to have efficient state reconstruction methods and control algorithms. In collaboration with Eugenio Cinquemani (IBIS group, INRIA) and Alessandro Abate (Oxford Univ.) we develop methods for stochastic systems. We expect that these methods will outperform their deterministic counterparts in conditions of single cell control. This would be the first experimental demonstration that stochastic models and methods improve our prediction capabilities at the single cell level. A second direction for improving single cell control is to tune the parameters of deterministic models to the individual cell that is controlled based on either mixed effect model parameter distributions or online learning methods.

The second main research direction deals with novel applications. Our publication in PNAS was a proof of concept for real-time control in yeast. It simply showed that closed loop control was possible

3. *'Cyborg' yeast genes run by computer* appeared in BBC news and *Une étape de plus vers la pleine maîtrise du vivant* appeared in l'Humanité Dimanche describing the works of Lygeros, Khammash, El Samad and colleagues, and Hersen, Batt and colleagues, respectively

with good accuracy in a simple eukaryotic cell. Going beyond the proof of concept and demonstrating that real-time control can be developed for higher eukaryotic cells are two natural extensions of our previous results.

In the context of the INRIA/INSERM "action d'envergure" project that notably aims at understanding the connection between the availability of the transcription machinery and the cell physiology and growth, we will "clamp" the level of key transcriptional factors for extended duration and observe the cellular effects. Closed loop control is motivated by the presence of endogenous feedback loops (at the very least, the transcription machinery components need to be transcribed). Without our control platform, the quantitative analysis of the long term effects of transcription deficiency can hardly be investigated. This project is done in collaboration with the group of Hidde de Jong and Hans Geiselman (INRIA Grenoble – Rhône-Alpes and CNRS/Grenoble University) who have been working for several years on the global regulation of gene expression in *E. coli*.

In the context of the ANR Investissement d'Avenir project Iceberg, we investigate real-time control of gene expression in mammalian cells. In close collaboration with the group of Pascal Hersen (CNRS/Paris7), and with four other partners, we are developing cell lines that enable us to observe and control gene expression in a reliable manner. One critical issue is to design and construct an induction system that is responsive enough to get interesting dynamics at the time scale of a cell cycle and for many cell generations. To develop this system, we will base our work on a "landing pad" technology developed with the Weiss lab (MIT). This platform uses recombinases and enables the efficient integration of a complex genetic construct at a unique and targeted position in the genome [33]. All the other elements of the platform (microfluidic device and microscopy for long term experiments, image analysis; modeling and control algorithms)need to be adapted to this new system. Being able to control in live cells protein concentrations in mammalian cells would open a number of interesting research directions for the pharmaceutical industry.

"The idea is to try to give all the information to help others to judge the value of your contribution; not just the information that leads to judgment in one particular direction or another."
Richard P. Feynman

References

[1] Bree B Aldridge, Suzanne Gaudet, Douglas a Lauffenburger, and Peter K Sorger. Lyapunov exponents and phase diagrams reveal multi-factorial control over TRAIL-induced apoptosis. *Molecular Systems Biology*, 7(553):1–21, 2011.

[2] Steven J Altschuler and Lani F Wu. Cellular heterogeneity: do differences make a difference? *Cell*, 141(4):559–63, 2010.

[3] Ernesto Andrianantoandro, Subhayu Basu, David K Karig, and Ron Weiss. Synthetic biology: new engineering rules for an emerging discipline. *Molecular Systems Biology*, 2:2006.0028, 2006.

[4] T. Ali Azam, A. Iwata., A. Nishimura, S. Ueda, and A. Ishihama. Growth phase-dependent variation in protein composition of the *Escherichia coli* nucleoid. *Journal of Bacteriology*, 181(20):6361–6370, 1999.

[5] William Bacchus and Martin Fussenegger. The use of light for engineered control and reprogramming of cellular functions. *Current opinion in biotechnology*, 23(5):695–702, 2012.

[6] G. Batt, C. Belta, and R. Weiss. Model checking genetic regulatory networks with parameter uncertainty. In A. Bemporad, A. Bicchi, and G. Buttazzo, editors, *Proceedings of the Tenth International Workshop on Hybrid Systems: Computation and Control, HSCC'07*, Lecture Notes in Computer Science. Springer, 2007.

[7] G. Batt, C. Belta, and R. Weiss. Model checking liveness properties of genetic regulatory networks. In O. Grumberg and M. Huth, editors, *Proceedings of the Thirteenth International Conference on Tools and Algorithms for the Construction and Analysis of Systems, TACAS'07*, Lecture Notes in Computer Science. Springer, 2007.

[8] G. Batt, B. Besson, P.-E. Ciron, H. de Jong, E. Dumas, J. Geiselmann, R. Monte, P.T. Monteiro, M. Page, F. Rechenmann, and D. Ropers. Genetic Network Analyzer : A tool for the qualitative modeling and simulation of bacterial regulatory networks. In J. van Helden, A. Toussaint, and D. Thieffry, editors, *Bacterial Molecular Networks*, pages 439–462. Humana Press, Springer, 2012.

[9] G. Batt, H. de Jong, J. Geiselmann, and M. Page. Analysis of genetic regulatory networks: a model-checking approach. In M. Benerecetti and C. Pecheur, editors, *Working Notes of the Second Workshop on Model Checking and Artificial Intelligence, MoChArt'03*, pages 51–58, Acapulco, Mexico, 2003.

[10] G. Batt, H. de Jong, J. Geiselmann, and M. Page. Analysis of genetic regulatory networks: a model-checking approach. In P. Salles and B. Bredeweg, editors, *Proceedings of the Seventeenth International Workshop on Qualitative Reasoning, QR'03*, pages 31–38, Brasilia, Brazil, 2003.

[11] G. Batt, M. Page, I. Cantone, G. Goessler, P. Monteiro, and H. de Jong. Efficient parameter search for qualitative models of regulatory networks using symbolic model checking. *Bioinformatics*, 26(18):i603–i610, 2010.

[12] Gregory Batt, Calin Belta, and Ron Weiss. Temporal Logic Analysis of Gene Networks Under Parameter Uncertainty. *IEEE Transactions on Automatic Control and IEEE Transactions on Circuits and Systems I*, 53(Joint Special Issue on Systems Biology):215–229, 2008.

[13] Grégory Batt, Boyan Yordanov, Ron Weiss, and Calin Belta. Robustness analysis and tuning of synthetic gene networks. *Bioinformatics*, 23(18):2415–22, 2007.

[14] C. Belta and L.C.G.J.M. Habets. Controlling a class of nonlinear systems on rectangles. *Transactions on Automatic Control*, 51(11):1749–1759, 2006.

[15] Sara Berthoumieux, Hidde de Jong, Guillaume Baptist, Corinne Pinel, Caroline Ranquet, Delphine Ropers, and Johannes Geiselmann. Shared control of gene expression in bacteria by transcription factors and global physiology of the cell. *Molecular Systems Biology*, 9(634):634, 2013.

[16] Frank J Bruggeman and Hans V Westerhoff. The nature of systems biology. *Trends in Microbiology*, 15(1):45–50, 2007.

[17] J.R. Burch, E.M. Clarke, K.L. McMillan, D.L. Dill, and L.J. Hwang. Symbolic model checking: 10^{20} states and beyond. In *Proceedings of the Fifth Annual IEEE Symposium on Logic in Computer Science, LICS'90*, pages 1–33. IEEE Computer Society Press, 1990.

[18] Irene Cantone, Lucia Marucci, Francesco Iorio, Maria Aurelia Ricci, Vincenzo Belcastro, Mukesh Bansal, Stefania Santini, Mario di Bernardo, Diego di Bernardo, and Maria Pia Cosma. A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches. *Cell*, 137(1):172–81, 2009.

[19] Ming-Tang Chen and Ron Weiss. Artificial cell-cell communication in yeast Saccharomyces cerevisiae using signaling elements from Arabidopsis thaliana. *Nature Biotechnology*, 23(12):1551–5, 2005.

[20] Susan Chen, Patrick Harrigan, Benjamin Heineike, Jacob Stewart-Ornstein, and Hana El-Samad. Building robust functionality in synthetic circuits using engineered feedback regulation. *Current opinion in biotechnology*, 24(4):1–7, 2013.

[21] A. Cimatti, E.M. Clarke, E. Giunchiglia, F. Giunchiglia, M. Pistore, M. Roveri, R. Sebastiani, and A. Tacchella. NuSMV2: An opensource tool for symbolic model checking. In E. Brinksma and K.G. Larsen, editors, *Proceedings of the Fourteenth International Conference on Computer Aided Verification, CAV'02*, volume 2404 of *Lecture Notes in Computer Science*, pages 359–364. Springer, Copenhagen, Denmark, 2002.

[22] E.M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith. Progress on state explosion problem in model checking. In R. Wilhelm, editor, *Informatics. 10 years back. 10 years ahead.*, volume 2000 of *Lecture Notes in Computer Science*, pages 176–194. Springer, 2001.

[23] E.M. Clarke, O. Grumberg, and D.A. Peled. *Model Checking*. The MIT Press, Cambridge, MA, 1999.

[24] Marie E Csete and John C Doyle. Reverse engineering of biological complexity. *Science*, 295(5560):1664–9, 2002.

[25] Maria I Davidich and Stefan Bornholdt. Boolean network model predicts cell cycle sequence of fission yeast. *PloS one*, 3(2):e1672, 2008.

[26] H. de Jong, J. Geiselmann, G. Batt, C. Hernandez, and M. Page. Qualitative simulation of the initiation of sporulation in *Bacillus subtilis*. *Bulletin of Mathematical Biology*, 66(2):261–299, 2004.

[27] H. de Jong, J. Geiselmann, C. Hernandez, and M. Page. Genetic Network Analyzer: Qualitative simulation of genetic regulatory networks. *Bioinformatics*, 19(3):336–344, 2003.

[28] 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. *Bulletin of Mathematical Biology*, 66(2):301–340, 2004.

[29] Bernard Delyon, Marc Lavielle, and Eric Moulines. Convergence of a stochastic approximation version of the EM algorithm. *The Annals of Statistics*, 27(1):94–128, 1999.

[30] Barbara Di Ventura, Caroline Lemerle, Konstantinos Michalodimitrakis, and Luis Serrano. From in vivo to in silico biology and back. *Nature*, 443(7111):527–33, 2006.

[31] L Y Dimberg, C K Anderson, R Camidge, K Behbakht, a Thorburn, and H L Ford. On the TRAIL to successful cancer therapy? Predicting and counteracting resistance against TRAIL-based therapeutics. *Oncogene*, 32(11):1341–50, 2013.

[32] Dirk Drasdo and Stefan Höhme. A single-cell-based model of tumor growth in vitro: monolayers and spheroids. *Physical biology*, 2(3):133–47, 2005.

[33] Xavier Duportet, Liliana Wroblewska, Patrick Guye, Yinqing Li, Justin Eyquem, Julianne Rieders, Gregory Batt, and Ron Weiss. Targeted efficient integration of large multi-unit genetic payloads in mammalian cells. 2013. submitted.

[34] Avigdor Eldar and Michael B Elowitz. Functional roles for noise in genetic circuits. *Nature*, 467(7312):167–73, 2010.

[35] Adrien Fauré, Aurélien Naldi, Claudine Chaouiya, and Denis Thieffry. Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle. *Bioinformatics*, 22(14):e124–31, 2006.

[36] Jasmin Fisher, David Harel, and Thomas a. Henzinger. Biology as reactivity. *Communications of the ACM*, 54(10):72, 2011.

[37] Deborah A Flusberg, Jérémie Roux, Sabrina L Spencer, and Peter K Sorger. Cells surviving fractional killing by TRAIL exhibit transient but sustainable resistance and inflammatory phenotypes. *Mol Biol Cell*, 24(14):2186–2200, 2013.

[38] Marc Folcher and Martin Fussenegger. Synthetic biology advancing clinical applications. *Current opinion in chemical biology*, 16(3-4):345–54, 2012.

[39] Jörg Galle, Markus Loeffler, and Dirk Drasdo. Modeling the effect of deregulated proliferation and apoptosis on the growth dynamics of epithelial cell populations in vitro. *Biophysical journal*, 88(1):62–75, 2005.

[40] Daniel G Gibson, John I Glass, Carole Lartigue, Vladimir N Noskov, Ray-Yuan Chuang, Mikkel a Algire, Gwynedd a Benders, Michael G Montague, Li Ma, Monzia M Moodie, Chuck Merryman, Sanjay Vashee, Radha Krishnakumar, Nacyra Assad-Garcia, Cynthia Andrews-Pfannkoch, Evgeniya a Denisova, Lei Young, Zhi-Qing Qi, Thomas H Segall-Shapiro, Christopher H Calvey, Prashanth P Parmar, Clyde a Hutchison, Hamilton O Smith, and J Craig Venter. Creation of a bacterial cell controlled by a chemically synthesized genome. *Science*, 329(5987):52–6, 2010.

[41] Andres M Gonzalez, Jannis Uhlendorf, Eugenio Cinquemani, Gregory Batt, and Giancarlo Ferrari-trecate. Identification of biological models from single-cell data: A comparison between mixed-effects and moment-based inference. In *European Control Conference, ECC'13*, 2013.

[42] J.-L. Gouzé and T. Sari. A class of piecewise-linear differential equations arising in biological models. *Dynamical Systems*, 17(4):299–316, 2002.

[43] Patrick Guye, Yinqing Li, Liliana Wroblewska, Xavier Duportet, and Ron Weiss. Rapid, modular and reliable construction of complex mammalian gene circuits. *Nucleic Acids Research*, 41(16):1–6, 2013.

[44] L.C.G.J.M. Habets, P.J. Collins, and J.H. van Schuppen. Reachability and control synthesis for piecewise-affine hybrid systems on simplices. *IEEE Transactions on Automatic Control*, 51(6):938–948, 2006.

[45] R David Hawkins, Gary C Hon, and Bing Ren. Next-generation genomics: an integrative approach. *Nature Reviews Genetics*, 11(7):476–86, 2010.

[46] William S Hlavacek, James R Faeder, Michael L Blinov, Richard G Posner, Michael Hucka, and Walter Fontana. Rules for modeling signal-transduction systems. *Science STKE*, 2006(344):re6, 2006.

[47] Stefan Hoehme and Dirk Drasdo. A cell-based simulation software for multi-cellular systems. *Bioinformatics*, 26(20):2641–2, 2010.

[48] S. Hooshangi, S. Thiberge, and R. Weiss. Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proceedings of the National Academy of Sciences of the USA*, 102(10):3581–3586, 2005.

[49] Sui Huang. Non-genetic heterogeneity of cells in development: more than just noise. *Development*, 136(23):3853–62, 2009.

[50] Pablo A Iglesias and Brian P Ingalls. *Control Theory and Systems Biology*. The MIT Press, 2009.

[51] Rudolf Jaenisch and Adrian Bird. Epigenetic regulation of gene expression: how the genome integrates intrinsic and environmental signals. *Nature Genetics*, 33 Suppl(March):245–54, 2003.

[52] Ahmad S Khalil and James J Collins. Synthetic biology: applications come of age. *Nature Reviews Genetics*, 11(5):367–79, 2010.

[53] H. Kitano. Towards a theory of biological robustness. *Molecular Systems Biology*, 3:137, 2007.

[54] G.-J. Kremers, S. G. Gilbert, P. J. Cranfill, M. W. Davidson, and D. W. Piston. Fluorescent proteins at a glance. *Journal of Cell Science*, 124(15):2676–2676, 2011.

[55] Michael J Lee, Albert S Ye, Alexandra K Gardino, Anne Margriet Heijink, Peter K Sorger, Gavin MacBeath, and Michael B Yaffe. Sequential application of anticancer drugs enhances cell death by rewiring apoptotic signaling networks. *Cell*, 149(4):780–94, 2012.

[56] Kevin D. Litcofsky, Raffi B. Afeyan, Russell J. Krom, Ahmad S. Khalil, and James J. Collins. Iterative plug-and-play methodology for constructing and modifying synthetic gene networks. *Nature Methods*, 9(11):1077–1080, 2012.

[57] Hui-li Ma, Qiao Jiang, Siyuan Han, Yan Wu, Jin Cui Tomshine, Dongliang Wang, Yaling Gan, and Guozhang Zou. Multicellular tumor spheroids as an in vivo-like tumor model for three-dimensional imaging of chemotherapeutic and nano material cellular penetration. *Molecular Imaging*, 11(6):487–498, 2012.

[58] A F Maarten Altelaar, Javier Munoz, and Albert J R Heck. Next-generation proteomics: towards an integrative view of proteome dynamics. *Nature Reviews Genetics*, 14(1):35–48, 2013.

[59] Oded Maler and Gregory Batt. Approximating Continuous Systems by Timed Automata. In *Formal Methods in Systems Biology, FMSB'08*, pages 77–89. Springer Verlag, 2008.

[60] Nuno D Mendes, Frédéric Lang, Yves-Stan Le Cornec, Radu Mateescu, Gregory Batt, and Claudine Chaouiya. Composition and abstraction of logical regulatory modules: application to multicellular systems. *Bioinformatics*, 29(6):749–57, 2013.

[61] L. Mendoza, D. Thieffry, and E.R. Alvarez-Buylla. Genetic control of flower morphogenesis in *Arabidopsis thaliana*: A logical analysis. *Bioinformatics*, 15(7-8):593–606, 1999.

[62] T. Mestl, E. Plahte, and S.W. Omholt. A mathematical framework for describing and analysing gene regulatory networks. *Journal of Theoretical Biology*, 176:291–300, 1995.

[63] Andreas Milias-Argeitis, Sean Summers, Jacob Stewart-Ornstein, Ignacio Zuleta, David Pincus, Hana El-Samad, Mustafa Khammash, and John Lygeros. In silico feedback for in vivo regulation of a gene expression circuit. *Nature Biotechnology*, 29(11):1114–1116, 2011.

[64] Ali Mortazavi, Brian A Williams, Kenneth Mccue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods*, 5(7):621–628, 2008.

[65] Brian Munsky, Gregor Neuert, and Alexander van Oudenaarden. Using gene expression noise to understand gene regulation. *Science*, 336(6078):183–7, 2012.

[66] Dale Muzzey and Alexander van Oudenaarden. Quantitative time-lapse fluorescence microscopy in single cells. *Annual review of cell and developmental biology*, 25:301–27, 2009.

[67] Gregor Neuert, Brian Munsky, Rui Zhen Tan, Leonid Teytelman, Mustafa Khammash, and Alexander van Oudenaarden. Systematic identification of signal-activated stochastic gene regulation. *Science*, 339(6119):584–7, 2013.

[68] Mario Niepel, Sabrina L Spencer, and Peter K Sorger. Non-genetic cell-to-cell variability and the consequences for pharmacology. *Current opinion in chemical biology*, 13(5-6):556–61, 2009.

[69] C J Paddon, P J Westfall, D J Pitera, K Benjamin, K Fisher, D McPhee, M D Leavell, A Tai, A Main, D Eng, D R Polichuk, K H Teoh, D W Reed, T Treynor, J Lenihan, M Fleck, S Bajad, G Dang, D Dengrove, D Diola, G Dorin, K W Ellens, S Fickes, J Galazzo, S P Gaucher, T Geistlinger, R Henry, M Hepp, T Horning, T Iqbal, H Jiang, L Kizer, B Lieu, D Melis, N Moss,

R Regentin, S Secrest, H Tsuruta, R Vazquez, L F Westblade, L Xu, M Yu, Y Zhang, L Zhao, J Lievense, P S Covello, J D Keasling, K K Reiling, N S Renninger, and J D Newman. High-level semi-synthetic production of the potent antimalarial artemisinin. *Nature*, 496(7446):528–32, 2013.

[70] Jeremy S Paige, Karen Y Wu, and Samie R Jaffrey. RNA mimics of green fluorescent protein. *Science*, 333(6042):642–6, 2011.

[71] M. Perego and J.A. Hoch. Sequence analysis of the *hpr* locus, a regulatory gene for protease production and sporulation in *Bacillus subtilis*. *Journal of Bacteriology*, 170(6):2560–2567, 1988.

[72] Amir Pnueli. The temporal logic of programs. In *18th Annual Symposium on Foundations of Computer Science, FOCS'77*, pages 46–57, 1977.

[73] A. Richard, J.-P. Comet, and G. Bernot. R. Thomas' modeling of biological regulatory networks: Introduction of singular states in the qualitative dynamics. *Fundamenta Informaticae*, 65(4):373–392, 2005.

[74] Aurélien Rizk, Grégory Batt, François Fages, and Sylvain Soliman. On Temporal Logic Constraint Solving for Analyzing Numerical Data Time Series. In *Computational Methods in Systems Biology, CMSB'08*, number LNCS 5307, pages 251–268. Springer-Verlag, 2008.

[75] Aurélien Rizk, Gregory Batt, François Fages, and Sylvain Soliman. A general computational method for robustness analysis with applications to synthetic gene networks. *Bioinformatics*, 25(12):i169–78, 2009.

[76] Aurélien Rizk, Grégory Batt, François Fages, and Sylvain Soliman. Continuous valuations of temporal logic specifications with applications to parameter optimization and robustness measures. *Theoretical Computer Science*, 412(26):2827–2839, 2011.

[77] Lydia Robert, Gregory Paul, Yong Chen, François Taddei, Damien Baigl, and Ariel B Lindner. Pre-dispositions and epigenetic inheritance in the Escherichia coli lactose operon bistable switch. *Molecular Systems Biology*, 6(1):357, 2010.

[78] D. Ropers, H. de Jong, M. Page, D. Schneider, and J. Geiselmann. Qualitative simulation of the carbon starvation response in *Escherichia coli*. *BioSystems*, 84(2):124–152, 2005.

[79] L. Sánchez and D. Thieffry. A logical analysis of the *Drosophila* gap genes. *Journal of Theoretical Biology*, 211(2):115–141, 2001.

[80] Y Setty, a E Mayo, M G Surette, and U Alon. Detailed map of a cis-regulatory input function. *Proceedings of the National Academy of Sciences of the United States of America*, 100(13):7702–7, 2003.

[81] Jay Shendure and Erez Lieberman Aiden. The expanding scope of DNA sequencing. *Nature Biotechnology*, 30(11):1084–94, 2012.

[82] Jill C Sible and John J Tyson. Mathematical modeling as a tool for investigating cell cycle control networks. *Methods*, 41(2):238–47, 2007.

[83] Alex Sigal, Ron Milo, Ariel Cohen, Naama Geva-Zatorsky, Yael Klein, Yuvalal Liron, Nitzan Rosenfeld, Tamar Danon, Natalie Perzov, and Uri Alon. Variability and memory of protein levels in human cells. *Nature*, 444(7119):643–6, 2006.

[84] Adrian L Slusarczyk, Allen Lin, and Ron Weiss. Foundations for the design and implementation of synthetic genetic circuits. *Nature Reviews Genetics*, 13(6):406–20, 2012.

[85] Berend Snijder and Lucas Pelkmans. Origins of regulated cell-to-cell variability. *Nature Reviews Molecular cell biology*, 12(2):119–25, 2011.

[86] Sabrina L Spencer, Suzanne Gaudet, John G Albeck, John M Burke, and Peter K Sorger. Non-genetic origins of cell-to-cell variability in TRAIL-induced apoptosis. *Nature*, 459(7245):428–32, 2009.

[87] David G Spiller, Christopher D Wood, David a Rand, and Michael R H White. Measurement of single-cell dynamics. *Nature*, 465(7299):736–45, 2010.

[88] Szymon Stoma, Alexandre Donzé, François Bertaux, Oded Maler, and Gregory Batt. STL-based analysis of TRAIL-induced apoptosis challenges the notion of type I/type II cell line classification. *PLoS computational biology*, 9(5):e1003056, 2013.

[89] M.A. Strauch and J.A. Hoch. Transition-state regulators: Sentinels of *Bacillus subtilis* post-exponential gene expression. *Molecular Microbiology*, 7(3):337–342, 1993.

[90] David M Suter, Nacho Molina, David Gatfield, Kim Schneider, Ueli Schibler, and Felix Naef. Mammalian Genes Are Transcribed with Widely Different Bursting Kinetics. *Science*, 332(6028):472–474, 2011.

[91] R. Thomas, D. Thieffry, and M. Kaufman. Dynamical behaviour of biological regulatory networks: I. Biological role of feedback loops and practical use of the concept of the loop-characteristic state. *Bulletin of Mathematical Biology*, 57(2):247–276, 1995.

[92] Jared E Toettcher, Delquin Gong, Wendell A Lim, and Orion D Weiner. Light-based feedback for controlling intracellular signaling dynamics. *Nature Methods*, 8(09):837–839, 2011.

[93] Tina Toni and Bruce Tidor. Combined model of intrinsic and extrinsic variability for computational network design with application to synthetic biology. *PLoS computational biology*, 9(3):e1002960, 2013.

[94] Jannis Uhlendorf, Samuel Bottani, François Fages, Pascal Hersen, and Gregory Batt. Towards real-time control of gene expression: controlling the hog signaling cascade. *Pacific Symposium on Biocomputing*, pages 338–349, January 2011.

[95] Jannis Uhlendorf, Agnès Miermont, Thierry Delaveau, Gilles Charvin, François Fages, Samuel Bottani, Gregory Batt, and Pascal Hersen. Long-term model predictive control of gene expression at the population and single-cell levels. *Proceedings of the National Academy of Sciences of the United States of America*, 109(35):14271–6, 2012.

[96] Eric Walter and Luc Pronzato. *Identification of Parametric Models from Experimental Data*. Springer-Verlag, 1997.

[97] C. Zechner, J. Ruess, P. Krenn, S. Pelet, M. Peter, J. Lygeros, and H. Koeppl. Moment-based inference predicts bimodality in transient gene expression. *Proceedings of the National Academy of Sciences of the United States of America*, 109(21), 2012.

[98] Weiwen Zhang, Feng Li, and Lei Nie. Integrating multiple 'omics' analysis for microbial biology: application and methodologies. *Microbiology*, 156(Pt 2):287–301, 2010.

[99] C J Zopf, Katie Quinn, Joshua Zeidman, and Narendra Maheshri. Cell-cycle dependence of transcription dominates noise in gene expression. *PLoS computational biology*, 9(7):e1003161, 2013.

BIOINFORMATICS

Vol. 21 Suppl. 1 2005, pages i19–i28
doi:10.1093/bioinformatics/bti1048

Validation of qualitative models of genetic regulatory networks by model checking: analysis of the nutritional stress response in Escherichia coli

Grégory Batt[1], Delphine Ropers[1], Hidde de Jong[1,], Johannes Geiselmann[2], Radu Mateescu[1], Michel Page[1,3] and Dominique Schneider[2]*

[1]INRIA Rhône-Alpes, Montbonnot, France, [2]Laboratoire Adaptation et Pathogénie des Microorganismes, CNRS UMR 5163, Université Joseph Fourier, Grenoble, France and [3]Université Pierre Mendès France, Grenoble, France

Received on January 15, 2005; accepted on March 27, 2005

ABSTRACT

Motivation: The modeling and simulation of genetic regulatory networks have created the need for tools for model validation. The main challenges of model validation are the achievement of a match between the precision of model predictions and experimental data, as well as the efficient and reliable comparison of the predictions and observations.

Results: We present an approach towards the validation of models of genetic regulatory networks addressing the above challenges. It combines a method for qualitative modeling and simulation with techniques for model checking, and is supported by a new version of the computer tool Genetic Network Analyzer (GNA). The model-validation approach has been applied to the analysis of the network controlling the nutritional stress response in *Escherichia coli*.

Availability: GNA and the model of the stress response network are available at http://www-helix.inrialpes.fr/gna

Contact: Hidde.de-Jong@inrialpes.fr

1 INTRODUCTION

The functioning and development of living organisms is controlled by large and complex networks of genes, proteins, small molecules and their mutual interactions, the so-called *genetic regulatory networks*. In order to gain an understanding of how the behavior of an organism, e.g. the response of a bacterial cell to a physiological or genetic perturbation, emerges from such a network of interactions, we need mathematical and computational tools for modeling and simulation (de Jong, 2002). The predictions obtained through the application of these tools have to be confronted with experimental data. This gives rise to the problem of *model validation*, the assessment of the adequacy of a model by comparing its predictions with observations, either already available in the literature or obtained through novel experiments suggested by the model.

The main challenges of model validation are twofold. First of all, the precision of the model predictions and the experimental data need to be brought in agreement. At present, quantitative information on kinetic parameters is usually absent, thus making traditional numerical models and analysis techniques difficult to apply. In addition, numerical predictions on the dynamics of the system are difficult to verify, because available data are mostly qualitative in nature. A second challenge is to ensure that the comparison of model predictions with experimental data is efficient and reliable. Models of genetic regulatory networks of biological interest may become quite large, as they include many genes and proteins, thus making manual verification of dynamical properties error-prone or even practically infeasible.

In this paper, we propose an approach towards model validation addressing the above two challenges. The approach extends our previous work on a method for the *qualitative modeling and simulation* of genetic regulatory networks, supported by the computer tool *Genetic Network Analyzer* (GNA) (de Jong *et al*., 2003, 2004). This method is based on a class of *piecewise-linear* (PL) *differential equations* that permits a coarse-grained, qualitative analysis of the network dynamics to be carried out. Instead of numerical values for the parameters, the method uses inequality constraints that can be inferred from the experimental literature. It yields predictions on the possible ways in which the sign pattern of the derivatives of the protein concentrations can evolve, a level of precision that is well-adapted to currently-available data. The novelty of the model-validation approach is that it integrates qualitative modeling and simulation with *model-checking* techniques (Clarke *et al*., 1999) to verify whether the predictions of the system behavior are consistent with experimental data.

*To whom correspondence should be addressed.

In particular, the measured evolution of the derivative sign pattern or other experimental observations can be formalized as properties in temporal logic, while model-checking techniques verify whether the predictions account for these properties. If they do not, then the model is inconsistent with the experimental data and may need to be revised or extended. The combination of qualitative modeling and simulation and model-checking allows large and complex networks to be verified, with the guarantee that no model is falsely ruled out.

Model-checking or other formal verification techniques have been used before in systems biology for analyzing genetic, metabolic, signal-transduction and cell-cycle networks. Most approaches start from discrete models, such as Petri nets (Koch *et al*., 2005), process algebras (Regev *et al*., 2001), concurrent transition systems (Chabrier-Rivier *et al*., 2004), rewriting logic (Eker *et al*., 2002), and Boolean networks and their generalizations (Bernot *et al*., 2004). In this paper we show that model-checking techniques can also be used for more conventional continuous models, in particular differential equation models, when using qualitative abstractions to discretize the dynamics of the system. In comparison with ideas along the same line (Antoniotti *et al*., 2004; Ghosh *et al*., 2003; Shults and Kuipers, 1997), our approach is adapted to a particular class of PL differential equations with favorable mathematical properties, allowing the development of tailored algorithms that scale up well to models of large and complex genetic regulatory networks.

The model validation approach proposed in this paper has been applied to the analysis of the network controlling the *nutritional stress response* in *Escherichia coli*. In case of nutritional stress, an *E.coli* population abandons exponential growth and enters a non-growth state called stationary phase (Huisman *et al*., 1996). At the molecular level, this growth phase transition is controlled by a complex genetic regulatory network (Hengge-Aronis, 2000). We have constructed a model including key proteins and their interactions involved in the carbon starvation response, and validated this model by comparing the predicted temporal evolution of the protein concentrations with available experimental data, both during the transition from exponential to stationary phase, and during the reentry into exponential phase after a nutrient upshift. Although some of the predictions have thus been confirmed, one prediction has been refuted, suggesting model revisions. Another prediction concerns a surprising phenomenon that has not been experimentally investigated yet.

In the next section of the paper, we briefly outline the qualitative modeling and simulation method used to predict the behavior of genetic regulatory networks. Section 3 describes the model-checking approach towards model validation in some detail, as well as its computer implementation in GNA. The initial results of the validation of our model of the *E.coli* nutritional stress response are summarized in Section 4, followed by a discussion of the achievements in the final section.

2 QUALITATIVE SIMULATION

The method for the qualitative modeling and simulation of genetic regulatory networks that we use in this paper is a refinement of the method that we previously presented (de Jong *et al*., 2003, 2004). It is based on a qualitative abstraction that preserves stronger properties of the network dynamics, in particular the sign patterns of the derivatives of the concentration variables. This information is critical for the experimental validation of models of genetic regulatory networks, since experimental measurements of the system dynamics by means of quantitative RT–PCR, reporter genes and DNA microarrays usually result in observations of changes in the sign of the derivatives. We will provide an intuitive overview of the method, using a simple example. For technical details, the reader is referred to Batt *et al*. (2005).

Figure 1a shows a network consisting of two genes. When a gene (*a* or *b*) is expressed, the corresponding protein (A or B) is synthesized. Proteins A and B regulate the expression of genes *a* and *b*. More specifically, protein B inhibits the expression of gene *a* above a certain threshold concentration, whereas protein A inhibits the expression of gene *b* above a threshold concentration, and the expression of its own gene above a second, higher threshold concentration. The degradation of the proteins is not regulated.

The dynamics of genetic regulatory networks can be modeled by a class of *piecewise-linear* (PL) *differential equation* models originally introduced by Glass and Kauffman (1973). The example network gives rise to the following model:

$$\dot{x}_a = \kappa_a \, s^-(x_a, \theta_a^2) \, s^-(x_b, \theta_b) - \gamma_a \, x_a, \tag{1}$$

$$\dot{x}_b = \kappa_b \, s^-(x_a, \theta_a^1) - \gamma_b \, x_b, \tag{2}$$

where x_a and x_b denote the concentrations of proteins A and B, $\dot{x}_a$ and $\dot{x}_b$ their time derivatives, θ_a^1, θ_a^2 and θ_b threshold concentrations, κ_a and κ_b synthesis parameters, and γ_a and γ_b degradation parameters. The step function $s^-(x, \theta)$ evaluates to 1, if $x < \theta$, and to 0, if $x > \theta$. Step functions are approximations of the steep sigmoid functions often characterizing gene regulation, preserving their non-linear, switch-like character. As a consequence, PL models are coarse-grained models that abstract from the fine aspects of gene regulation, such as stochasticity, but have been shown adequate for a wide range of applications (see de Jong *et al*., 2004, for references).

Equations (1) and (2) describe the rate of change of the protein concentrations. Equation (2) states that protein B is produced (at a rate κ_b), if and only if $s^-(x_a, \theta_a^1) = 1$, that is, if and only if $x_a < \theta_a^1$. This captures the inhibition of the expression of gene *b* by protein A. Equation (1) states that protein A is produced (at a rate κ_a), if and only if neither $x_a > \theta_a^2$ nor $x_b > \theta_b$. Both proteins are degraded at a rate proportional to their own concentration.

Mathematical analysis of this model reveals that mere knowledge of the relative order of the threshold parameter(s) and

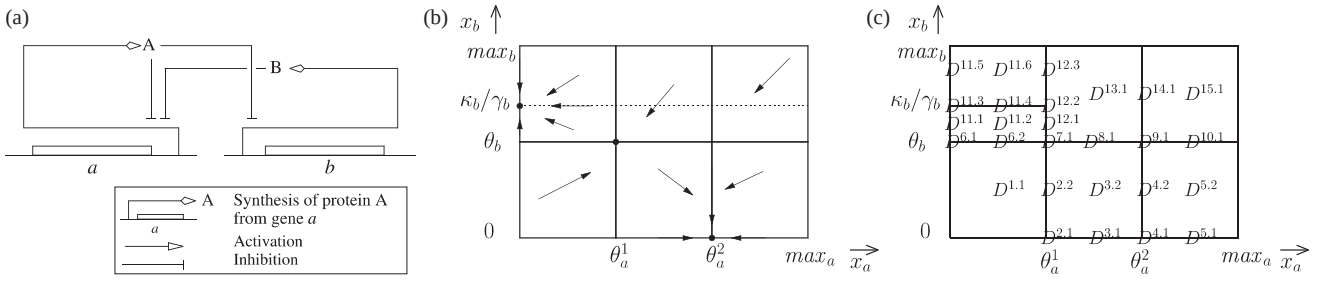


Fig. 1. (**a**) Simple genetic regulatory network consisting of two genes. (**b**) Sketch of the dynamics in the phase space of the two-gene network. The system has three equilibrium points, represented by dots. (**c**) Domain partition of the phase space.

the quotient of the synthesis and degradation parameter, for each of the two variables, is sufficient to sketch the flow in the phase space. This result has been shown to be generalizable to the whole class of PL models considered here. More particularly, assuming that

$$0 < \theta_a^1 < \theta_a^2 < \frac{\kappa_a}{\gamma_a} < max_a, \tag{3}$$

$$0 < \theta_b < \frac{\kappa_b}{\gamma_b} < max_b, \tag{4}$$

the phase space can be partitioned into hyperrectangular boxes, called *domains*, in which the flow is qualitatively identical, in the sense that either all solutions of the system traverse a domain instantaneously (*instantaneous* domain) or they have the same derivative sign pattern while remaining in the domain (*persistent* domain). Figures 1b and c represent the flow in the phase space and the domain partition of the phase space for the two-gene example. $D^{2.2}$ is an instantaneous domain, while $D^{1.1}$, $D^{4.2}$ and $D^{4.1}$ are persistent. Moreover, the latter domain coincides with an equilibrium point of the system. The domain partition is finer grained than the one used in our earlier work, for which the property that all solutions in a domain have the same derivative sign pattern does not generally hold.[1]

Using the domain partition of the phase space, together with the qualitative characterization of the dynamics in each of the domains, we can discretize the continuous dynamics. In the resulting abstract description, the state of the system is represented by a domain and its associated dynamical properties. There exists a transition from a domain D to another domain D', if and only if there exists a solution reaching D' from D, without leaving $D \cup D'$. This naturally leads to the introduction of a so-called *qualitative transition system*, consisting of the set of all domains, the set of all transitions between the domains and a labeling function that associates to every domain the sign of the derivatives of the concentration variables and an indication of whether the domain is persistent or instantaneous. The graph representation of the qualitative transition system is called a *state transition graph* and the domains are also called *qualitative states* (or qualitative *equilibrium* states, if the domains consist in equilibrium points). Figure 2 shows the qualitative transition system of the two-gene model.

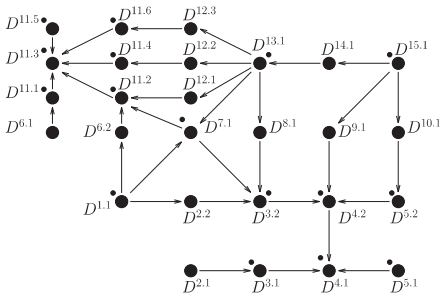
A sequence of qualitative states in the state transition graph is called a *path*. A path qualitatively describes a possible behavior of the system. In our two-gene example, ($D^{1.1}$, $D^{2.2}$, $D^{3.2}$, $D^{4.2}$, $D^{4.1}$) is a path leading to a qualitative equilibrium state (Fig. 2c). The qualitative transition system is defined such that it provides a *conservative approximation* of the dynamics of the original PL system, in the sense that to every solution of the model corresponds a path in the state transition graph. Note that the converse is not true: some paths may not correspond to any solution, and therefore represent spurious behaviors. The state transition graph has been shown to be invariant for all values of the parameters satisfying the parameter inequality constraints.

Simple rules have been formulated for the symbolic computation of the qualitative transition system from a PL model of the network. These rules exploit the favorable analytical properties of the class of PL models, thus allowing the qualitative states, the transitions between qualitative states, and the labeling function to be inferred from the parameter inequality constraints. The implementation of these rules has resulted in a new version of the computer tool GNA (de Jong *et al*., 2003). The new version of GNA, available at http://www-helix.inrialpes.fr/gna, has also been equipped with a strongly improved graphical user interface.

The paths in the state transition graph correspond to predicted qualitative behaviors of the system and can be compared with experimental data. The resulting model-validation problem is easy to solve for the simple two-gene example. For instance, the observation shown in Figure 3 is consistent with predictions, since there exists a path, ($D^{1.1}$, $D^{2.2}$, $D^{3.2}$, $D^{4.2}$, $D^{4.1}$), verifying the observed derivative sign pattern (Fig. 2c). However, the analysis of realistic models leads to large state transition graphs, which

[1] In this simple presentation of the method, we omit the problems raised by the discontinuities in the right-hand side of the PL differential equations, whose treatment goes beyond the scope of this article. See de Jong *et al*. (2004) and Gouzé and Sari (2002) for a detailed description.

(a)



(b)

	type	*value*		*dsign*	
		x_a	x_b	x_a	x_b
$D^{1.1}$	*pers*	$0 \le x_a < \theta_a^1$	$0 \le x_b < \theta_b$	1	1
$D^{2.2}$	*inst*	$x_a = \theta_a^1$	$0 \le x_b < \theta_b$	–	–
$D^{3.2}$	*pers*	$\theta_a^1 < x_a < \theta_a^2$	$0 \le x_b < \theta_b$	1	−1
$D^{4.2}$	*pers*	$x_a = \theta_a^2$	$0 \le x_b < \theta_b$	0	−1
$D^{4.1}$	*eq*	$x_a = \theta_a^2$	$x_b = 0$	0	0

(c)

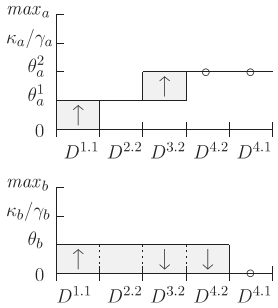


Fig. 2. Qualitative transition system of the two-gene model, with (**a**) the state transition graph and (**b**) the properties of some of the qualitative states in the graph. The following abbreviations have been used: *pers*, persistent state; *inst*, instantaneous state; *eq*, equilibrium state; *dsign*, derivative sign. The numbers −1, 0 and 1 denote the sign of the derivative of the protein concentrations. In instantaneous domains, the derivatives are not defined (Batt *et al*., 2005), indicated by a dash. The equilibrium states are $D^{4.1}$, $D^{7.1}$ and $D^{11.3}$, while dots next to states represent self-transitions. (**c**) Temporal evolution of the concentrations of proteins A and B in the path ($D^{1.1}$, $D^{2.2}$, $D^{3.2}$, $D^{4.2}$, $D^{4.1}$). Arrows indicate the sign of the derivatives for persistent states (up arrow for 1, down arrow for −1 and open circle for 0).

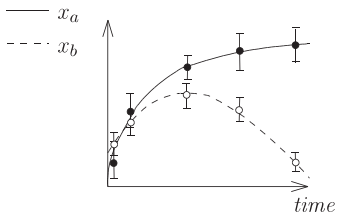


Fig. 3. Hypothetical experimental observation of the temporal evolution of the concentrations of proteins A and B.

make manual verification of dynamical properties error-prone or even practically infeasible. This has motivated the development of an automated, efficient method for model validation.

3 MODEL VALIDATION BY MODEL-CHECKING

Our model-validation approach combines the qualitative modeling and simulation method outlined above with techniques for *model checking* (Clarke *et al*., 1999). These techniques allow for the verification of properties of the behavior of discrete transition systems, expressed as formulas in some *temporal logic*. Using suitable model-checking algorithms and tools, it is possible to automatically and efficiently test whether the system satisfies the property. Model checking has been successfully applied to the verification of software, telecommunication systems, electronic circuits and other complex systems (for examples, see http://www.inrialpes.fr/vasy/cadp/case-studies/ and http://nusmv.irst.itc.it/).

Various model-checking frameworks exist, differing by their expressiveness, user-friendliness and computational efficiency. For the sake of simplicity, we focus here on one particular framework, in which the discrete transition system takes the form of a *Kripke structure*, and the behavioral properties are expressed in *Computation Tree Logic* (CTL) (Clarke *et al*., 1999). We describe the relation between qualitative simulation and model checking at the conceptual level, and briefly present an extension of GNA that connects the qualitative simulator with the model checker NuSMV. However, we emphasize that our approach is not restricted to CTL model-checking, and allows other more expressive temporal logics to be used as well (Section 3.3).

3.1 Translate qualitative transition system into Kripke structure

As a preliminary step, we introduce a set of *atomic propositions* to describe the state of the system. To be more precise, the set of atomic propositions we use consists of simple expressions describing the range of a protein concentration (e.g. $value\_x_a < \theta_a^1$), the sign of the derivative of a protein concentration (e.g. $dsign\_x_a = 1$) or the type of a state (e.g. $type = pers$). That is, in the example of Figure 2, the set of atomic propositions AP is given by

$$AP = \{value\_x_a = 0, value\_x_a > 0, value\_x_a < \theta_a^1, \ldots, dsign\_x_a = -1, dsign\_x_a = 0, dsign\_x_a = 1, \ldots, type = pers, type = inst, type = eq\}.$$

In general, a Kripke structure over a set of atomic propositions AP is a triple $\langle S, R, L \rangle$, where S is a set of states, $R \subseteq S \times S$ a total transition relation between the states, and $L{:}S \to 2^{AP}$ a labeling function that associates to each state, the set of atomic propositions true in that state (Clarke *et al*., 1999). The qualitative transition systems introduced in Section 2 are Kripke structures. As an illustration, the qualitative transition system of the two-gene network, graphically represented in Figure 2, can be alternatively represented as

the triple $\langle S, R, L\rangle$, where,

$$S = \{D^{1.1}, D^{2.1}, D^{2.2}, \ldots, D^{15.1}\},$$

$$R = \{(D^{1.1}, D^{2.2}), (D^{1.1}, D^{6.2}), \ldots, (D^{15.1}, D^{14.1})\},$$

$$L : \begin{cases} L(D^{1.1}) = \{value\_x_a \geq 0, value\_x_a < \theta_a^1, \ldots, \\ \qquad value\_x_b \geq 0, value\_x_b < \theta_b, \ldots, \\ \qquad dsign\_x_a = 1, dsign\_x_b = 1, \\ \qquad type = pers\}, \\ L(D^{2.1}) = \{value\_x_a = \theta_a^1, \ldots, type = inst\}, \\ \ldots \\ L(D^{15.1}) = \{value\_x_a > \theta_a^2, \ldots, type = pers\}. \end{cases}$$

3.2 Express dynamical properties in temporal logic

A CTL formula is built upon atomic propositions. The usual operators from propositional logic, such as negation ($\neg$), logical or ($\vee$), logical and ($\wedge$), and implication ($\rightarrow$), can also be used. In addition, CTL provides two types of operators: *path quantifiers*, **E** and **A**, and *temporal operators*, such as **F** and **G**. Path quantifiers are used to specify that a property p is satisfied by some ($\mathbf{E}p$) or every ($\mathbf{A}p$) path starting from a given state. Temporal operators are used to specify that, given a state and a path starting from that state, a property p holds for some ($\mathbf{F}p$) or for every ($\mathbf{G}p$) state of the path. Each path quantifier must be paired with a temporal operator.[2]

Informally speaking, path quantifiers are used to quantify over the possible behaviors of the system, since $\mathbf{A}p$ means that p must hold for every behavior, and $\mathbf{E}p$ means that p must hold for at least one behavior. Temporal operators are used to specify, given a behavior, temporal constraints on the state of the system, since $\mathbf{F}p$ and $\mathbf{G}p$ can be interpreted as meaning that for some future state and for every future state, respectively, p must hold.

How can the properties of interest for model validation be expressed as CTL formulas? This can be illustrated by means of the hypothetical experimental observation in Figure 3. The observation allows us to infer that the system reaches a state in which the concentrations of proteins A and B are both increasing, and from that state onwards, a second state in which the concentration of protein A is increasing and that of B decreasing. The property can be formalized by the CTL formula

$$\mathbf{EF}(dsign\_x_a = 1 \wedge dsign\_x_b = 1 \wedge \mathbf{EF}(dsign\_x_a = 1 \wedge dsign\_x_b = -1)). \quad (5)$$

The expression $\mathbf{EF}p$ means that there exists at least one path (**E**) leading to a future state (**F**) where p holds, thus expressing the *reachability* of that state. More generally, any time-series measurement of gene expression can be given as a combination of **EF** operators with conjunctions of atomic propositions describing the derivative sign patterns.

When understood in a broader sense, model validation does not just amount to the comparison of model predictions with time-series measurements of protein concentrations, but also involves the testing of other biologically meaningful properties (Bernot *et al*., 2004; Chabrier-Rivier *et al*., 2004). Suppose that we are interested in knowing whether every behavior of the system will eventually satisfy some property, for example, reach a specific state. We can investigate this by means of formulas using **AF** operators, which express the *inevitability* of a behavior. The following CTL formula expresses the conjecture that the two-gene network of Figure 1 will inevitably reach the equilibrium state $D^{11.3}$:

$$\mathbf{AF}(type = eq \wedge value\_x_a = 0). \quad (6)$$

As a second example, CTL can be used to express the sufficiency of certain conditions to cause the system to behave in a particular way. For example, one could ask, given that protein B is the only regulator of gene *a*, whether a high concentration of protein B guarantees the eventual disappearance of protein A. This *response* property can be expressed by the CTL formula

$$\mathbf{AG}(value\_x_b > \theta_b \rightarrow \mathbf{AF}value\_x_a = 0), \quad (7)$$

where $\mathbf{AG}p$ specifies that the property p must hold for every state.

3.3 Check if model satisfies dynamical properties

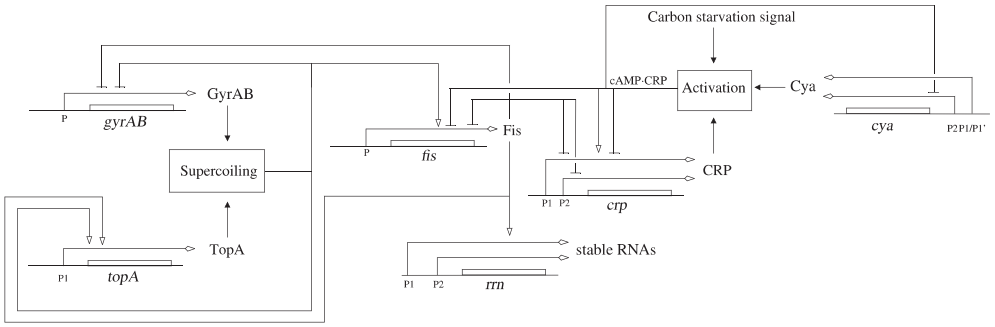
In order to test whether a discrete transition system satisfies a given temporal-logic formula, highly efficient algorithms have been developed and implemented in a range of model checkers. In addition to a yes/no answer, these tools return a diagnostic, either a witness or a counterexample, depending on whether the property holds or not. The diagnostic often provides valuable information for understanding why the property is satisfied or not.

In order to combine our qualitative simulator with model-checking tools, we have integrated export functionalities in the new version of GNA, allowing the user to generate text files describing the qualitative transition system in the format accepted by two widely used model checkers, NuSMV (Cimatti *et al*., 2002) and Evaluator, a component of the CADP toolbox (Mateescu and Sighireanu, 2003). NuSMV is an efficient, state-of-the-art model checker for CTL, whereas Evaluator is an on-the-fly model checker for the alternation-free μ-calculus, a temporal logic based on regular expressions. The text files generated by GNA can be imported in the model checkers, after which the verification of the properties of interest continues in the environment of the latter tools.

In this paper, we focus on the relation between GNA and NuSMV. Given a description of the Kripke structure, an initial state and a CTL formula, it is possible to check whether the

[2]For the formal syntax and semantics of CTL, see Clarke *et al*. (1999).

(a)



(b)

$$\dot{x}_{topA} = \kappa^1_{topA} + \kappa^2_{topA}\, s^+(x_{gyrAB}, \theta^3_{gyrAB})\, s^-(x_{topA}, \theta^1_{topA})\, s^+(x_{fis}, \theta^4_{fis}) - \gamma_{topA}\, x_{topA}$$

$$0 < \kappa^1_{topA}/\gamma_{topA} < \theta^1_{topA} < \theta^2_{topA} < \theta^3_{topA} < (\kappa^1_{topA} + \kappa^2_{topA})/\gamma_{topA} < max_{topA}$$

Fig. 4. (**a**) Network of key genes, proteins and regulatory interactions involved in the nutritional stress network in *E.coli*. The contents of the boxes labeled 'Activation' and 'Supercoiling' are detailed in Ropers *et al*. (2004). (**b**) PL differential equation and parameter inequality constraints for the topoisomerase TopA.

qualitative transition system in Figure 2 satisfies the property described by the formula. Provided that $D^{1.1}$ is the initial state, property (5) holds, and the path ($D^{1.1}$, $D^{2.2}$, $D^{3.2}$, $D^{4.2}$, $D^{4.1}$), shown in Figure 2c, is returned as a witness. Also, NuSMV shows that neither of the properties (6) and (7) hold.

Suppose that an experimentally-observed behavior does not correspond to any path in the state transition graph. Does this imply that the model must be rejected? Since the qualitative simulation method produces a conservative approximation of the dynamics of the original PL system (Section 2), one can be sure that a path corresponding to the experimentally-observed behavior must be present in the state transition graph, unless the model is invalid. As a consequence, the model can be safely rejected in the above case. On the other hand, if a path in the state transition graph corresponds to an experimentally-observed behavior, then the model is not necessarily corroborated by the observation, because the path may be a spurious behavior.

4 ANALYSIS OF NUTRITIONAL STRESS RESPONSE IN *E.COLI*

4.1 Model of nutritional stress response

In case of nutritional stress, an *E.coli* population abandons exponential growth and enters a non-growth state called *stationary phase*. This growth-phase transition is accompanied by numerous physiological changes in the bacteria, concerning among other things the morphology and the metabolism of the cells, as well as gene expression (Huisman *et al*., 1996). At the molecular level, the transition from exponential phase to stationary phase is controlled by a complex genetic regulatory network integrating various environmental signals.

Understanding the molecular basis of this essential developmental decision has been the focus of extensive studies for decades (Hengge-Aronis, 2000). However, notwithstanding the enormous amount of information accumulated on the genes, proteins and other molecules known to be involved in the stress adaptation process, there is currently no global understanding of how the response of the cell emerges from the network of molecular interactions. Moreover, with some exceptions, numerical values for the parameters characterizing the interactions and the molecular concentrations are absent from the literature, which makes it difficult to apply traditional methods for the dynamical modeling of genetic regulatory networks.

The above circumstances have motivated the qualitative analysis of the nutritional stress response network in *E.coli* by means of the method presented in this paper (Ropers *et al*., 2004). On the basis of literature data, we have decided to focus, as a first step, on a network of six genes that are believed to play a key role in the response of the cell to carbon starvation (Figure 4). The network includes genes involved in the transduction of the carbon starvation signal (the global regulator *crp* and the adenylate cyclase *cya*), metabolism (the global regulator *fis*), cellular growth (the *rrn* genes coding for stable RNAs) and DNA supercoiling, an important modulator of gene expression (the topoisomerase *topA* and the gyrase *gyrAB*).

Based on data in the experimental literature, a PL model of seven variables has been constructed, one protein concentration variable for each of the six genes and one input variable representing the presence or absence of the carbon starvation signal (Ropers *et al*., 2004). Seven differential equations, one for each variable, and forty inequality constraints describe the dynamics of the system. As an illustration, the differential equation and the parameter inequality constraints for the state variable x_{topA} are given in Figure 4b. For instance, the constraints $0 < \kappa^1_{topA}/\gamma_{topA} < \theta^1_{topA}$ express that without stimulation of the *topA* promoter, the TopA

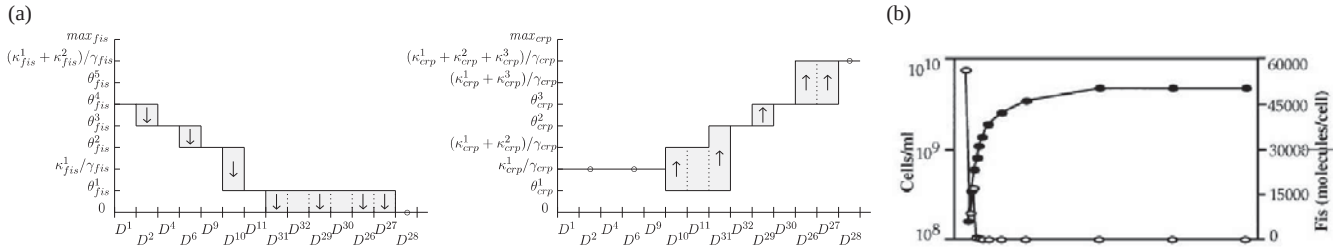


Fig. 5. Temporal evolution of the concentration of the proteins in the nutritional stress response network during the transition from exponential to stationary phase. (**a**) Predictions for Fis and CRP in a path in the state transition graph generated by qualitative simulation. (**b**) Observation for Fis (open circles) during the growth-phase transition, as indicated by cell density (closed circles) (Ali Azam *et al*., 1999).

concentration decreases towards a background level, below the threshold θ^1_{topA}.

Using the new version of the computer tool GNA, described in the previous sections, we have simulated two phenomena, namely the transition from exponential to stationary phase, and the reentry into exponential phase after a nutrient upshift. In order to validate the model, the simulation results have been compared with the available experimental data, using the export functionalities of GNA and the model checker NuSMV.

4.2 Validation of nutritional stress response model

In the absence of the carbon starvation signal, the system reaches a single qualitative equilibrium state that corresponds to the physiological conditions found in exponentially-growing *E.coli* cells. Starting from this equilibrium state, we perturb the system by switching on the carbon starvation signal and simulate the transition from exponential to stationary phase. This gives rise to a state transition graph of 66 states (27 of which are persistent), computed in less than one second on a PC (800 MHz, 256 MB). The graph contains a single equilibrium state corresponding to stationary-phase conditions. Figure 5 represents the temporal evolution of two of the protein concentrations in a path in the state transition graph. It shows that the concentration of Fis monotonically decreases to 0 and that of CRP monotonically increases to $(\kappa^1_{crp} + \kappa^2_{crp} + \kappa^3_{crp})/\gamma_{crp}$.

Are the predictions obtained from the model verified by the experimental data? Figure 5b shows the measured evolution of the Fis concentration (Ali Azam *et al*., 1999). Towards the end of the exponential phase, the concentration of Fis decreases and then becomes steady in stationary phase, which is characterized by a low concentration of stable RNAs x_{rrn}, that is, a concentration below the threshold θ_{rrn}. This observation can be translated into the following CTL formula:

$$\mathbf{EF}(dsign\_x_{fis} = -1 \wedge \mathbf{EF}(dsign\_x_{fis} = 0 \wedge value\_x_{rrn} < \theta_{rrn})). \quad (8)$$

The qualitative transition system has been exported to the model checker, in order to verify the property. Verification takes a fraction of a second to complete and shows that the observed temporal evolution of the Fis concentration is reproduced by the model, i.e. there exists a path in the state transition graph satisfying the property (8).

Figure 5b suggests that we could be even more precise in our temporal-logic formulation of the experimental data. Not only $dsign\_x_{fis} = 0$ in stationary phase, but in addition it would seem that $value\_x_{fis} = 0$. However, since the precision of the measurements is limited, there may remain some small amount of Fis in the cell in stationary phase. The description $value\_x_{fis} = 0$ is therefore too strong and might falsely rule out the model. Also, in this and similar examples, we use the temporal operator **F** instead of **G**, which would allow us to express that a property holds all of the time. The use of **G** is compromised by the fact that the usually low sampling frequency may cause us to miss phenomena predicted by simulation (e.g. a transient increase in a protein concentration) and thus, falsely rule out the model.

It would be interesting to put the predictions of the nutritional stress response model to more severe experimental tests. Unfortunately, time-series measurements of the evolution of the concentration of the other proteins in the network in Figure 4 during the transition from exponential to stationary phase are currently not available. However, even from the weak data that are available today, some interesting conclusions for model validation can be drawn. For instance, from the data in Balke and Gralla (1987) it can be inferred that the level of DNA supercoiling decreases during and after the transition to stationary phase. Since the level of DNA supercoiling is determined by the ratio of the concentration of GyrAB (which introduces supercoils into the DNA molecule) and the concentration of TopA (which removes supercoils from the DNA molecule) (Drlica, 1990), we require the following property to be satisfied by our model:

$$\mathbf{EF}((dsign\_x_{gyrAB} = -1 \vee dsign\_x_{topA} = 1) \wedge value\_x_{rrn} < \theta_{rrn}). \quad (9)$$

That is, during stationary phase, the concentration of GyrAB must decrease or the concentration of TopA must increase. Interestingly, the model does not satisfy the property (9),

as revealed by model checking: in all paths in the state transition graph, the TopA concentration remains constant, while the GyrAB concentration increases! The inconsistency between the model and the observed level of DNA supercoiling indicates a flaw in the model. It demonstrates that our picture of the nutritional stress response is incomplete, in the sense that the network of Figure 4 may need to be extended with interactions not yet identified or with regulators not yet considered. In Ropers *et al*. (2004) we propose experiments and model extensions to further investigate these possibilities.

In addition to simulating the transition from exponential to stationary phase, we have also studied the reentry into exponential phase after a nutrient upshift, i.e. when cells in stationary phase have been put into fresh medium. Using the same model as above, but starting the simulation from the qualitative state characterizing stationary-phase conditions and with the carbon starvation signal switched off, qualitative simulation results in a state transition graph of 1143 states (202 of which are persistent), generated in 1.7 s. The graph is more complex than that generated for the transition from exponential to stationary phase, in the sense that it contains several cyclic paths. From all states in the graph, one of these cyclic paths can be reached, which we have shown to be attractive. To be more precise, the qualitative transition system satisfies the property

$$\mathbf{AG}(\mathit{statesInCycle} \rightarrow \mathbf{AG}\mathit{statesInCycle}), \qquad (10)$$

where the predicate *statesInCycle* is satisfied by all and only states in the cyclic path. That is, if the system has reached this path, it always remains in the path (testing this property takes NuSMV 9.1 s). Further mathematical analysis has revealed that the cyclic path arises from solutions spiraling inwards to an equilibrium point (Ropers *et al*., 2004). In other words, during the reentry into stationary phase, the concentrations of some of the proteins oscillate towards a new equilibrium level. This is a surprising result, which has not been subject to investigation so far. We are currently carrying out experiments in our laboratory to measure the temporal evolution of the protein concentrations in the nutritional stress response network, directly after a nutrient upshift, in order to verify this prediction and continue the validation of our model.

5 DISCUSSION

We have presented an approach for the validation of models of genetic regulatory networks, which combines a method for qualitative modeling and simulation with techniques for model checking. The qualitative modeling and simulation method, exploiting favorable mathematical properties of a class of coarse-grained models of genetic regulations, is a refinement of our previous work (de Jong *et al*., 2003). The method yields predictions on the derivative sign patterns of the concentration variables that are particularly well adapted to the currently available experimental methods. The methodological novelty of this paper is that we use model-checking techniques to deal with the problem that the state transition graphs generated by qualitative simulation may become prohibitively large for biologically-interesting networks. They permit observed dynamical properties of the system to be reliably and efficiently verified. Moreover, due the fact that the state transition graphs are conservative approximations of the dynamics of the underlying PL models, the latter are guaranteed not to be ruled out falsely. The model-validation approach is supported by a new version of the computer tool GNA.

The applicability of our model-validation approach has been illustrated by the analysis of the complex regulatory network underlying the nutritional stress response of *E.coli*. We have constructed a model of a part of this network, consisting of key proteins and their interactions involved in the carbon starvation response, and validated this model by the available experimental data in the literature. Although most predictions on the entry into stationary phase are consistent with the observations, in one case they contradict the experimental data, i.e. the observed decrease of the DNA supercoiling level, and necessitate revisions of the model. In addition, we have used model checking to further analyze the surprising prediction of the model that some of the protein concentrations oscillate after a nutrient upshift. This involves verifications that would be difficult to achieve by visual inspection.

Several applications of model checking and other formal verification techniques for the analysis and validation of biochemical network models have been proposed recently. Most approaches apply to discrete models, such as Petri nets (Koch *et al*., 2005), process algebras (Regev *et al*., 2001), concurrent transition systems (Chabrier-Rivier *et al*., 2004), rewriting logic (Eker *et al*., 2002) and Boolean networks and their generalizations (Bernot *et al*., 2004). For instance, in Bernot *et al*. (2004), a logical modeling approach is used in combination with CTL model checking to analyze models of mucus production in *Pseudomonas aeruginosa*, while the validation of a Petri net model of the sucrose breakdown pathway is investigated in Koch *et al*. (2005). The work presented in this paper shows that model checking can also be used for more conventional continuous models, like differential equation models. However, this requires a preliminary discretization of the dynamics of the system using abstractions. Several other approaches taking this direction can be mentioned (Antoniotti *et al*., 2004; Ghosh *et al*., 2003; Shults and Kuipers, 1997), based on qualitative differential equations (Shults and Kuipers, 1997) or hybrid automata (Antoniotti *et al*., 2004; Ghosh *et al*., 2003). However, contrary to our approach, these methods either do not result in a conservative approximation of the dynamics of the underlying continuous models (Antoniotti *et al*., 2004) or they are based on general purpose analysis techniques (Ghosh *et al*., 2003; Shults and Kuipers, 1997). The conservative approximation

that we obtain is critical for preventing that models are unnecessarily rejected. The particular mathematical form of the PL models allows simple, tailor-made algorithms to be used, which promote the upscalability of our approach to large and complex networks, but at the same time limits its generality.

The model-validation approach of this paper has been illustrated in the context of CTL model checking. While CTL allows a variety of biologically meaningful properties to be expressed, some properties fall outside its scope. For instance, in Section 4.2 we would have liked to express the occurrence of oscillations in some of the protein concentrations after a nutrient upshift. That is, we would have liked to state that there exists a path in the qualitative transition system, such that from a state satisfying p it is always possible to reach a state satisfying $\neg p$, and from a state satisfying $\neg p$, it is always possible to reach a state satisfying p, where p might express that the concentration of some protein is above a threshold and $\neg p$ that it is below this threshold. The formula $\mathbf{EG}(p \rightarrow \mathbf{F}\neg p \wedge \neg p \rightarrow \mathbf{F}p)$ expresses this property, but unfortunately it is not a CTL formula (because **F** is not paired with a path quantifier) and it does not admit any CTL equivalent (Clarke and Draghicescu, 1988). However, the above property can be expressed in the μ-calculus and evaluated using XTL, a component of the CADP toolbox (Mateescu and Garavel, 1998). The capability of GNA to generate export files for different model checkers, allows one to take advantage from the specific strengths of each of them.

A problem encountered in the validation of our model is that time-series measurements of the concentrations of the proteins in the model are currently rare and usually have a low sampling frequency. In addition, the measurements for different proteins are difficult to combine, because they have been carried out under different conditions (using different strains, different culture media, etc.). This has the practical consequence that many interesting predictions obtained through qualitative simulation cannot currently be tested. In order to validate the model more rigorously, we are currently working on fine-grained measurements of gene expression in wild-type and mutant strains during growth-phase transitions. More generally, as systems biology takes hold, we expect such model-driven experiments to become more prominent.

REFERENCES

Ali Azam,T., Iwata,A., Nishimura,A., Ueda,S. and Ishihama,A. (1999) Growth phase-dependent variation in protein composition of the *E.coli* nucleoid. *J. Bacteriol.*, **181**, 6361–6370.

Antoniotti,M., Piazza,C., Policriti,A., Simeoni,M. and Mishra,B. (2004) Taming the complexity of biochemical models through bisimulation and collapsing: theory and practice. *Theor. Comput. Sci.*, **325**, 45–67.

Balke,V.L. and Gralla,J.D. (1987) Changes in the linking number of supercoiled DNA accompany growth transitions in *Escherichia coli*. *J. Bacteriol.*, **169**, 4499–4506.

Batt,G. Ropers,D., de Jong,H., Geiselmann,J., Page,M. and Schneider,D. (2005) Qualitative analysis and verification of hybrid models of genetic regulatory networks. In Morari,M. and Thiele,L. (eds), *HSCC' 05*, Lecture Notes in Computer Science Vol. 3414, Springer, Berlin, pp. 134–150.

Bernot,G., Comet,J.-P., Richard,A. and Guespin,J. (2004) A fruitful application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic. *J. Theor. Biol.*, **229**, 339–347.

Chabrier-Rivier,N., Chiaverini,M., Danos,V., Fages,F. and Schächter,V. (2004) Modeling and querying biomolecular interaction networks. *Theor. Comput. Sci.*, **325**, 25–44.

Cimatti,A., Clarke,E.M., Giunchiglia,E., Giunchiglia,F., Pistore,M., Roveri,M., Sebastiani,R. and Tacchella,A. (2002) NuSMV2: an opensource tool for symbolic model checking. In Brinksma,E. and Larsen,K.G. (eds), *CAV'02*, Lecture Notes in Computer Science, Vol. 2404, Springer, Berlin, pp. 359–364.

Clarke,E.M. and Draghicescu,I.A. (1988) Expressibility results for linear-time and branching-time logics. In de Bakker,J.W., de Roever,W.P. and Rozenberg,G. (eds), *REX Workshop*, Lecture Notes in Computer Science Vol. 354, Springer, Berlin, pp. 428–437.

Clarke,E.M., Grumberg,O. and Peled,D.A. (1999) *Model Checking*. MIT Press, Cambridge, MA.

de Jong,H. (2002) Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.*, **9**, 69–105.

de Jong,H., Geiselmann,J., Hernandez,C. and Page,M. (2003) Genetic Network Analyzer: qualitative simulation of genetic regulatory networks. *Bioinformatics*, **19**, 336–344.

de Jong,H., Gouzé,J.-L., Hernandez,C., Page,M., Sari,T. and Geiselmann,J. (2004) Qualitative simulation of genetic regulatory networks using Piecewise-linear models. *Bull. Math. Biol.*, **66**, 301–340.

Drlica,K. (1990) Bacterial topoisomerases and the control of DNA supercoiling. *Trends Genet.*, **6**, 433–437.

Eker,S., Knapp,M., Laderoute,K., Lincoln,P., Meseguer,J. and Sönmez,M.K. (2002) Pathway logic: symbolic analysis of biological signaling. In Altman,R.B., Dunker,A.K., Hunter,L., Jung,T., and Klein,T.C. (eds), *PSB'02*, World Scientific Publishing, Singapore, pp. 400–412.

Ghosh,R., Tiwari,A. and Tomlin,C.J. (2003) Automated symbolic reachability analysis, with application to Delta-Notch signaling automata. In Maler,O. and Pnueli,A. (eds), *HSCC' 03*, Lecture Notes in Computer Science Vol. 2623, Springer, Berlin, pp. 233–248.

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

Gouzé,J.-L. and Sari,T. (2002) A class of piecewise-linear differential equations arising in biological models. *Dyn. Syst.*, **17**, 299–316.

Hengge-Aronis,R. (2000) The general stress response in *E.coli*. In Storz,G. and Hengge-Aronis,R. (eds), *Bacterial Stress Responses*. ASM Press, Washington, DC, pp. 161–177.

Huisman,G.W., Siegele,D.A., Zambrano,M.M. and Kolter,R. (1996) Morphological and physiological changes during stationary phase. In Neidhardt,F.C., Curtiss III,R., Ingraham,J.L., Lin,E.C.C., Low,K.B., Magasanik,B., Reznikoff,W.S., Riley,M.,

Schaechter,M. and Umbarger,H.E. (eds), *Escherichia coli and Salmonella: Cellular and Molecular Biology*. ASM Press, Washington, DC, pp. 1672–1682.

Koch,I., Junker,B.H. and Heiner,M. (2005) Application of Petri net theory for modelling and validation of the sucrose breakdown pathway in the potato tuber. *Bioinformatics* **21**, 1219–1226.

Mateescu,R. and Sighireanu,M. (2003) Efficient on-the-fly model-checking for regular alternation-free mu-calculus. *Sci. Comput. Program.*, **46**, 255–281.

Mateescu,R. and Garavel,H. (1998) XTL: a meta-language and tool for temporal logic model-checking. In Margaria,T. and Steffen,B. (eds) STTT'98. Brics, Aalborg, pp. 33–42.

Regev,A., Silverman,W. and Shapiro,E. (2001) Representation and simulation of biochemical processes using the pi-calculus process algebra. In Altman,R.B., Dunker,A.K., Hunter,L. and Klein,T.E. (eds), *PSB'01*. World Scientific Publishing, Singapore, pp. 459–470.

Ropers,D., de Jong,H., Page,M., Schneider,D. and Geiselmann,J. (2004) Qualitative simulation of nutritional stress response in *Escherichia coli. Technical Report INRIA RR-5412*.

Shults,B. and Kuipers,B.J. (1997) Proving properties of continuous systems: qualitative simulation and temporal logic. *Artif. Intell.*, **92**, 91–130.

BIOINFORMATICS

Vol. 26 ECCB 2010, pages i603–i610
doi:10.1093/bioinformatics/btq387

Efficient parameter search for qualitative models of regulatory networks using symbolic model checking

Gregory Batt[1,*], Michel Page[2,3], Irene Cantone[4], Gregor Goessler[2], Pedro Monteiro[2,5] and Hidde de Jong[2]

[1]INRIA Paris - Rocquencourt, Le Chesnay, [2]INRIA Grenoble - Rhône-Alpes, Montbonnot, [3]IAE, Université Pierre Mendès France, Grenoble, France, [4]Clinical Sciences Center, Imperial College, London, UK and [5]INESC/Instituto Superior Técnico, Lisbon, Portugal

ABSTRACT

Motivation: Investigating the relation between the structure and behavior of complex biological networks often involves posing the question if the hypothesized structure of a regulatory network is consistent with the observed behavior, or if a proposed structure can generate a desired behavior.

Results: The above questions can be cast into a parameter search problem for qualitative models of regulatory networks. We develop a method based on symbolic model checking that avoids enumerating all possible parametrizations, and show that this method performs well on real biological problems, using the IRMA synthetic network and benchmark datasets. We test the consistency between IRMA and time-series expression profiles, and search for parameter modifications that would make the external control of the system behavior more robust.

Availability: GNA and the IRMA model are available at http://ibis.inrialpes.fr/

Contact: gregory.batt@inria.fr

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

A central problem in the analysis of biological regulatory networks concerns the relation between their structure and dynamics. This problem can be narrowed down to the following two questions: (a) Is a hypothesized structure of the network consistent with the observed behavior? (b) Can a proposed structure generate a desired behavior?

Qualitative models of regulatory networks, such as (synchronous or asynchronous) Boolean models and piecewise-affine differential equation (PADE) models, have been proven useful for addressing the above questions. The models are coarse-grained, in the sense that they do not explicitly specify the biochemical mechanisms. However, they include the logic of gene regulation and allow different expression levels of the genes to be distinguished. They are interesting in their own right, as a way to capture in a simple manner the complex dynamics of a large regulatory network (Chaves *et al*., 2009; Fauré *et al*., 2006; Monteiro *et al*., 2008; Saez-Rodriguez *et al*., 2009). They can also be used as a first step to orient the development of more detailed quantitative ODE models.

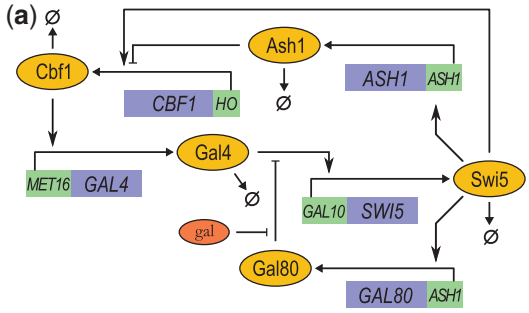
Qualitative models bring specific advantages when studying the relation between structure and dynamics. In order to answer questions (a) and (b), one has to search the parameter space to check if for some parameter values the network is consistent with the data or can attain a desired control objective. In qualitative models, the number of different parametrizations is finite and the number of possible values for each parameter is usually rather low. This makes parameter search easier to handle than in quantitative models, where exhaustive search of the continuous parameter space is in general not feasible. Moreover, qualitative models are concerned with trends rather than with precise quantitative values, which corresponds to the nature of much of the available biological data (Cantone *et al*., 2009).

Nevertheless, the parametrization of qualitative models remains a complex problem. For most models of networks of biological interest the state and parameter spaces are too large to exhaustively test all combinations of parameter values. The aim of this article is to address this search problem for PADE models by treating it in the context of formal verification and symbolic model checking (Clarke *et al*., 1999; Fisher and Henzinger, 2007).

Our contributions are twofold. On the methodological side, we develop a method that in comparison with our previous work (Batt *et al*., 2005) makes it possible to efficiently analyze large and possibly incompletely parametrized PADE models. This is achieved by a *symbolic encoding* of the model structure, constraints on parameter values and transition rules describing the qualitative dynamics of the system. We can thus take full advantage of symbolic model checkers for testing the consistency of the network structure with dynamic properties expressed in temporal logics. The computer tool GNA has been extended to export the symbolic encoding of PADE models in the NuSMV language (Cimatti *et al*., 2002). In comparison with related work (Barnat *et al*., 2009; Bernot *et al*., 2004; Corblin *et al*., 2009; Fromentin *et al*., 2007), our method applies to incompletely instead of fully parametrized models, provides more precise results and the encoding is efficient without (strongly) simplifying the PADE dynamics.

On the application side, we show that the *method performs well on real problems*, by means of the IRMA synthetic network and benchmark experimental datasets (Cantone *et al*., 2009). More precisely, we are able to find parameter values for which the network satisfies temporal-logic properties describing observed expression profiles, both on the level of individual and averaged time series. The method is selective in the sense that only a small part of the parameter space is found to be compatible with the observations. Analysis of these parameter values reveals that biologically relevant constraints have been identified. Moreover, we make suggestions to improve the robustness of the external control of the IRMA behavior by proposing a rewiring of the network.

*To whom correspondence should be addressed.



(b)

$$\dot{x}_{Gal4} = \kappa^0_{Gal4} + \kappa_{Gal4}\, s^+(x_{Cbf1}, \theta_{Cbf1}) - \gamma_{Gal4}\, x_{Gal4}$$
$$\dot{x}_{Swi5} = \kappa^0_{Swi5} + \kappa_{Swi5}\, s^+(x_{Gal4}, \theta_{Gal4})\,(1 - s^+(x_{Gal80}, \theta_{Gal80}) s^-(u_{gal}, \theta_{gal})) - \gamma_{Swi5}\, x_{Swi5}$$
$$\dot{x}_{Ash1} = \kappa^0_{Ash1} + \kappa_{Ash1}\, s^+(x_{Swi5}, \theta^a_{Swi5}) - \gamma_{Ash1}\, x_{Ash1}$$
$$\dot{x}_{Cbf1} = \kappa^1_{Cbf1}\, s^+(x_{Swi5}, \theta^c_{Swi5}) + \kappa^2_{Cbf1}\, s^+(x_{Swi5}, \theta^c_{Swi5}) s^-(x_{Ash1}, \theta_{Ash1}) - \gamma_{Cbf1}\, x_{Cbf1}$$
$$\dot{x}_{Gal80} = \kappa^0_{Gal80} + \kappa_{Gal80}\, s^+(x_{Swi5}, \theta^g_{Swi5}) - \gamma_{Gal80}\, x_{Gal80}$$

Fig. 1. Synthetic IRMA network in yeast. (**a**) Schematic representation of the network constructed in Cantone *et al*. (2009). The green and blue boxes are promoter and genes, and the yellow and red ovals are proteins and metabolites. (**b**) PADE model of IRMA, with state variables x, protein synthesis constants κ, decay constants γ and thresholds θ. The input variable u_{gal} refers to the presence of galactose ($\dot{u}_{gal} = 0$). The subscripts $_{Gal4}$, $_{Swi5}$, $_{Ash1}$, $_{Cbf1}$, $_{Gal80}$ refer to the proteins.

2 QUALITATIVE MODEL OF IRMA NETWORK

2.1 IRMA network

IRMA is a synthetic network constructed in yeast and proposed as a benchmark for modeling and identification approaches (Cantone *et al*., 2009). The network consists of five well-characterized genes that have been chosen such that different kinds of interactions are included, notably transcription regulation and protein–protein interactions. The endogenous copies of the genes were deleted to reduce crosstalk of IRMA with the regulatory networks of the host cell. In order to further isolate the synthetic network from its cellular environment, the genes belong to distinct, non-redundant pathways.

The structure of the IRMA network is shown in Figure 1a. The expression of the *CBF1* gene is under the control of the *HO* promoter, which is positively regulated by Swi5 and negatively regulated by Ash1. *CBF1* encodes the transcription factor Cbf1 that activates expression of the *GAL4* gene. The *GAL10* promoter is activated by Gal4, but only in the absence of Gal80 or in the presence of galactose. Gal80 binds to the Gal4 activation domain, but galactose releases this inhibition of transcription. The *GAL10* promoter controls the expression of *SWI5*, whose product not only activates the above-mentioned *HO* promoter, but also the *ASH1* promoter controlling the expression of the *GAL80* and *ASH1* genes.

The network contains one positive (Swi5/Cbf1/Gal4/Swi5) and two negative (Swi5/Gal80/Swi5; Swi5/Ash1/Cbf1/Gal4/Swi5) feedback loops. Negative feedback loops are a necessary condition for the occurrence of oscillations (Thomas and d'Ari, 1990), while the addition of positive feedback is believed to increase the robustness of the oscillations (Tsai *et al*., 2008). Consequently, for suitable parameter values IRMA might function as a synthetic oscillator.

2.2 Measurements of IRMA dynamics

The behavior of the network has been monitored in response to two different perturbations (Cantone *et al*., 2009): shifting cells from glucose to galactose medium (switch-on experiments), and from galactose to glucose medium (switch-off experiments). The terms 'switch-on' ('switch-off') refer to the activation (inhibition) of *SWI5* expression during growth on galactose (glucose). For these two perturbations, the temporal evolution of the expression of *all* the genes in the network was monitored by qRT-PCR with good time resolution.

Figure 2a represents the expression of all genes, averaged over five (switch-on) or four (switch-off) independent experiments. In the switch-off experiments (galactose to glucose), the transcription of all genes is shut off. In the switch-on experiments, a seemingly oscillatory behavior is present with Swi5 peaks at 40 and 180 min, and Swi5, Cbf1 and Ash1 expressed at moderate to high levels (Cantone *et al*., 2009).

The analysis of the individual time series reveals that in some cases the gene expression profiles are indeed similar, at least qualitatively, whereas in other cases notable differences exist (e.g. the oscillatory behavior is not present in all switch-on time series, see Fig. 2c). In the latter case, averaged expression levels may be a misleading representation of the network behavior.

2.3 PADE model of IRMA network

We built a qualitative model of the IRMA dynamics using PADE models of genetic regulatory networks. PADE models, originally introduced in Glass and Kauffman (1973), provide a coarse-grained picture of the network dynamics. They have the following general form:

$$\dot{x}_i = f_i(x) \triangleq \sum_{l \in L_i} \kappa^l_i\, b^l_i(x) - \gamma_i x_i,\ i \in [1,n] \qquad (1)$$

where $x \in \Omega \subset \mathbb{R}^n_{\geq 0}$ represents a vector of n protein (or RNA) concentrations. The synthesis rate is composed of a sum of synthesis constants κ^l_i, each modulated by a regulation function $b^l_i(x) \in \{0,1\}$, with l in an index set L_i. A regulation function is an algebraic expression of step functions $s^+(x_j,\theta_j)$ or $s^-(x_j,\theta_j)$ which formalizes the regulatory logic of gene expression. θ_j is a so-called threshold for the concentration x_j. The step function $s^+(x_j,\theta_j)$ evaluates to 1 if $x_j > \theta_j$, and to 0 if $x_j < \theta_j$, thus capturing the switch-like character of gene regulation ($s^-(x_j,\theta_j) = 1 - s^+(x_j,\theta_j)$). The degradation of a gene product is a first-order term, with a degradation constant γ_i.

In the case of IRMA, we define five variables, each corresponding to the total concentration of a protein, and an input variable denoting the concentration of galactose. Notice that the measurements of the network dynamics concern mRNA and not protein levels. We assume that the variations in mRNA and protein levels are the same, even though this may not always be the case. A similar approximation is made in Cantone *et al*. (2009), where protein and mRNA levels are assumed to be proportional.

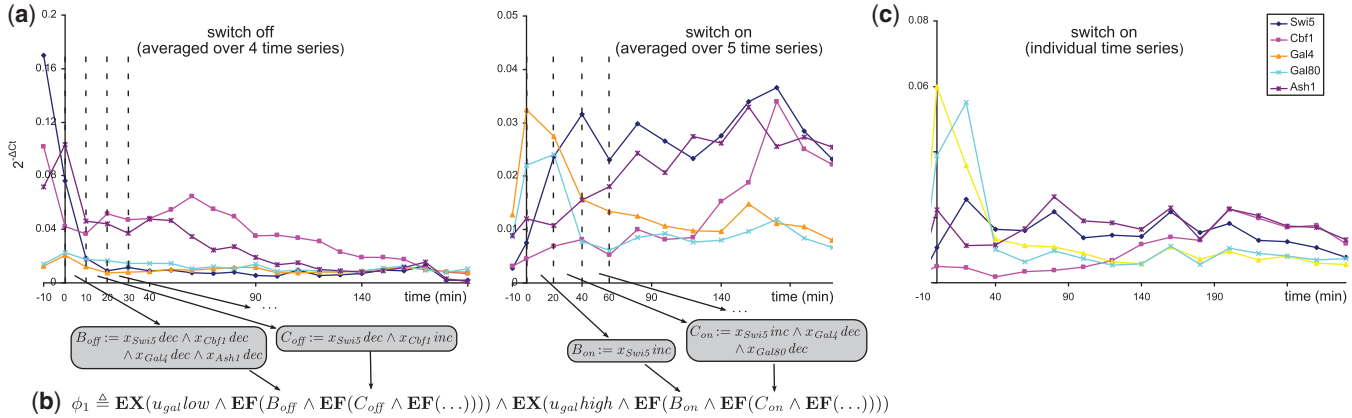


Fig. 2. Dynamic behavior of the IRMA network in response to medium shift perturbations. (**a**) Temporal profiles of averaged gene expression measured with qRT-PCR during switch-off (left) and switch-on (right) experiments (data from Cantone *et al*., 2009). (**b**) Temporal logic encoding of the switch-off and switch-on behaviors. The operator $\mathbf{EF}\phi$ expresses the possibility to reach a future state satisfying ϕ, whereas the operator $\mathbf{EX}\phi$ is used to require the existence of an initial state satisfying ϕ. $u_{gal}\,low$ and $u_{gal}\,high$ denote the absence and presence of galactose, respectively. See Clarke *et al*. (1999) for more details on the temporal logic CTL. Only changes greater than 5×10^{-3} units are considered significant. (**c**) Temporal gene expression profile in an individual switch-on experiment showing a switch-off-like behavior.

The PADE model of the IRMA network is shown in Figure 1b. Consider the equation for the protein Gal4. κ^0_{Gal4} is its basal synthesis rate, and $\kappa^0_{Gal4}+\kappa_{Gal4}$ its maximal synthesis rate when the *GAL4* activator Cbf1 is present (i.e. $x_{Cbf1} > \theta_{Cbf1}$). Swi5 is regulated in a more complex way. The expression of its gene is activated by Gal4, but only when not both Gal80 is present and galactose absent (which would lead to Gal4 inactivation by Gal80). The step-function expression in Figure 1b mathematically describes this condition. The IRMA PADE model is described in more detail in Section 1 of the Supplementary Material.

The model resembles the ODE model in Cantone *et al*. (2009), but notably approximates the Hill-type kinetic rate laws by step functions. It thus makes the implicit assumption that important qualitative dynamical properties of the network are intimately connected with the network structure and the regulatory logic, independently from the details of the kinetic mechanisms and precise parameter values. Several studies have shown this assumption to be valid in a number of model systems (Chaves *et al*., 2009; Davidich and Bornholdt, 2008), although care should be exercised in deciding exactly when modeling approximations are valid (Polynikis *et al*., 2009).

To investigate for the possible existence of unknown interactions between the synthetic network and the host, we would like to test by means of the PADE model if the network structure and the regulatory logic alone can fully account for the trends in the gene expression profiles observed in Cantone *et al*. (2009). Because the addition of galactose does not always lead to an effective activation of the IRMA genes, we also search for parameter modifications that would render the network response to galactose more robust.

3 SEARCH OF PARAMETER SPACE USING SYMBOLIC MODEL CHECKING

3.1 Qualitative analysis of PADE models

The advantage of PADE models is that the qualitative dynamics of high-dimensional systems are relatively easy to analyze, using only the total *order* on parameter values rather than exact numerical values (Batt *et al*., 2008; Edwards and Glass, 2006). The main difficulty lies in treating the discontinuities in the right-hand side of the differential equations, at the threshold values of the step functions. Following Gouzé and Sari (2002), the use of differential inclusions based on Filippov solutions has been proposed in Batt *et al*. (2008) and implemented in the computer tool GNA (Batt *et al*., 2005). Here, we recast this analysis in a form that underlies the symbolic encoding of the dynamics below.

The key to our reformulation of the qualitative analysis of the PADE dynamics is the extension of step functions s^+ to interval-valued functions S^+, where

$$S^+(x_j,\theta_j)=\begin{cases}[0,0] \text{ if } x_j<\theta_j\\ [0,1] \text{ if } x_j=\theta_j\\ [1,1] \text{ if } x_j>\theta_j\end{cases} \tag{2}$$

Because the step functions are not defined at their thresholds, we conservatively assume that they can take any value between 0 and 1 [see Chaves *et al*. (2009) for a similar idea]. When replacing the step functions by their extensions, the regulation functions $b^l_i(x)$ become *interval-valued* functions $B^l_i:\mathbb{R}^n_{\geq 0}\to\{[0,0],[0,1],[1,1]\}$, and Equation (1) generalizes to the following differential inclusion using interval arithmetic (Moore, 1979):

$$\dot{x}_i \in F_i(x) \triangleq \sum_{l\in L_i}\kappa^l_i B^l_i(x)-\gamma_i x_i,\ i\in[1,n] \tag{3}$$

The solutions of (3) are for practical purposes the same as the solutions of the differential inclusions defined in Batt *et al*. (2008) (see Section 2 of the Supplementary Material).

The starting point for our qualitative analysis is the introduction of a rectangular partition $\mathcal{D}$ of the state space Ω. This partition is a rectangular grid defined by the threshold parameters $\Theta_i=\{\theta^j_i \mid j\in J_i\}$, where J_i is an index set, and the so-called focal parameters $\Lambda_i=\{\sum_{l\in B}\kappa^l_i/\gamma_i \mid B\subseteq L_i\}$, $i\in[1,n]$. Focal parameters are steady-state concentrations towards which the system locally converges in a monotonic way (Glass and Kauffman, 1973). For Gal4,

we have $\Theta_{Gal4}=\{\theta_{Gal4}\}$ and $\Lambda_{Gal4}=\{0,\kappa^0_{Gal4}/\gamma_{Gal4},(\kappa^0_{Gal4}+\kappa_{Gal4})/\gamma_{Gal4}\}$.

Interestingly, the partition has the property that in each domain $D\in\mathcal{D}$, the protein production rates are identical: for all $x,y\in D$, it holds that $B^l_i(x)=B^l_i(y)\triangleq B^l_i(D)$. As a consequence, the derivatives of the concentration variables have a *unique sign pattern*: for all $x,y\in D$, it holds that $\mathrm{sign}(F_i(x))=\mathrm{sign}(F_i(y))\subseteq\{-1,0,1\}$, where $\mathrm{sign}(A)\triangleq\{\mathrm{sign}(a)\mid a\in A\}$ denotes the signs of the elements in A (Batt *et al*., 2008). Notice that this property is not obtained for less fine-grained partitions used in related work (Barnat *et al*., 2009; Bernot *et al*., 2004; Chaves *et al*., 2009; Corblin *et al*., 2009; Fauré *et al*., 2006; Fromentin *et al*., 2007). It will be found critical for the search of parametrized models of IRMA that satisfy the time-series data.

The above considerations motivate a discrete abstraction, resulting in a *state transition graph*. In this graph, the states are the domains $D\in\mathcal{D}$, and there is a transition from a domain D to another domain D', if there exists a solution of the differential inclusion (3) that starts in D and reaches D', without leaving $D\cup D'$. The state transition graph defines the qualitative dynamics of the system, in the sense that paths in this graph describe how the qualitative state of the system evolves over time (Batt *et al*., 2008).

In Batt *et al*. (2008), three different types of transitions are defined: *internal*, from a domain D to itself; *dimension-increasing*, from a domain D to another, higher dimensional domain D' ($D\subseteq\partial D'$); and *dimension-decreasing*, from a domain D to a lower dimensional domain D' ($D'\subseteq\partial D$), where ∂D denotes the boundary of D in its supporting hyperplane. We reformulate here the transition rules using the interval extensions of the regulation functions. We introduce an interval-valued function F_i: $\mathcal{D}\times\mathcal{D}\to 2^{\mathbb{R}}$, where $F_i(D,D')=\sum_{l\in L_i}\kappa^l_i B^l_i(D)-\gamma_i D'_i$, for $D,D'\in\mathcal{D}$. $F_i(D,D')$ represents the flow in D infinitely close to D'. In order to evaluate $F_i(D,D')$, we use interval arithmetic (Moore, 1979). For instance, in a domain in which $x_{Swi5}>\theta^c_{Swi5}$ and $x_{Ash1}=\theta_{Ash1}$, we have $S^+(x_{Swi5},\theta^c_{Swi5})=[1,1]$ and $S^-(x_{Ash1},\theta_{Ash1})=[0,1]$, so that the differential inclusion for x_{Cbf1} becomes $[\kappa^1_{Cbf1}-\gamma_{Cbf1}x_{Cbf1},\kappa^1_{Cbf1}+\kappa^2_{Cbf1}-\gamma_{Cbf1}x_{Cbf1}]$. We obtain the following transition rule:

PROPOSITION 1 (Dimension-increasing transition). *Let $D,D'\in\mathcal{D}$ and $D\subseteq\partial D'$, that is, D lies in the boundary of D'. $D\to D'$ is a dimension-increasing transition iff*

(1) *$\forall i\in[1,n]$, such that D_i and D'_i coincide with a value in $\Theta_i\cup\Lambda_i$, it holds that $0\in F_i(D',D)$, and*

(2) *$\forall i\in[1,n]$, such that $D_i\neq D'_i$, it holds that $\exists\alpha>0$ such that $\alpha\in F_i(D',D)(D'_i-D_i)$*

Condition 1 guarantees that solutions can remain in domains located in threshold and focal planes, while Condition 2 expresses that the direction of the flow in the domains $(F_i(D',D))$ agrees with their relative position (D'_i-D_i). The proof of the rule and the rules for other types of transition can be found in Section 3 of the Supplementary Material.

It can be shown that exact parameter values are *not* needed for the analysis of the qualitative dynamics of a PADE model: it is sufficient to know the *ordering of the threshold and focal parameters* (Batt *et al*., 2008). This comes from the fact that the sign of F_i, and hence the transitions and the state transition graph, are invariant for regions of the parameter space defined by a total order on $\Theta_i\cup\Lambda_i$. We call each such total order a *parametrization* of the PADE model.

3.2 A model-checking approach for parameter search

For large graphs like that obtained for IRMA (which has about 50 000 states), verifying the compatibility of the network structure with an observed or desired behavioral property is impossible to do by hand. This has motivated the use of model-checking tools (e.g. Barnat *et al*., 2009; Batt *et al*., 2005; Bernot *et al*., 2004; Fisher and Henzinger, 2007). For PADE models, each state in the graph is described by atomic propositions whose truth values are preserved under the discrete abstraction, such as the above-mentioned derivative sign patterns. The atomic propositions are used to formulate properties in a *temporal-logic formula* ϕ and *model checkers* automatically test if the state transition graph T satisfies the formula ($T\models\phi$).

Because the number of possible parametrizations and the size of state transition graphs rapidly grow with the number of genes, the naive approach consisting in enumerating all parametrizations of a PADE model, and for each of these generating the state transition graph and testing $T\models\phi$, is only feasible for the simplest networks. We therefore propose an alternative approach, based on the *symbolic* encoding of the above search problem, without explicitly generating the possible parametrizations of the PADE models and the corresponding state transition graphs. This enables one to exploit the capability of symbolic model checkers to efficiently manipulate *implicit* descriptions of the state and parameter space.

3.3 Symbolic encoding of PADE model and dynamics

We summarize the main features of the encoding. We particularly focus on the discretization of the state space, which connects the symbolic encoding to the mathematical analysis of PADE models, and the use of the discretization for the computation of $F_i(D',D)$, which is essential for determining state transitions.

We call $\mathcal{C}$ a discretization function that maps $D\in\mathcal{D}$ to a set of unique integer coordinates, and $\mathcal{C}(D)=\mathcal{C}(D_1)\times\cdots\times\mathcal{C}(D_n)$. Let m_i be the number of non-zero parameters in $\Theta_i\cup\Lambda_i$, $i\in[1,n]$. Then $\mathcal{C}(D_i)\in\{0,1,\ldots,2m_i+1\}$, and more specifically, $\mathcal{C}(D_i)\in\{0,2,\ldots,2m_i\}$ if D_i coincides with a threshold or focal plane, and $\mathcal{C}(D_i)\in\{1,3,\ldots,2m_i+1\}$ otherwise. More generally, $\mathcal{C}(S)=\{\mathcal{C}(D)\mid D\subseteq S\}$, for any set of domains S. Obviously, $\mathcal{C}$ can also be used for the discretization of parameter values. Given the following total order on the threshold and focal parameters of variable x_{Gal4}, $0<\kappa^0_{Gal4}/\gamma_{Gal4}<\theta_{Gal4}<(\kappa^0_{Gal4}+\kappa_{Gal4})/\gamma_{Gal4}$, we find $\mathcal{C}(0)=0$ (by definition), $\mathcal{C}(\kappa^0_{Gal4}/\gamma_{Gal4})=2$, $\mathcal{C}(\theta_{Gal4})=4$ and $\mathcal{C}(\kappa^0_{Gal4}+\kappa_{Gal4})/\gamma_{Gal4})=6$.

The above discretization motivates the introduction of symbolic variables $\hat{D}_i$, $\hat{D}'_i$, $\hat{\theta}^j_i$, $\hat{\lambda}^j_i$ encoding $\mathcal{C}(D_i)$, $\mathcal{C}(D'_i)$, $\mathcal{C}(\theta^j_i)$, $\mathcal{C}(\lambda^j_i)$, respectively, with $\theta^j_i\in\Theta_i$ and $\lambda^j_i\in\Lambda_i$. The different conditions in Proposition 1 can be expressed in terms of these variables. For instance, $\mathrm{sign}(D'_i-D_i)$ becomes $\mathrm{sign}(\hat{D}'_i-\hat{D}_i)$. In the case of $F_i(D',D)$, multiplication by $1/\gamma_i$ does not change the sign, but gives the more convenient expression

$$F_i(D,D')/\gamma_i=\sum_{l\in L_i}(\kappa^l_i/\gamma_i)B^l_i(D)-D'_i \qquad (4)$$

The first term in the right-hand side is simply an interval whose upper and lower bounds are focal parameters, determined by the regulation functions $B_i^l(D)$. By redefining the step functions in terms of the symbolic variables:

$$S^+(D_j,\theta_j)=\begin{cases}[0,0] & \text{iff } \hat{D}_j<\hat{\theta}_j\\ [0,1] & \text{iff } \hat{D}_j=\hat{\theta}_j\\ [1,1] & \text{iff } \hat{D}_j>\hat{\theta}_j\end{cases} \quad (5)$$

each $B_i^l(D)$ can be simply computed using interval arithmetic. This allows the interval bounds of $\sum_{l\in L_i}(\kappa_i^l/\gamma_i)B_i^l(D)$ to be computed, which are simply given by variables $\hat{\lambda}_i^j$. Subtracting $\hat{D}'_i$ allows the sign of $F_i(D',D)$ and thus the conditions for a transition $D\to D'$ to be evaluated.

The specification of transitions in a symbolic way is the main stumble block for the efficient encoding of the PADE dynamics, especially when D is located on a threshold plane. In our previous work (Batt *et al*., 2008), the computation of transitions required the enumeration of an exponential number of domains surrounding D (Barnat *et al*., 2009). The interval-based formulation proposed here allows (the sign of) $F_i(D,D')$ to be computed in one stroke.

The implementation in a model checker such as NuSMV (Cimatti *et al*., 2002) is straightforward with the above encoding. We apply invariant constraints on the symbolic variables to exclude all valuations of $\hat{D}_i$, $\hat{D}'_i$, $\hat{\theta}_i^j$, $\hat{\lambda}_i^j$ that do not correspond to a valid transition from D to D'. We apply three types of invariants. The first ones constrain parameters to remain constant. The second ones constrain D and D' to be neighbors in the state space (e.g. $D\subseteq\partial D'$ for dimension-increasing transitions). The last ones constrain the relative position of D_i and D'_i and the parameter order as stated in the transitions conditions. For comparison with experimental data, we also need to know the variations of concentrations of gene products in each state. These correspond to the derivative sign pattern, sign$(F_i(D,D'))$.

The initial states of our symbolic description include each possible parametrization, that is, all possible values for $\hat{\theta}_i^j$ and $\hat{\lambda}_i^j$, and transition towards all states D. In CTL, a temporal logic property ϕ holds if all initial states satisfy ϕ. Therefore, by testing whether $\neg\phi$ holds, we verify the absence of a parametrization satisfying ϕ. A counterexample to $\neg\phi$ thus directly returns a valid parametrization. The current version 8 of GNA has been extended with export functionalities to generate the symbolic encoding of PADE models in the NuSMV language.

4 VALIDATION: CONSISTENCY OF IRMA NETWORK WITH EXPERIMENTAL DATA

4.1 Temporal-logic encoding of observations

Even when genetic constructs are tested separately and assembled with care, it is not obvious that a synthetic network will function in its cellular context as initially planned. Here, we test the consistency between the IRMA network and the experimental data by expressing that for each condition, switch-on and switch-off, there must exist an initial state of the system and a path starting from this state along which the gene expression changes correspond to the observed time-series data. For example, for the switch-off time-series we encode that there exists an initial state where in absence of galactose the expression of *SWI5*, *CBF1*, *GAL4* and *ASH1* decreases (in the interval $[0,10]$ min), and from which a state can be reached where the expression of *SWI5* decreases and the expression of *CBF1* increases (in the interval $[10,20]$ min), etc. The generation of this property ϕ_1 from the experimental data leads to the temporal-logic formula shown in Figure 2b. To disregard small fluctuations due to biological and experimental noise, we considered changes of magnitude less than 5×10^{-3} units not significant. Moreover, we ignore in our specification the very first measurements (in the interval $[-10,0]$), just before shifting cells to a new medium, as they probably reflect network-independent effects (Cantone *et al*., 2009).

The data presented in Cantone *et al*. (2009) for switch-on and switch-off conditions are the average of 5 and 4 individual experiments, respectively. As noticed in Section 2.2, considering the averaged gene expression profile may be misleading. Asking for consistency between our model and the result of each individual experiment might therefore be more appropriate. This leads us to define a second property ϕ_2 similar to ϕ_1 but requiring the existence of nine paths in the graph, one for each of the observed behaviors in the nine individual experiments. Although the information we extract from the experimental data only concerns trends in gene product levels, the accumulation of these simple observations leads to fairly complex constraints. Property ϕ_2 involves nearly 160 constraints on derivative signs.

4.2 Testing consistency of network with observations

We use our symbolic encoding of the PADE dynamics to test $\neg\phi_1$. NuSMV returns false, meaning that a parametrization satisfying the averaged time-series data exists (Section 3.3). The result was obtained in 49 s on a laptop (PC, 2.2 GHz, 1 core, 2 GB RAM), with an additional 100 s to provide the counterexample (Table 1). When analyzing the corresponding parametrization, the thresholds are mostly greater than the focal parameter for basal expression and smaller than the focal parameter for upregulated expression, e.g. $\kappa^0_{Ash1}/\gamma_{Ash1}<\theta_{Ash1}<(\kappa^0_{Ash1}+\kappa_{Ash1})/\gamma_{Ash1}$. This is not surprising as the focal parameters correspond to the lowest and highest possible expression levels. The threshold at which Ash1 controls *CBF1* expression is expected to lie between the two extremes. The only exception is Gal80, for which it holds $(\kappa^0_{Gal80}+\kappa_{Gal80})/\gamma_{Gal80}<\theta_{Gal80}$. According to this constraint, Gal80 plays no role in the system, since it cannot exceed the threshold concentration above which it inhibits Swi5. This is interesting because it suggests that the switch-off behavior may occur even without any inhibition by Gal80, and consequently, in a galactose-independent manner.

The dynamic properties of the PADE model can be analyzed in more detail by means of GNA. This shows the existence of an asymptotically stable steady state corresponding to switch-off conditions, with low Swi5, Gal4, Cbf1, Ash1 and Gal80 concentrations. In addition, GNA finds strongly connected components (SCCs) consistent with the observed damped oscillations in galactose media. However, the attractors co-exist irrespectively of the presence or absence of galactose, revealing that galactose does not necessarily drive the system to a single attractor for this particular parametrization.

We also tested whether the above parametrization is consistent with time-series data from the individual experiments. The model checker shows that it does not satisfy the more constraining property ϕ_2. However, we do find another parametrization for which ϕ_2 holds.

Table 1. Summary of parametrizations found by checking the consistency of the IRMA structure with the observed and desired behaviors, expressed as temporal-logic properties ϕ_1, ϕ_2 and ϕ_3. The table shows the parametrization returned when testing the truth value of the property on the symbolically encoded PADE model and gene expression profiles (left) and summarizes all parametrizations satisfying the properties (right).

	Symbolic state space and symbolic parameter space		Symbolic state space and fully parametrized models	
Property	Existence of parametrization	Parametrization[a]	Number of parametrizations	Parametrization[a]
ϕ_1: averaged time-series	Yes (49 s)	$\frac{\kappa^0_{Swi5}}{\gamma_{Swi5}} < \theta^g_{Swi5} < \theta^c_{Swi5} < \theta^a_{Swi5} < \frac{\kappa^0_{Swi5}+\kappa_{Swi5}}{\gamma_{Swi5}}$ $\wedge\ \frac{\kappa^0_{Gal80}}{\gamma_{Gal80}} < \frac{\kappa^0_{Gal80}+\kappa_{Gal80}}{\gamma_{Gal80}} < \theta_{Gal80}$	64 (885 s)	See Section 4 of Supplementary Material
ϕ_2: individual time-series	Yes (131 s)	$\frac{\kappa^0_{Swi5}}{\gamma_{Swi5}} < \theta^c_{Swi5} < \theta^a_{Swi5} < \theta^g_{Swi5} < \frac{\kappa^0_{Swi5}+\kappa_{Swi5}}{\gamma_{Swi5}}$ $\wedge\ \frac{\kappa^0_{Gal80}}{\gamma_{Gal80}} < \theta_{Gal80} < \frac{\kappa^0_{Gal80}+\kappa_{Gal80}}{\gamma_{Gal80}}$	4 (2021 s)	$\frac{\kappa^0_{Swi5}}{\gamma_{Swi5}} < \theta^c_{Swi5} < (\theta^a_{Swi5}, \theta^g_{Swi5}) < \frac{\kappa^0_{Swi5}+\kappa_{Swi5}}{\gamma_{Swi5}}$ $\wedge\ (\frac{\kappa^0_{Gal80}}{\gamma_{Gal80}}, \theta_{Gal80}) < \frac{\kappa^0_{Gal80}+\kappa_{Gal80}}{\gamma_{Gal80}}$
ϕ_3: single attractor	Yes (126 s)	$\theta^c_{Swi5} < \frac{\kappa^0_{Swi5}}{\gamma_{Swi5}} < \theta^g_{Swi5} < \theta^a_{Swi5} < \frac{\kappa^0_{Swi5}+\kappa_{Swi5}}{\gamma_{Swi5}}$ $\wedge\ \theta_{Gal80} < \frac{\kappa^0_{Gal80}}{\gamma_{Gal80}} < \frac{\kappa^0_{Gal80}+\kappa_{Gal80}}{\gamma_{Gal80}}$	7 (1300 s)	$\theta^c_{Swi5} < \frac{\kappa^0_{Swi5}}{\gamma_{Swi5}} < \theta^a_{Swi5} < \frac{\kappa^0_{Swi5}+\kappa_{Swi5}}{\gamma_{Swi5}}$ $\wedge\ \theta_{Gal80} < \frac{\kappa^0_{Gal80}+\kappa_{Gal80}}{\gamma_{Gal80}}$ $\wedge\ (\theta^g_{Swi5} < \frac{\kappa^0_{Swi5}}{\gamma_{Swi5}} \vee \theta_{Gal80} < \frac{\kappa^0_{Gal80}}{\gamma_{Gal80}})$

[a]All parametrizations shown additionally include $[\kappa^1_{Cbf1}/\gamma_{Cbf1} < \theta_{Cbf1} < (\kappa^1_{Cbf1}+\kappa^2_{Cbf1})/\gamma_{Cbf1}] \wedge [\kappa^0_{Gal4}/\gamma_{Gal4} < \theta_{Gal4} < (\kappa^0_{Gal4}+\kappa_{Gal4})/\gamma_{Gal4}] \wedge [\kappa^0_{Ash1}/\gamma_{Ash1} < \theta_{Ash1} < (\kappa^0_{Ash1}+\kappa_{Ash1})/\gamma_{Ash1}]$.

In this case, all thresholds are situated between the basal and upregulated focal parameters.

4.3 Detailed analysis of valid parameter set

Our consistency tests only confirm that a parametrization exists for which the structure of the network is consistent with the observed behavior. However, it does not say if this is trivially the case (for most parametrizations) or if the properties are selective (for only a few parametrizations). To investigate this we exhaustively generated all parametrizations, and tested for each of them properties ϕ_1 and ϕ_2. Although the total number of parameter orderings is fairly large, the exhaustive analysis is still manageable for networks of this size.

Out of the 4860 completely parametrized PADE models, we found that only a surprisingly small subset is consistent with the observations. For the averaged time series, only 64 parametrizations are consistent, while for the individual time series this subset is further reduced to 4 (Table 1). The properties extracted from the data are thus quite selective.

The results for individual time series indicate that to be consistent with the experimental data, the activation threshold of *CBF1* by Swi5 (θ^c_{Swi5}), must be smaller than the activation thresholds of *ASH1* and *GAL80* by Swi5 (θ^a_{Swi5} and θ^g_{Swi5}). Interestingly, this result is corroborated by independent measurements of promoter activities, which show that the activation threshold for the *ASH1* promoter, controlling *ASH1* and *GAL80* expression, is nearly twice as high as the one for the *HO* promoter controlling *CBF1* expression (Table S1 of Cantone *et al*., 2009).

A second finding is that the dynamics of the system is consistent with the experimental data even if $\theta_{Gal80} < \kappa^0_{Gal80}/\gamma_{Gal80}$, that is when *GAL80* is constitutively expressed above its inhibition threshold. This indicates that an effective regulation of *GAL80* expression by Swi5 is of little importance for the functioning of the network. Indeed, it was found that *GAL80* is not much responsive to changes in Swi5 availability: Cantone *et al*. observed that a 6-fold increase of *SWI5* expression leads to only a negligible (1.08-fold) increase in *GAL80* expression levels (Fig. 4A in Cantone *et al*., 2009).

5 RE-ENGINEERING: IMPROVING EXTERNAL CONTROL BY GALACTOSE

In one experiment at least, the addition of galactose does not significantly change the system's behavior: a switch-off-like response is observed in switch-on conditions. To obtain a more robust external control of the system, we would like to ensure that the addition of galactose drives the system out of the low-Swi5 state.

5.1 Temporal-logic specification of design objective

We start by specifying that in switch-off conditions the Swi5 concentration must eventually remain low, that is, equal to its basal expression level $\kappa^0_{Swi5}/\gamma_{Swi5}$. This is expressed in CTL as $\mathbf{AFAG}\ x_{Swi5}low$. In switch-on conditions, an oscillatory behavior in the concentration of Swi5 is expected. It can be formulated by means of the formula $\mathbf{AGAF}(x_{Swi5}inc \wedge \mathbf{AF}\ x_{Swi5}dec)$, requiring that an increase in x_{Swi5} is observed infinitely often and necessarily followed by a decrease in x_{Swi5}. In addition to these two basic requirements, we impose that in presence of galactose, the Swi5 concentration cannot indefinitely stay low: $u_{gal}high \rightarrow \mathbf{AF}\neg x_{Swi5}low$. We prefix these specifications so as to express the possibility (**EX**) to reach the appropriate attractor from at least one initial state, and the necessity (**AX**) to leave the switch-off steady state for all initial states in switch-on conditions:

$$\begin{aligned}\phi_3 \triangleq\ & \mathbf{EX}(u_{gal}high \wedge \mathbf{AGAF}(x_{Swi5}inc \wedge \mathbf{AF}\ x_{Swi5}dec))\\ & \wedge \mathbf{EX}(u_{gal}low \wedge \mathbf{AFAG}\ x_{Swi5}low)\\ & \wedge \mathbf{AX}(u_{gal}high \rightarrow \mathbf{AF}\neg x_{Swi5}low)\end{aligned}$$

5.2 Parametrizations consistent with design objective

Using symbolic model checking, we test the feasibility of ϕ_3. In about 2 min, we find a valid parametrization (Table 1). For this parametrization, in the presence of galactose GNA finds two terminal SCCs attracting the major part of the state space, and notably the switch-off state. In the absence of galactose, although SCCs are present, they are non-terminal and one can show that a unique stable steady state with all genes off (i.e. corresponding to switch-off conditions) is eventually always reached.

Recall that one of the time series in the switch-on conditions contradicts our specification. It is consequently not surprising that none of the parametrizations consistent with the experimental data satisfies ϕ_3. We searched for all valid parametrizations and found that only 7 out of 4860 are consistent with our specification (Table 1).

A first surprising feature is that $\theta^c_{Swi5} < \kappa^0_{Swi5}/\gamma_{Swi5}$: Swi5 must always activate *CBF1*. Stated differently, this constraint simply suggests to remove the regulation of *CBF1* by Swi5. This can be explained by a qualitative analysis of the system dynamics. In the presence of galactose, we expect oscillations for Swi5. However, the presence of Swi5 is required for the expression of *CBF1* since the *HO* promoter functions like an AND gate: *HO* is on if and only if Swi5 is present *and* Ash1 is absent. So, if Swi5 is not permanently present, Cbf1 and then Gal4 might disappear, causing the system to converge to the switch-off state.

A second surprising feature is that the regulation of *GAL80* by Swi5 should not be effective. Indeed $\theta^g_{Swi5} < \kappa^0_{Swi5}/\gamma_{Swi5}$ or $\theta_{Gal80} < \kappa^0_{Gal80}/\gamma_{Gal80}$ means that either the *GAL80* promoter is always activated, or that the Gal80 concentration is always sufficient to repress *SWI5*. As above, this suggests to remove an interaction, namely the regulation of *GAL80* by Swi5. Interestingly, the demand for increased external control of the system leads us to a simplified design in which two out of the three feedback loops (Swi5/Cbf1/Gal4/Swi5 and Swi5/Gal80/Swi5) are removed.

6 DISCUSSION

We propose a method for efficient search of the parameter space of qualitative models of regulatory networks, to investigate the relation between structural and behavioral properties of these systems.

On the methodological side, the main novelty is that we develop a *symbolic encoding* of the dynamics of PADE models, enabling the use of highly efficient model-checking tools for analyzing *incompletely parametrized models*. The symbolic encoding avoids explicit state space generation and the enumeration of possible parametrizations. We demonstrate that the proposed approach scales up to relatively complex synthetic networks. Although developed for PADE models, the main ideas underlying the approach carry over to logical models (Thomas and d'Ari, 1990).

On the biological side, we show the *practical relevance* of the approach by means of an application to the IRMA network. The parameter constraints we obtained are precise, have a clear biological interpretation, and are consistent with independent experimental observations. Even when considering complex dynamical properties, the search of the parameter space takes at most a few minutes. Our results seem to confirm the intended separation of IRMA from the host network, and suggest that to obtain a more robust response to the addition of galactose, an effective rewiring of the network would be needed.

In comparison with traditional quantitative approaches, the results we obtain are quite general, since they do not depend on specific molecular mechanisms or parameter values. Moreover, the analysis is exhaustive in the sense that the entire parameter space is scanned. These two features are particularly interesting for 'negative results', such as showing that a given design is not likely to show a desired behavior. In contrast, quantitative ODE models like those developed in Cantone *et al*. (2009) do not predict a range of possible behaviors but rather single out one likely behavior with quantitative precision. Qualitative and quantitative approaches provide complementary information on system dynamics.

In comparison with other analysis and verification methods developed for similar modeling formalisms (Barnat *et al*., 2009; Bernot *et al*., 2004; Corblin *et al*., 2009; Fromentin *et al*., 2007), our approach is original in two respects. First, it applies to incompletely parametrized models and can handle any dynamical property expressible in temporal logics supported by the model checker. Second, we reason at a finer abstraction level, in that we take into account dynamics on the thresholds and work with a partition of the state space preserving derivative sign patterns. The latter feature is particularly well-suited for the comparison of model predictions with time-series data in IRMA.

An interesting direction for further research is to consider more general problems in which not only parameters but also regulation functions are incompletely specified. This would make a connection with work on the reverse engineering of Boolean models (Martin *et al*., 2007; Perkins *et al*., 2004).

ACKNOWLEDGEMENTS

We would like to thank Delphine Ropers, Maria Pia Cosma and Diego di Bernardo for helpful discussions and contributions.

Funding: The European Commission COBIOS FP6-2005-NEST-PATH-COM/043379; the French ANR Calamar ANR-08-SYSC-003.

Conflict of Interest: none declared.

REFERENCES

Barnat,J. *et al*. (2009) On algorithmic analysis of transcriptional regulation by LTL model checking. *Theor. Comput. Sci.*, **410**, 3128–3148.

Batt,G. *et al*. (2008) Symbolic reachability analysis of genetic regulatory networks using discrete abstractions. *Automatica*, **44**, 982–989.

Batt,G. *et al*. (2005) Validation of qualitative models of genetic regulatory networks by model checking. *Bioinformatics*, **21** (Suppl. 1), i19–i28.

Bernot,G. *et al*. (2004) Application of formal methods to biological regulatory networks. *J. Theor. Biol.*, **229**, 339–348.

Cantone,I. *et al*. (2009) A yeast synthetic network for *in vivo* assessment of reverse-engineering and modeling approaches. *Cell*, **137**, 172–181.

Chaves,M. *et al*. (2009) Geometry and topology of parameter space: investigating measures of robustness in regulatory networks. *J. Math. Biol.*, **59**, 315–358.

Cimatti,A. *et al*. (2002) NuSMV2: an opensource tool for symbolic model checking. In *CAV'02*, Vol. 2404 of *LNCS*. Springer, pp. 359–364.

Clarke,E.M. *et al*. (1999) *Model Checking*. MIT Press, Cambridge, USA.

Corblin,F. *et al*. (2009) A declarative constraint-based method for analyzing discrete genetic regulatory networks. *Biosystems*, **98**, 91–104.

Davidich,M. and Bornholdt,S. (2008) The transition from differential equations to Boolean networks: a case study in simplifying a regulatory network model. *J. Theor. Biol.*, **255**, 269–277.

Edwards,R. and Glass,L. (2006) A calculus for relating the dynamics and structure of complex biological networks. In Berry,R.S. and Jortner,J. (eds) *Adventures in Chemical Physics*, Vol. 132. Wiley, Hobeken, USA, pp. 151–178.

Fauré,A. *et al*. (2006) Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle. *Bioinformatics*, **22**, e124–e131.

Fisher,J. and Henzinger,T.A. (2007) Executable cell biology. *Nat. Biotechnol.*, **25**, 1239–1250.

Fromentin,J. *et al*. (2007) Analysing gene regulatory networks by both constraint programming and model-checking. In *IEEE EMBC07*, pp. 4595–4598.

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

Gouzé,J.-L. and Sari,T. (2002) A class of piecewise linear differential equations arising in biological models. *Dyn. Syst.*, **17**, 299–316.

Martin,S. *et al*. (2007) Boolean dynamics of genetic regulatory networks inferred from microarray time series data. *Bioinformatics*, **23**, 866–874.

Monteiro,P.T. *et al*. (2008) Temporal logic patterns for querying dynamic models of cellular interaction networks. *Bioinformatics*, **24**, i227–i233.

Moore,R.E. (1979) *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, USA.

Perkins,T.J. *et al*. (2004) Inferring models of gene expression dynamics. *J.Theor. Biol.*, **230**, 289–299.

Polynikis,A. *et al*. (2009) Comparing different ODE modelling approaches for gene regulatory networks. *J. Theor. Biol.*, **261**, 511–530.

Saez-Rodriguez,J. *et al.* (2009) Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction. *Mol. Syst. Biol.*, **5**, 331.

Thomas,R. and d'Ari,R. (1990) *Biological Feedback*. CRC Press, Boca Raton, USA.

Tsai,T.Y.-C. *et al*. (2008) Robust, tunable biological oscillations from interlinked positive and negative feedback loops. *Science*, **321**, 126–129.

BIOINFORMATICS ORIGINAL PAPER Vol. 23 no. 18 2007, pages 2415–2422
doi:10.1093/bioinformatics/btm362

Systems Biology

Robustness analysis and tuning of synthetic gene networks

Grégory Batt[1,*], Boyan Yordanov[2], Ron Weiss[3] and Calin Belta[1]

[1]Centers for Information and Systems Engineering and for BioDynamics, [2]Department of Biomedical Engineering, Boston University, Boston, MA and [3]Departments of Electrical Engineering and Molecular Biology, Princeton University, Princeton, NJ, USA

Received on March 17, 2007; revised on June 8, 2007; accepted on July 8, 2007

Advance Access publication July 27, 2007

Associate Editor: Chris Stoeckert

ABSTRACT

Motivation: The goal of synthetic biology is to design and construct biological systems that present a desired behavior. The construction of synthetic gene networks implementing simple functions has demonstrated the feasibility of this approach. However, the design of these networks is difficult, notably because existing techniques and tools are not adapted to deal with uncertainties on molecular concentrations and parameter values.

Results: We propose an approach for the analysis of a class of uncertain piecewise-multiaffine differential equation models. This modeling framework is well adapted to the experimental data currently available. Moreover, these models present interesting mathematical properties that allow the development of efficient algorithms for solving robustness analyses and tuning problems. These algorithms are implemented in the tool RoVerGeNe, and their practical applicability and biological relevance are demonstrated on the analysis of the tuning of a synthetic transcriptional cascade built in *Escherichia coli*.

Availability: RoVerGeNe and the transcriptional cascade model are available at http://iasi.bu.edu/%7Ebatt/rovergene/rovergene.htm

Contact: gregory.batt@imag.fr

1 INTRODUCTION

The main goal of the nascent field of synthetic biology is to design and construct biological systems that present a desired behavior (Andrianantoandro *et al.*, 2006; Endy, 2005). Synthetic biology is foreseen to have important applications in biotechnology and medicine, and to contribute significantly to a better understanding of the functioning of complex biological systems (McDaniel and Weiss, 2005). The construction of networks of interregulating genes, so-called genetic regulatory networks, has demonstrated the feasibility of this approach (e.g. Gardner *et al.*, 2000). Still, the development of gene networks is difficult: most newly created networks are non-functioning and need tuning. One important reason is that the lack of precise knowledge on molecular concentrations and on parameter values hampers the design of synthetic networks. These uncertainties are the consequence of current technological limitations and also of the fluctuations of intra- and extracellular environments.

Existing solutions for the analysis of dynamical properties of gene networks consist essentially either in qualitative simulation of coarse-grained models or in extensive numerical simulations of nonlinear differential equation models or stochastic versions thereof (de Jong, 2002; Szallasi *et al.*, 2006). For applications in synthetic biology, these approaches are not satisfying. For qualitative models, the predictions obtained are generally too coarse for answering the—often quantitative—questions of interest. For uncertain quantitative models, a common approach is to perform many numerical simulations so as to 'sample' the state and parameter spaces, often in conjunction with local sensitivity analyses. This approach provides only a partial description of all the possible behaviors of a network. In particular, it cannot provide the guaranty that a network behaves as expected for all initial conditions and parameters in given ranges. Moreover, obtaining a 'reasonably dense' coverage of the state and parameter spaces quickly becomes computationally intractable when the size of the networks grows.

In this work, we demonstrate the biological relevance of a method specifically developed to support the *design of synthetic gene networks*. This method allows to analyze dynamical properties of uncertain, yet quantitative models of gene networks. More precisely, we consider piecewise-multiaffine differential equation models in which uncertain initial conditions and parameters are given by intervals. These models capture essential aspects of genetic regulations and still allow for efficient analyses by tailored formal verification techniques. Dynamical properties of the network are given by temporal logic formulas that specify temporal constraints on the state of the system, that is, on protein concentrations. Temporal logics are specification languages that allow to express a variety of properties on the behavior of dynamical systems (Emerson, 1990). Then, the proposed approach allows to *check* automatically that a network satisfies a given dynamical property for all initial conditions and all parameter values in the given intervals. This provides us a means to *assess the robustness* of the expected behavior of a network with respect to parameter variations. In particular, our technique does *not* rely on numerical simulations. Additionally, the proposed approach has the capability to generate constraints on parameters, and can consequently be extended to *search* for parameter sets for which a given property is satisfied. This feature allows to solve *network tuning problems* by suggesting modifications of biological parameters. These techniques are implemented in a publicly available tool called RoVerGeNe (for Robust Verification of Gene Networks) and their applicability and biological relevance is demonstrated on the analysis of the tuning of a synthetic transcriptional cascade.

*To whom correspondence should be addressed.

The remainder of this article is organized as follows. In the next section, we provide a brief description of the proposed method and of its implementation in the computer tool RoVerGeNe. In Section 3, we detail the application of our method to the tuning of a synthetic transcriptional cascade built in *Escherichia coli*. The results are summarized in the last section. We refer the reader to (Batt *et al.*, 2007a, b) for a detailed presentation of the method and for computational results using a preliminary version of the transcriptional cascade model.

2 ANALYSIS OF PIECEWISE-MULTIAFFINE MODELS WITH PARAMETER UNCERTAINTY

In this section, we provide an intuitive overview of the proposed approach by means of a simple example.

2.1 Piecewise-multiaffine models and LTL properties

Consider the cross-inhibition network represented in Figure 1. The network is made of two genes, a and b, that code for two repressor proteins, A and B. More specifically, protein B represses the expression of gene a, whereas protein A represses the expression of gene b, and at a higher concentration, the expression of its own gene. Protein degradations are not regulated.

This system can be modeled by differential equations as follows.

$$\dot{x}_a = \kappa_a\, r_{a1}(x_b)\, r_{a2}(x_a) - \gamma_a\, x_a, \tag{1}$$

$$\dot{x}_b = \kappa_b\, r_b(x_a) - \gamma_b\, x_b, \tag{2}$$

with $x = (x_a, x_b) \in \mathcal{X} = [0, \max_a] \times [0, \max_b]$. The state variables x_a and x_b denote the concentrations of protein A and B. x is the vector of state variables and $\mathcal{X}$ is the state space. $\max_a$ and $\max_b$ denote a maximal concentration for proteins A and B. κ's and γ's are respectively *production* and *degradation rate parameters*, and r's are *regulation functions*. The latter capture the regulatory effect of an effector protein on gene expression. In contrast to most nonlinear models in which the regulation functions are smooth sigmoidal functions (e.g. Hill functions) (de Jong, 2002), we assume that regulation functions are *piecewise-affine* (Fig. 2). These functions are uniquely defined by their values at *breakpoints*, denoted by λ's. For our example model, we used the simplest piecewise-affine functions approximating sigmoidal curves: ramp functions. These functions have only four break points (including 0 and $\max_i$). The ordered set of all breakpoints associated with the variable x_i is denoted by Λ_i. For example, we have $\Lambda_a = \{0, \lambda_{a1}, \lambda_{a2}, \lambda_{a3}, \lambda_{a4}, \max_a\}$ and $\Lambda_b = \{0, \lambda_{b1}, \lambda_{b2}, \max_b\}$.

Products of regulation functions (involving different state variables) can be used to capture complex genetic regulations. In Equation (1), for example, the product of regulation functions captures the hypothesis that in order to have a maximal expression of gene a both proteins must be present in low concentration (i.e. below λ_{a3} and λ_{b1}). Because products of piecewise-affine functions are allowed, the resulting models are in general *piecewise-multiaffine*. We recall that a multiaffine function is a polynomial with the property that the degree in any of its variable is at most 1 (Belta and Habets, 2006). In particular, products of different variables are allowed. A motivation for considering piecewise-affine regulation functions is that

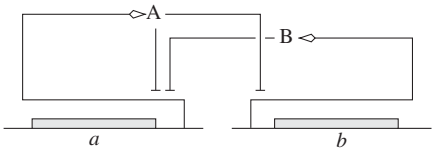


Fig. 1. Cross-inhibition network.

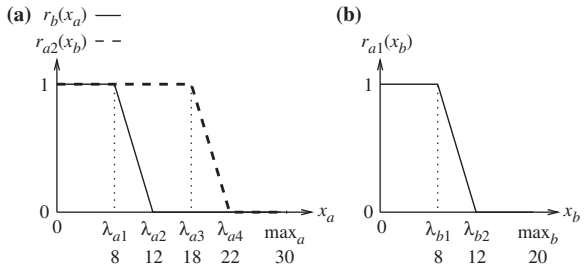


Fig. 2. Regulation functions in Equations (1) and (2) for the cross-inhibition network.

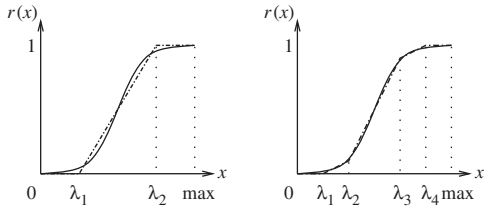


Fig. 3. Approximation of sigmoidal functions by piecewise-affine functions. Better approximations can be obtained by using more breakpoints.

piecewise-affine functions have universal approximation properties (Lin and Unbehauen, 1992), which means that any nonlinear function can be approximated by a piecewise-affine function with arbitrary accuracy (Fig. 3). Moreover, while the analysis of general nonlinear systems (e.g. Hill-type models) is notoriously difficult, efficient approaches have been recently developed for multiaffine systems (Belta and Habets, 2006).

Some parameters might be uncertain. Their values are then given by intervals. We assume that production and/or degradation rate parameters can be uncertain (i.e. κ's and γ's), but that regulation functions are precisely known. We denote by p the vector of uncertain parameters and by $\mathcal{P}$ the parameter space. For the cross-inhibition network, we assume that parameters satisfy $\kappa_a \in [0, 30]$, $\kappa_b \in [0, 40]$, $\gamma_a = 1$, and $\gamma_b = 2$. So, we have $p = (\kappa_a, \kappa_b) \in \mathcal{P} = [0, 30] \times [0, 40]$. More generally, the models that we consider are *piecewise-multiaffine (PMA) systems* Σ of the general form

$$\dot{x} = f(x, p), \quad x \in \mathcal{X}, p \in \mathcal{P} \tag{3}$$

where f is a piecewise-multiaffine function of the state variables x and an affine function of the uncertain parameters p.

A number of different formalisms has been proposed to describe gene networks (de Jong, 2002). The use of piecewise-multiaffine models for gene networks was first proposed by Belta *et al.* (2002) (see Mestl *et al.*, 1995 for a related, piecewise-continuous approach). The class of piecewise-multiaffine models that we consider is also related to the class

of piecewise-affine (PA) differential equation models proposed by Glass and Kauffman (1973). Even if multiaffine models do not present the monotonicity properties that make the qualitative, symbolic analysis of PA models attractive (de Jong *et al*., 2004; Ghosh and Tomlin, 2004; see also Kauffman, 1969; Thomas *et al*., 1995, for alternative, discrete formalisms), the use of piecewise-affine functions to represent genetic regulations (instead of step functions for PA models) allows to develop finer-grained models, better adapted to quantitative analyses. In particular, PMA models capture the graded response of gene expression to continuous changes in effector (activator or repressor) concentrations.

We use Linear Temporal Logic (LTL) to express dynamical properties of gene networks. Temporal logics have been developed to specify the behavior of (usually discrete) dynamical systems (Emerson, 1990). Typical properties include reachability (the system can reach a given state), inevitability (the system will necessarily reach a given state), invariance (a property is always true), response (an event necessarily triggers a specific behavior) and infinite occurrences (such as oscillations). Illustrative examples of the expressiveness of temporal logics in systems biology can be found in Antoniotti *et al*. (2003), Batt *et al*. (2005), Bernot *et al*. (2004) and Fages *et al*. (2004). LTL formulas are built using atomic propositions *and* LTL operators. In our approach, atomic propositions express simple constraints on protein concentrations and are of type '$x_i < \lambda$' or '$x_i > \lambda$' .[1] LTL operators include the usual logical operators, such as *negation* ($\neg$), logical *and* ($\wedge$), logical *or* ($\vee$), and *implication* ($\rightarrow$), and specific temporal operators, such as *future* ($\mathbf{F}$), *globally* ($\mathbf{G}$) and *until* ($\mathbf{U}$). $\mathbf{F}\,p$, $\mathbf{G}\,p$ and $p\,\mathbf{U}\,q$ respectively mean that a property p holds at some future time, holds for all future times, or holds continuously until an other property q holds. These operators can be combined to express complex dynamical properties.

The cross-inhibition network is known to be *bistable*. If the system is in a state in which the concentration of protein A is low and the concentration of protein B is high, then it will remain in such a state for all time. A symmetrical property holds with the concentrations of A and B being high and low, respectively. This property can be expressed in LTL by the formula ϕ_1, where, for example, the first part of the property expresses that if the concentrations of protein A and B are respectively low ($x_a < \lambda_{a1}$) and high ($x_b > \lambda_{b2}$), then the system will always ($\mathbf{G}$) remain in such a state.

$$\begin{aligned}\phi_1 = {} & (x_a < \lambda_{a1} \wedge x_b > \lambda_{b2} \rightarrow \mathbf{G}\,(x_a < \lambda_{a1} \wedge x_b > \lambda_{b2})) \\ & \wedge (x_b < \lambda_{b1} \wedge x_a > \lambda_{a3} \rightarrow \mathbf{G}\,(x_b < \lambda_{b1} \wedge x_a > \lambda_{a3}))\end{aligned} \quad (4)$$

The semantics of LTL formulas is defined over executions of transition systems (Emerson, 1990). Transition systems consist of a (finite or infinite) set of states and of a set of transitions between states. Transition systems define a set of executions, which are sequences of states for which there exists a transition from each state to its successor. So, in order to define what it means that a PMA system Σ satisfies an LTL property ϕ for a given parameter $p \in \mathcal{P}$, we introduce an embedding transition system, denoted by $T_{\mathcal{X}}(p)$, in which the states are the points x in $\mathcal{X}$, and the transitions between two points correspond to the existence of a solution of the differential equation (3) going from one point to the other. Consequently, executions of $T_{\mathcal{X}}(p)$ correspond to solution trajectories of (3). Then, a PMA system Σ satisfies an LTL property ϕ for a given parameter p if every execution of the associated embedding transition system $T_{\mathcal{X}}(p)$ satisfies the property ϕ, denoted by $T_{\mathcal{X}}(p) \models \phi$. We say that the parameter p is *valid* for ϕ. Finally, a parameter *set* P is valid for ϕ if every parameter in P is valid for ϕ. In this work, we consider the following two problems.

Problem *Let Σ be a PMA system, $\mathcal{P}$ an hyperrectangular parameter space, and ϕ an LTL formula.*
Problem 1. Robustness analysis: *Check whether $\mathcal{P}$ is valid for ϕ.*
Problem 2. Tuning: *Find a set $P \subseteq \mathcal{P}$ such that P is valid for ϕ.*

The state space associated with our two-gene example is shown in Figure 4a. The flow and a solution trajectory passing through three points, x^1, x^2 and x^3, are also represented for a given parameter $\hat{p} = (26, 34)$. In $T_{\mathcal{X}}(\hat{p})$, there is for example a transition from x^1 to x^2, and from x^2 to x^3. The solution trajectory represented in Figure 4a can be associated with the execution $(x^1, x^2, x^3, \ldots)$.

2.2 Analysis of uncertain PMA systems

Problems 1 and 2 amount to prove that a given property is satisfied for sets of initial conditions and for sets of parameters. Consequently, these problems cannot be solved by numerical integration, since it would require to check whether the property holds for an infinite number of solution trajectories. Instead, we use a combination of techniques developed for the verification of continuous, and more generally hybrid (i.e. continuous and discrete) dynamical systems. The principle of the analysis is simple. *Discrete abstractions* (Alur *et al*., 2000) are used to transpose problems defined on (infinite) continuous state and parameter spaces into problems defined on (finite) discrete spaces. Algorithmic analysis by *model checking* (Clarke *et al*., 1999) is then possible.

The first step of our analysis is to define a partition of the state space. Given the piecewise nature of the differential equation system (3), it is natural to partition the state space into regions in which the differential equations have a same expression. So we consider the hyperrectangular partition defined by the breakpoints in Λ_i for every variable x_i (Fig. 4a). Full-dimensional regions of the partition are called *rectangles* $R \in \mathcal{R}$. For our example network, $\mathcal{R}$ contains 15 rectangles: $R^1, \ldots, R^{15}$ (Fig. 4a).

For every parameter set P, we define the discrete abstraction of a PMA system Σ as the discrete transition system $T_{\mathcal{R}}(P)$ in which the states are the rectangles, and the transitions between (adjacent) rectangles correspond to the existence of solution trajectories of (3) for some parameter in P, going from one rectangle to the other. In Batt *et al*. (2007a), we have shown that the discrete abstraction $T_{\mathcal{R}}(P)$ captures every possible behavior of the original system Σ for every parameter $p \in P$. More precisely, $T_{\mathcal{R}}(P)$ is a *conservative approximation* of Σ, in the sense that to every solution trajectory of (3), there exists a corresponding execution in $T_{\mathcal{R}}(P)$.[2] Note however, that $T_{\mathcal{R}}(P)$

[1]Note that the assumption $\lambda \in \Lambda_i$ is made without loss of generality.

[2]In fact, this property holds only for *almost all* solution trajectories of (3). For our biological applications, this technical restriction is of no practical importance and is disregarded in the sequel (see Batt *et al*., 2007a).

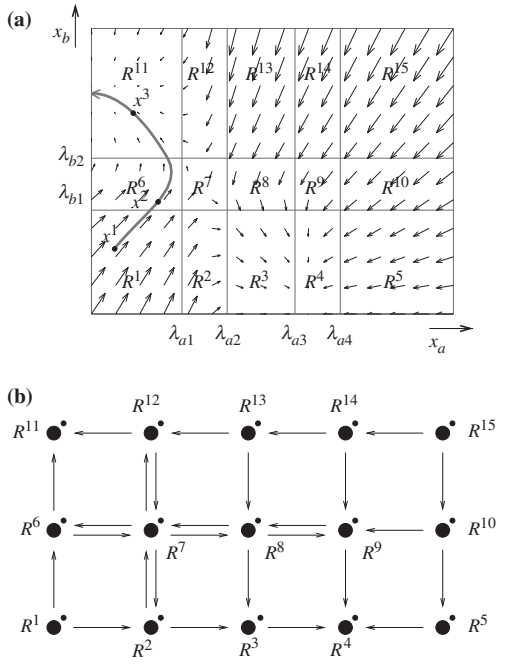


Fig. 4. (**a**) Continuous dynamics in the state space of the cross-inhibition network for a given parameter, $\hat{p} = (\kappa_a, \kappa_b) = (26, 34)$. (**b**) Discrete abstraction $T_{\mathcal{R}}(\hat{P})$ for the parameter set $\hat{P} = [20, 30] \times [30, 40]$. Dots denote self transitions.

may contain spurious executions, that is, executions corresponding to no solution trajectory of the original system. For our example network, the discrete abstraction of the system associated with the parameter set $\hat{P} = [20, 30] \times [30, 40] \subseteq \mathcal{P}$ is represented in Figure 4b. For parameter $\hat{p} \in \hat{P}$, there exists a solution reaching R^6 from R^1, and R^{11} from R^6 (Fig. 4a), so there exists transitions from R^1 to R^6 and from R^6 to R^{11} in $T_{\mathcal{R}}(\hat{P})$ (Fig. 4b). The solution trajectory represented in Figure 4a corresponds to the execution $(R^1, R^6, R^{11}, R^{11}, \ldots)$ in $T_{\mathcal{R}}(\hat{P})$.

Contrary to the original, infinite system, the abstract system being finite can be analyzed by model checking techniques. Model checking techniques are highly efficient automatic techniques developed for the analysis of finite transition systems. In particular, off-the-shelf tools exist to check whether discrete transition systems satisfy given temporal logic properties. Using these tools, we can test whether $T_{\mathcal{R}}(P) \models \phi$, and if this holds, we can conclude that the original system Σ satisfies the property ϕ for every parameter in P using the fact that conservative approximations weakly preserve LTL (Browne *et al.*, 1988): if a property is true for the abstract system, then it holds for the original system. Note, however, that due to the possible existence of spurious executions in the abstract system, the converse is not necessarily true. Stated differently, we might fail to prove some properties.

It is easy to check on the discrete transition system $T_{\mathcal{R}}(\hat{P})$ represented in Figure 4b that the network satisfies the bistability property ϕ_1 for every parameter value in $\hat{P}$. If the state $x = (x_a, x_b)$ of the system satisfies $x_a < \lambda_{a1}$ and $x_b > \lambda_{b2}$, then $x \in R^{11}$ (Fig. 4a), and because there is no transition leaving R^{11} in $T_{\mathcal{R}}(\hat{P})$ (Fig. 4b), the system can not leave R^{11}. By a similar reasoning, one can check that the second half of property ϕ_1 also holds (the system always remains in R^4 or R^5, where protein A and B concentrations are respectively high and low).

We have still not provided a means to actually *compute* the discrete abstraction $T_{\mathcal{R}}(P)$. In fact, we need to be able to decide whether solutions starting from a rectangle can enter an adjacent rectangle. For general, uncertain nonlinear dynamical system, there is no known method to solve this problem. Fortunately, we can exploit two specific properties of the class of models that we consider. First, because the models are piecewise-multiaffine functions of the state variables, the existence of transitions between two adjacent rectangles only depends on the direction of the vector field at the vertices of the facet that separates the two rectangles. This comes from convexity properties of multiaffine functions in hyperrectangular regions (Belta and Habets, 2006). Second, because the models are affine functions of the uncertain parameters, the vector field at a vertex v depends affinely on the unknown parameters. So we can show that the set of parameters for which there exists a transition between two rectangles corresponds to a union of polyhedral sets in the parameter space (Batt *et al.*, 2007a). As a consequence, the discrete transition system $T_{\mathcal{R}}(P)$ can be computed by means of polyhedral operations for a hyperrectangular, or more generally, for a polyhedral parameter set P.

For the cross-inhibition network, there exists a transition from R^1 to R^2 if and only if the vector field at one of the vertices of the separating facet (of coordinates $(\lambda_{a1}, 0)'$ or $(\lambda_{a1}, \lambda_{b1})'$) points 'to the right' (i.e. is such that $\dot{x}_a > 0$). It holds for both vertices that $\dot{x}_a = \kappa_a - \gamma_a \lambda_{a1}$, which is positive for every $\kappa_a \in [20, 30]$ ($\gamma_a = 1$ and $\lambda_{a1} = 8$). So there is a transition from R^1 to R^2 in $T_{\mathcal{R}}(\hat{P})$. Conversely, one can show that there is no transition from R^2 to R^1 in $T_{\mathcal{R}}(\hat{P})$.

We can now solve robustness problems (Problem 1) by the following two-step procedure: first, compute the discrete abstraction $T_{\mathcal{R}}(P)$ by means of polyhedral operations, and second, test on the discrete abstraction whether the property ϕ is true by model checking. If it is true, then we can conclude that the property is true for all parameters in the parameter set, or stated differently, that the parameter set P is valid for ϕ. Note, however, that if the discrete abstraction does not satisfy the property, no conclusion can be drawn on the original system.

In order to deal with tuning problems (Problem 2), we use the observation made previously that the existence of transitions in the discrete abstractions depends on a set of affine constraints on parameters. All these constraints define a polyhedral partition $\mathcal{P}$ of the parameter space, represented in Figure 5 for our example network. All parameters in a same region $P \in \mathcal{P}$ are equivalent, in the sense that they are associated with a same discrete abstraction.

Then, a naive approach to find solutions to Problem 2 is to test the validity of every parameter equivalence class $P \in \mathcal{P}$ of the parameter space using the previous approach (i.e. for every $P \in \mathcal{P}$, compute $T_{\mathcal{R}}(P)$ and test whether $T_{\mathcal{R}}(P) \models \phi$). Every parameter set identified this way provides solutions to the tuning problem, since it suggests a way to modify network parameters such that the tuned system is guaranteed to satisfy the expected property. Conversely, not all valid parameters are guaranteed to be found by our approach. For our example network, if we test the validity of the regions $P^1, P^2, \ldots, P^{12}$

represented in Figure 5, we find that P^{12} is a valid parameter set. So we can conclude that the gene network is bistable if $\kappa_a > 18$ and $\kappa_b > 24$. In fact, we have developed a more efficient approach that allows to reason with fewer, larger regions in parameter space, corresponding to unions of parameter equivalence classes $P \in \mathcal{P}$ (Batt *et al.*, 2007a). Still, computational times generally increase exponentially with the number of genes and uncertain parameters. Consequently, the applicability of our method is currently limited to the analysis of networks of moderate size (i.e. having less than a half-dozen genes) as currently encountered in most synthetic gene networks. The analysis of future, larger synthetic networks will require to extend our approach to exploit their modularity (Chin, 2006).

This method has been implemented in a freely available tool for Robust Verification of Gene Networks (RoVerGeNe), written in Matlab on top of several other tools (MPT, MatlabBGL, NuSMV). Additionally, RoVerGeNe supports an extension of the method presented here, dealing with problems specifically encountered when verifying liveness properties (Batt *et al.*, 2007b).

3 TUNING OF A SYNTHETIC TRANSCRIPTIONAL CASCADE

In this section, we illustrate the practical applicability and biological relevance of the approach presented in the previous section for the analysis of synthetic gene networks.

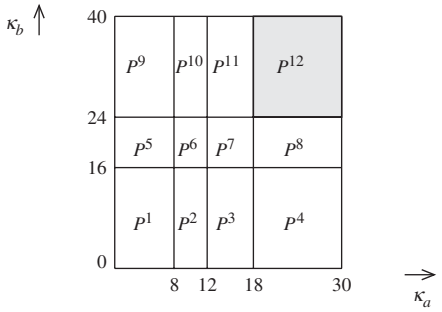


Fig. 5. Partition of the parameter space $\mathcal{P}$ of the cross-inhibition network. Valid regions are shaded.

3.1 Problem

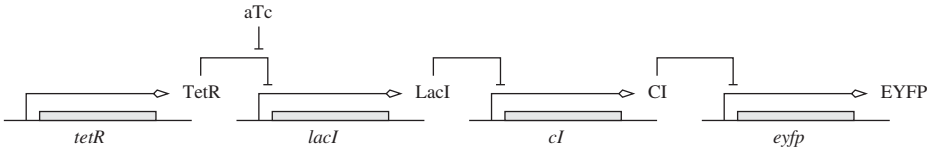
We consider a cascade of transcriptional inhibitions built in *E.coli* by Hooshangi *et al.* (2005). The network is represented in Figure 6a. It is made of four genes: *tetR*, *lacI*, *cI*, and *eyfp* that code respectively for three repressor proteins, TetR, LacI and CI, and the fluorescent protein EYFP. The fluorescence of the system, due to the protein EYFP, is the measured output. The system can be controlled by the addition or removal of a small diffusible molecule, aTc, in the growth media. More precisely, aTc binds to TetR and relieves the repression of *lacI*. The aTc concentration thus serves as a controllable input to the system. It is intuitively clear that the output (i.e. the fluorescence) of the system at steady state will be low for low inputs (i.e. aTc concentration), and high for high inputs. Moreover, because of the topology of the network (cascade of inhibitions), an *ultrasensitive* response may be achieved: the output at steady state undergoes a dramatic change for a moderate change of the input in a transition region. More precisely, we would like that the system at steady state satisfies the input/output specifications represented in Figure 7a, in which the output of the system is expected to remain between the two dotted lines. In particular, this specifies that a 1000-fold increase of the output is obtained for a 4-fold increase of the input.

Unfortunately, the actual network does not meet these specifications (Fig. 7a). So we used our method and tool to investigate how to tune it. In a preliminary step, we have developed a PMA model of the network. Then, using RoVerGeNe, we have investigated the possibility to tune the network by modifying some of its parameters (Problem 2), proposed parameter modifications, and evaluated computationally the robustness of the modified system (Problem 1). Note that it is important to perform this last step before experimentally tuning the network in order to gain confidence that the tuned system will behave as expected despite errors in parameter identification, incorrect parameter modifications or environmental fluctuations.

3.2 Modeling and specification

We have developed a piecewise-multiaffine model of the cascade, represented in Figure 6b. The notations are similar

(a)



(b)

$$\dot{x}_{tetR} = \kappa_{tetR} - \gamma_{tetR}\, x_{tetR}, \tag{5}$$

$$\dot{x}_{lacI} = \kappa_{lacI,0} + \kappa_{lacI}\,(r_{lacI,1}(x_{tetR}) + r_{lacI,2}(u_{aTc}) - r_{lacI,1}(x_{tetR})\, r_{lacI,2}(u_{aTc})) - \gamma_{lacI}\, x_{lacI}, \tag{6}$$

$$\dot{x}_{cI} = \kappa_{cI,0} + \kappa_{cI}\, r_{cI}\,(x_{lacI}) - \gamma_{cI}\, x_{cI}, \tag{7}$$

$$\dot{x}_{eyfp} = \kappa_{eyfp,0} + \kappa_{eyfp}\, r_{eyfp}(x_{cI}) - \gamma_{eyfp}\, x_{eyfp}, \tag{8}$$

$$\dot{u}_{aTc} = 0. \tag{9}$$

Fig. 6. (**a**) Synthetic transcriptional cascade. TetR represses *lacI*, LacI represses *cI*, and CI represses *eyfp*. aTc controls the repression of *lacI* by TetR. The fluorescence of the protein EYFP is the output. (**b**) Piecewise-multiaffine model of the cascade in (a). The concentrations of protein TetR, LacI, CI, EYFP and of aTc are denoted by x_{tetR}, x_{lacI}, x_{cI}, x_{eyfp} and u_{aTc}, respectively. Other notations follow those introduced in Section 2.1.

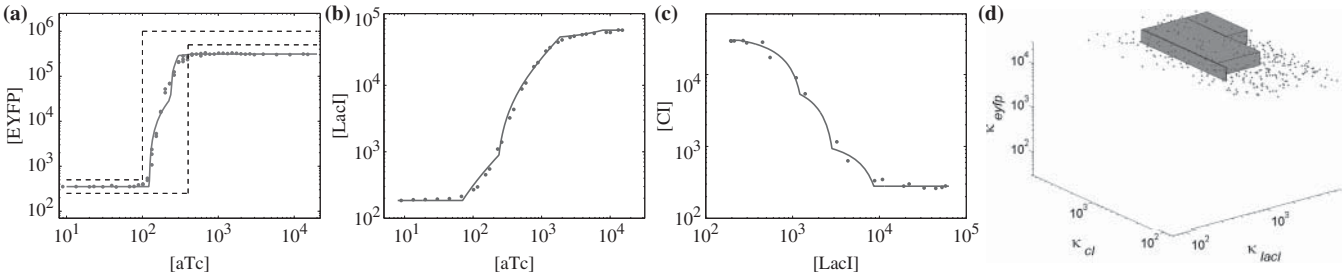


Fig. 7. (**a**) Steady-state input/output behavior of the cascade: desired (region delimited by black dashed lines) measured (red dots) and predicted (red line). (**b** and **c**) Relations between the concentrations of (b) LacI and aTc and (c) CI and LacI of the cascade at steady-state: experimental data (dots) and piecewise affine fits (solid lines). Note that the curved shape of the line segments is due to the log-log representation used. (**d**) Valid parameters in the parameter space as identified by RoVerGeNe (rectangular regions) or by brute-force sampling (dots). κ_{lacI}, κ_{cI} and κ_{eyfp} are production rate parameters for protein LacI, CI and EYFP, respectively.

to those introduced in Section 2. For regulated genes, two production terms are distinguished: a leakage term (with subscript 0) and a regulated term. The expression in Equation (6) for the regulation of *lacI* by TetR and aTc states that the expression of *lacI* increases if the concentration of aTc increases ($r_{lacI,2}$ is an increasing function since aTc is an activator) or if the concentration of TetR decreases ($r_{lacI,1}$ is a decreasing function since TetR is a repressor). We recall that the function $d(a,b) = a + b - ab$ increases when a or b increase and that $d(a,b) \in [0,1]$ if $a \in [0,1]$ and $b \in [0,1]$. It can thus be considered as an arithmetic equivalent of the logical *or*. Finally, Equation (9) states that the concentration of aTc is constant.

Because the proteins in the cascade are relatively stable, we neglected protein degradation and assumed that degradation rate parameters were simply equal to the dilution rate corresponding to the observed division time of the cells (about 45 minutes). Under the assumption that the concentration at steady state of the constitutively expressed protein TetR is sufficient to fully repress the expression of *lacI* (i.e. $r_{lacI,1}(x^*_{tetR}) = 0$), we deduce from our model that the concentrations at steady state of the proteins LacI, CI and EYFP satisfy the following relations.

$$x^*_{lacI} = \kappa_{lacI,0}/\gamma_{lacI} + \kappa_{lacI}/\gamma_{lacI}\ r_{lacI,2}(u_{aTc}) \quad (10)$$

$$x^*_{cI} = \kappa_{cI,0}/\gamma_{cI} + \kappa_{cI}/\gamma_{cI}\, r_{cI}(x^*_{lacI}) \quad (11)$$

$$x^*_{eyfp} = \kappa_{eyfp,0}/\gamma_{eyfp} + \kappa_{eyfp}/\gamma_{eyfp}\, r_{eyfp}(x^*_{cI}) \quad (12)$$

Under the assumption that the concentration of the protein EYFP corresponds to the fluorescence intensities measured for this cascade, and that the concentrations of the intermediate proteins of the cascade LacI and CI correspond to the fluorescence intensities of other, shorter cascades (not shown here, (Hooshangi *et al.*, 2005)), experimental data is available to describe these relations. More precisely, the relation between x^*_{lacI} and u_{aTc} is directly known from measurements, and the relations between x^*_{cI} and x^*_{lacI}, and x^*_{eyfp} and x^*_{cI} can be deduced from the experimentally-measured relations between x^*_{lacI} and u_{aTc}, x^*_{cI} and u_{aTc}, and x^*_{eyfp} and u_{aTc}. Then, existing techniques for fitting piecewise-affine functions to data can be used to identify the values of production rate parameters and the piecewise-affine regulation functions appearing in Equations (10)–(12) (Fig. 7b and c). We used an in-house implementation of the algorithm proposed in (Ferrari-Trecate *et al.*, 2001) that imposes the identification of horizontal plateaus. To obtain better fits, we interpolated the experimental data with splines, and fitted the piecewise-affine functions to the interpolated data.[3] For TetR, no experimental data was available. So, we have simply chosen a ramp function for $r_{lacI,1}(x_{tetR})$ and parameter values that guarantee that at steady state the concentration of TetR ($x^*_{tetR} = \kappa_{tetR}/\gamma_{tetR}$) is sufficient to fully repress the expression of $lacI$.

In order to assess the validity of the model, we compared model predictions with experimental data. For various concentrations of aTc and for randomly chosen initial concentrations, we computed the steady state of the network by numerical simulation (Fig. 7a). Given the simplicity of the model, we obtained a reasonably good fit between data and predictions. Additionally, we simulated the network response to the addition or removal of aTc and obtained a good agreement with experimental data (data not shown).

The last step was to formalize in temporal logic the desired behavior of the network depicted in Figure 7a. This specification can be expressed as a conjunction of three constraints of the type: if the aTc concentration is in a given range, then the concentration of EYFP at steady state must be in another given range:

$$\begin{aligned}\phi_2 = \quad & u_{aTc} < 100 \rightarrow \mathbf{FG}(2.5\,10^2 < x_{eyfp} < 5\,10^2)\\ \wedge\ & 100 < u_{aTc} < 400 \rightarrow \mathbf{FG}(2.5\,10^2 < x_{eyfp} < 10^6)\\ \wedge\ & u_{aTc} > 400 \rightarrow \mathbf{FG}(5\,10^5 < x_{eyfp} < 10^6),\end{aligned}$$

where to express that a property p holds at steady state, we used $\mathbf{FG}p$, meaning 'eventually (**F**), property p will always (**G**) hold'.

3.3 Parameter tuning

Using RoVerGene, we looked for parameter modifications that would improve the network behavior. Stated differently, we searched for valid parameters. We have chosen to tune production rate parameters, since recently developed techniques allow to tune promoter or ribosome binding site

[3]In Batt *et al.* (2007a,b), we used the simplest piecewise-affine function (i.e. a ramp function) to describe gene regulations. The use of more general piecewise-affine functions allows us to obtain a more faithful model.

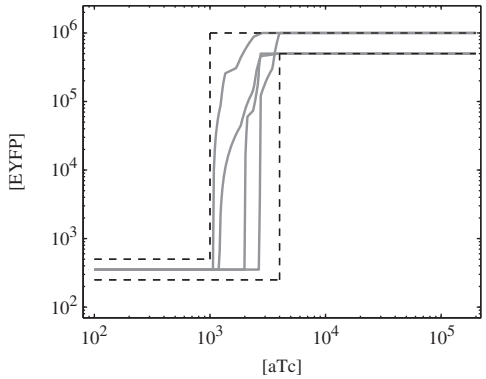


Fig. 8. Steady-state input/output behavior of the cascade for extreme parameter values in the valid parameter sets represented in Figure 7d.

efficiencies with relative ease and precision (Hammer *et al.*, 2006). So, we assumed that the production rate parameters of LacI, CI and EYFP were unconstrained, or more precisely, were ranging in intervals spanning at least two orders of magnitude (their original values are $\kappa_{lacI} = 875$, $\kappa_{cI} = 386$ and $\kappa_{eyfp} = 4048$):

$$\kappa_{lacI} \in [70, 7000], \kappa_{cI} \in [75, 8000] \text{ and } \kappa_{eyfp} \in [30, 30000]$$

Using RoVerGeNe, we identified 15 valid parameter sets (< 5 h, PC, 3.4 GHz processor, 1 GB RAM). The analysis of these sets, represented in the parameter space in Figure 7d, suggests to increase by at least 50% the production rates of LacI and CI, and to approximatively double the production rate of EYFP. In particular, this might be achieved by tuning ribosome-binding sites as done by Basu *et al.* (2004).

In order to illustrate the relevance of the constraints found, we considered extreme parameter values in these sets (i.e. vertices of the rectangular regions in Fig. 7d), and computed the input/output behavior of the network at steady state for these parameters (Fig. 8). This clearly illustrate that relevant constraints on the parameters were found.

We recall that not all valid parameters are guaranteed to be found by our method. So in order to evaluate the capacity of our approach to successfully identify valid parameters, we compared our results with those obtained by a brute-force sampling of the parameter space using numerical simulations. 20 000 different samples were considered. For each sample, we randomly chose parameter values and initial protein concentrations. Given that possible parameter values and initial protein concentrations span several orders of magnitude, we considered uniform distributions in the log-transformed spaces. Then, for four different aTc concentrations (10, 100, 400 and 20 10^4 nM), we simulated the network behavior and considered that a parameter value is valid if the concentration of EYFP at steady state satisfies the constraints depicted in Figure 7a. Note that 'valid' here has not the same meaning than previously, since the validity of a parameter is tested solely for one initial condition and four different aTc concentrations. Out of the 20 000 different parameter values considered, we found that 2.27% were valid (Fig. 7d). This figure should be compared with the fact that the volume of the valid parameter sets found using RoVerGeNe represents 1.8% of the volume of the (log-transformed) parameter space. These figures indicate that using RoVerGeNe we were able to identify a significant subset of the set of all valid parameters.

3.4 Robustness of the tuned network

Before experimentally modifying network parameters as suggested by the previous analysis, it is important to verify that the modified network will robustly behave as expected. So, we let all production and degradation rate parameters range in ±10% (or ±20%) intervals centered at their reference values and tested whether the property is robustly satisfied for these parameter variations. Eleven parameters were thus considered uncertain. For tuned parameters, new references values were chosen in the valid parameter sets found in the previous approach. More precisely, we chose $\kappa_{lacI} = 1600, \kappa_{cI} = 1400$ and $\kappa_{eyfp} = 8100$. Using RoVerGeNe, we proved that the property ϕ_2 holds for every parameter in the ±10% parameter set, and we were not able to prove that the same hold for the ±20% set. This proves that the tuned network presents the desired behavior for modest (±10%) parameter variations, and suggests that it does not do so for large (±20%) parameter variations. We recall that there are two reasons for obtaining negative results with RoVerGeNe: either because the property is false, or because our approach fails to prove the property, due to excessive approximations. In this case, a manual analysis of the output given by RoVerGeNe (or more precisely of the counter-example given by the model checker) revealed that for some parameters (minimal production rates and maximal degradation rate for EYFP) the concentration of EYFP at steady state is below the minimal value allowed by the specifications (5 10^5). So, as suggested by RoVerGeNe, the property is indeed not robustly satisfied for ±20% parameter variations. This analysis again illustrates that relevant constraints on parameters have been identified by our approach.

These results were obtained in less than 1h. Consequently, our approach can be considered as rather efficient, especially given the difficulty of the problem: verifying that a non-trivial dynamical property holds for all initial conditions in a 5-dimensional state space and for all parameters in an 11-dimensional parameter space.

4 CONCLUSION

We have presented a method for the analysis of dynamical properties of genetic regulatory networks with parameter uncertainty. Given a PMA model, an LTL specification of a dynamical property and intervals defining a set of uncertain parameters, the proposed approach deals with two problems. The first one is to *test* whether the property is satisfied for every parameter in the parameter set. The second one is to *find* subsets of the given parameter set such that the property is satisfied for every parameter in these subsets. Both problems are of practical importance in quantitative biology. The first one amounts to assess the robustness of the behavior of a network, in the sense that we show that the system presents a given behavior despite environmental fluctuations or

inaccurate parameter estimation. The second one suggests parameter modifications to tune network behaviors and is particularly important for network design, since most initial attempts at constructing gene networks do not result in a system exhibiting the desired behavior.

The motivation for considering uncertain piecewise-multiaffine differential equation models is twofold. First, they are well adapted to model genetic regulatory networks in the face of incomplete quantitative information. This is of utmost importance for applications in systems and synthetic biology, since precise quantitative information are generally not available. Second, they present interesting mathematical properties that allow the development of efficient, tailored algorithms implementing a combination of techniques for the formal verification of continuous dynamical systems, based on discrete abstraction and model checking. These algorithms are implemented in the publicly-available tool RoVerGeNe, and their practical applicability and biological relevance are demonstrated on the tuning of a synthetic network built in *E.coli*.

To the best of our knowledge, the approach presented here is the first computational approach developed specifically for tuning synthetic gene networks. In a different context, Kuepfer *et al*. (2007) have recently developed an approach based on semidefinite programming for partitioning the parameter space of polynomial differential equation models into so-called feasible and infeasible regions. It should be noted that in this approach, 'feasible' simply refers to the existence of a steady state of the system. In contrast, our approach allows to find parameter sets for which the system presents a particular *behavior*, expressed in the rich language of temporal logics.

Finally, the most promising direction for future applications seems to be the use of the proposed methods to support the modular design of large gene networks (Chin, 2006). In this perspective, the behavior of each module (i.e. subnetwork) could be described by a temporal logic property that holds for sets of parameters, initial conditions and inputs. These properties could then be viewed as *certificates* of the robust behavior of the modules and could be used to support module assemblage on a sound basis. In particular, this would pave the way for the approach advocated by Collins and colleagues (Kobayashi *et al*., 2004) in which biologists use network modules as 'plug-and-play' devices to build complex synthetic systems.

ACKNOWLEDGEMENTS

We would like to thank Ilaria Mogno for sharing her implementation of the algorithm for piecewise-affine regression and the reviewers for their interesting comments. We acknowledge financial support by NSF 0432070.

Conflict of Interest: none declared.

REFERENCES

Alur,R. *et al*. (2000) Discrete abstractions of hybrid systems. *Proc. IEEE*, **88**, 971–984.

Andrianantoandro,E. *et al*. (2006) Synthetic biology: new engineering rules for an emerging discipline. *Mol. Syst. Biol*., **2**, 0028.

Antoniotti,M. *et al*. (2003) Model building and model checking for biochemical processes. *Cell Biochem. Biophys*., **38**, 271–286.

Basu,S. *et al*. (2004) Spatiotemporal control of gene expression with pulse-generating networks. *Proc. Nalt Acad. Sci. USA*, **101**, 6355–6360.

Batt,G. *et al*. (2005) Validation of qualitative models of genetic regulatory networks by model checking: analysis of the nutritional stress response in *E. coli*. *Bioinformatics*, **21** (Suppl.1), i19–i28.

Batt,G. *et al*. (2007a) Model checking genetic regulatory networks with parameter uncertainty. In Bemporad,A. *et al*. (eds). *Hybrid Systems: Computation and Control, HSCC'07, Lecture Notes in Computer Science 4416*. Springer, Berlin, pp. 61–75.

Batt,G. *et al*. (2007b) Model checking liveness properties of genetic regulatory networks. In Grumberg,O. and Huth,M. (eds). *Tools and Algorithms for the Construction and Analysis of Systems, TACAS'07, Lecture Notes in Computer Science 4424*. Springer, Berlin, pp. 323–338.

Belta,C. and Habets,L.C.G.J.M. (2006) Controlling a class of nonlinear systems on rectangles. *Trans. Aut. Control*, **51**, 1749–1759.

Belta,C. *et al*. (2002) Control of multi-affine systems on rectangles with applications to hybrid biomolecular networks. In *IEEE Conference on Decision and Control, CDC'02*.

Bernot,G. *et al*. (2004) Application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic. *J. Theor. Biol*., **229**, 339–347.

Browne,M.C. *et al*. (1988) Characterizing finite Kripke structures in propositional temporal logic. *Theor. Comput. Sci*., **59**, 115–131.

Chin,J.W. (2006) Modular approaches to expanding the functions of living matter. *Nature Chem. Biol*., **2**, 304–311.

Clarke,E.M. *et al*. (1999) *Model Checking*, MIT Press, Cambridge, USA.

de Jong,H. (2002) Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Bio*., **9**, 69–105.

de Jong,H. *et al*. (2004) Qualitative simulation of genetic regulatory networks using piecewise-linear models. *Bull. Math. Biol*., **66**, 301–340.

Emerson,E.A. (1990) Temporal and modal logic. In van Leeuwen,J. (eds). *Handbook of Theoretical Computer Science*. vol. B: Formal Models and Sematics, MIT Press. Cambridge, USA, pp. 995–1072.

Endy,D. (2005) Foundations for engineering biology. *Nature*, **438**, 449–453.

Fages,F. *et al*. (2004) Modelling and querying interaction networks in the biochemical abstract machine biocham. *J. Biol. Phys. Chem*., **4**, 64–73.

Ferrari-Trecate,G. *et al*. (2001) A learning algorithm for piecewise linear regression. In Marinaro,M. and Tagliaferri,R. (eds.) *Neural Nets: WIRN Vietri-01*. Springer, Berlin, pp. 114–119.

Gardner,T.S. *et al*. (2000) Construction of a genetic toggle switch in *E. coli*. *Nature*, **403**, 339–342.

Ghosh,R. and Tomlin,C.J. (2004) Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modelling: Delta-Notch protein signalling. *IEE Proc. Syst. Biol*., **1**, 170–183.

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

Hammer,K. *et al*. (2006) Synthetic promoter libraries — tuning of gene expression. *Trends Biotechnol*, **24**, 53–55.

Hooshangi,S. *et al*. (2005) Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proc. Nalt Acad. Sci. USA*, **102**, 3581–3586.

Kauffman,S.A. (1969) Homeostasis and differentiation in random genetic control networks. *Nature*, **224**, 177–178.

Kobayashi,H. *et al*. (2004) Programmable cells: interfacing natural and engineered gene networks. *Proc. Nalt Acad. Sci. USA*, **101**, 8414–8419.

Kuepfer,L. *et al*. (2007) Efficient classification of complete parameter regions based on semidefinite programming. *BMC Bioinformatics*, **8**, 12.

Lin,J.N. and Unbehauen,R. (1992) Canonical piecewise-linear approximations. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, **39**, 697–699.

McDaniel,R. and Weiss,R. (2005) Advances in synthetic biology: on the path from prototypes to applications. *Curr. Opin. Biotechnol*., **16**, 476–483.

Mestl,T. *et al*. (1995) A mathematical framework for describing and analysing gene regulatory networks. *J. Theor. Biol*., **176**, 291–300.

Szallasi,Z. *et al*. (2006) (eds) *System Modeling in Cellular Biology: From Concepts to Nuts and Bolts*. MIT Press, Cambridge, USA.

Thomas,R. *et al*. (1995) Dynamical behaviour of biological regulatory networks: I. Biological role of feedback loops and practical use of the concept of the loop-characteristic state. *Bull. Math. Biol*., **57**, 247–276.

BIOINFORMATICS

Vol. 25 ISMB 2009, pages i169–i178
doi:10.1093/bioinformatics/btp200

A general computational method for robustness analysis with applications to synthetic gene networks

Aurélien Rizk, Gregory Batt*, François Fages and Sylvain Soliman

INRIA Paris-Rocquencourt, 78153 Le Chesnay Cedex, France

ABSTRACT

Motivation: Robustness is the capacity of a system to maintain a function in the face of perturbations. It is essential for the correct functioning of natural and engineered biological systems. Robustness is generally defined in an *ad hoc*, problem-dependent manner, thus hampering the fruitful development of a theory of biological robustness, recently advocated by Kitano.

Results: In this article, we propose a general definition of robustness that applies to any biological function expressible in temporal logic LTL (linear temporal logic), and to broad model classes and perturbation types. Moreover, we propose a computational approach and an implementation in BIOCHAM 2.8 for the automated estimation of the robustness of a given behavior with respect to a given set of perturbations. The applicability and biological relevance of our approach is demonstrated by testing and improving the robustness of the timed behavior of a synthetic transcriptional cascade that could be used as a biological timer for synthetic biology applications.

Availability: Version 2.8 of BIOCHAM and the transcriptional cascade model are available at http://contraintes.inria.fr/BIOCHAM/

Contact: gregory.batt@inria.fr

1 INTRODUCTION

Robustness can be defined as the capacity of a system to maintain a function in the face of perturbations. Over the years, many studies have demonstrated theoretically and experimentally that robustness is a key property of numerous biological processes, and have proposed mechanisms that promote robustness (e.g. Barkai and Leibler, 1997; Batt *et al.*, 2007; Chaves *et al.*, 2007; Ciliberti *et al.*, 2007; Davidson and Levine, 2008; El-Samad *et al.*, 2005; Gonze *et al.*, 2002; Ingolia, 2004; Shen *et al.*, 2008; Shinar *et al.*, 2007; Stelling *et al.*, 2004b; von Dassow *et al.*, 2000). Robustness is now regarded as one of the fundamental characteristics of biological systems because it allows their correct functioning in presence of molecular noise and environmental fluctuations. Excellent reviews have surveyed the role of biological robustness, and discussed its interesting relations with evolvability of biological systems, modularity of biological networks and the trade-off between robustness and fragility (e.g. Kitano, 2004; Stelling *et al.*, 2004a; Yi *et al.*, 2000). In particular, in the context of synthetic biology, these are key issues to take into account at the design level.

Intuitively, the notion of robustness seems easy to define. One considers (i) a particular system, (ii) a particular function and (iii) a particular set of perturbations, and one assesses how perturbations affect (or not) the given function. However, with the notable exception of Kitano (2007), no general formal definition of robustness has been proposed. The precise definition of robustness is generally highly problem-specific. This makes it difficult to discuss and compare the robustness found in different contexts, or even in similar contexts but computed using different formal definitions of robustness. In Kitano (2007), the mathematical foundations of a theory of biological robustness is proposed, with the aim of providing a unified perspective on robustness.

Although very interesting from a theoretical point of view, Kitano's definition might be too general when applying it to particular problems. Indeed the definition relies on a so-called evaluation function, defined using an unspecified, problem-dependent real-valued performance function. Here, we propose to define the evaluation function using the newly introduced notion of violation degree of temporal logic formulae (Fages and Rizk, 2008). Intuitively, the violation degree reflects the distance between a particular behavior of the perturbed system, given as a numerical timed trace, and the expected reference behavior, expressed by a temporal logic formula. Because (i) temporal logics are expressive languages to formalize temporal behavior of dynamical systems and (ii) the violation degree can be automatically computed, our instantiation of Kitano's definition is both *general* and *computational*. The main contribution of our work is that we propose a computational approach for—and an implementation of—the automatic estimation of the robustness that applies to a broad class of dynamical properties and a large variety of possible perturbations. We simply require that the property describing the expected behavior can be expressed in temporal logic and that the behavior of the system can be represented by a numerical trace (possibly obtained by numerical simulation of deterministic or stochastic models).

A second contribution of this work is that we propose two closely related but different notions of robustness that have been used indiscriminately in publications, namely the *absolute robustness* of a system, representing the average functionality of the system under perturbations, and the *relative robustness* with respect to a given nominal behavior of the system, quantifying the impact of perturbations on the nominal behavior. We believe that distinguishing these two notions will help to clarify the analysis of robustness of biological systems. Undoubtedly, formal definitions are useful for making this distinction.

The applicability and biological relevance of our approach is illustrated on the analysis of the robustness of the timed response of a synthetic transcriptional cascade built in *Escherichia coli*. This system presents a high cell-to-cell variability that prevents using it as a biological timer. We look for parameter modifications that improve the robustness of a 'well-timed' behavior.

The remainder of this article is organized as follows. In the next section, we provide a brief description of the violation degree notion introduced in Fages and Rizk (2008). In Section 3, we present the proposed method for robustness estimation and its implementation in BIOCHAM 2.8. In Section 4, we detail the application of our method

*To whom correspondence should be addressed.

to the analysis of the robustness of the synthetic transcriptional cascade.

2 VIOLATION DEGREE OF TEMPORAL LOGIC PROPERTIES

We first define the Boolean semantics of linear temporal logic (LTL) on timed numerical traces (Section 2.1). Then, we show how using the variable abstraction technique of Section 2.2, we can define a continuous satisfaction degree for temporal logic formulae (Section 2.3) better adapted to a quantitative notion of robustness.

2.1 Temporal logic semantics of numerical traces

In this article, we consider that the behavior of a biological system is described by numerical timed traces. These traces can be obtained either by experimentation on the actual system or by numerical simulation of stochastic or deterministic models. Formally, a *numerical trace* is a finite sequence of tuples describing system's evolution with time: $T=(s_0,s_1,\ldots,s_n)$, with $s_i=(t_i,\mathbf{x}_i,\dot{\mathbf{x}}_i)$, $(t_i)_{i\in[0,n]}$ being a strictly increasing sequence of time points, and $\mathbf{x}_i,\dot{\mathbf{x}}_i\in\mathbb{R}^m$ being vectors of state variable values and of their derivatives at time t_i. In Figure 1a, a hypothetical evolution of a protein concentration is represented. The associated trace is $T=((0,6,1.3),(2,8,0.8),\ldots,(24,5,0))$.

We use LTL to express dynamical properties of biological systems. Temporal logics have been developed to specify the behavior of (usually discrete) dynamical systems (Emerson, 1990). Typical properties include reachability (the system can reach a given state), inevitability (the system will necessarily reach a given state), invariance (a property is always true), response (an event necessarily triggers a specific behavior) and infinite occurrences of events (such as oscillations). Illustrative examples of the expressiveness of temporal logics in systems biology can be found in Antoniotti *et al.* (2003); Batt *et al.* (2005); Bernot *et al.* (2004); Calzone *et al.* (2006) and Chabrier and Fages (2003). LTL formulae are built using atomic propositions and LTL operators.

In our approach, atomic propositions π express real-valued linear constraints on time, protein concentrations and their derivatives. The infinite set of atomic propositions is denoted by Π.

LTL operators include the usual logical operators, such as *negation* ($\neg$), logical *and* ($\wedge$), logical *or* ($\vee$) and *implication* ($\rightarrow$), and specific temporal operators, such as *next* (**X**), *future* (**F**), *globally* (**G**) and *until* (**U**). $\mathbf{X}\phi$, $\mathbf{F}\phi$, $\mathbf{G}\phi$ and $\phi\mathbf{U}\psi$, respectively, mean that a property ϕ holds at the next time, at some future time, holds for all future times or holds continuously until another property ψ holds. These operators can be combined to express complex dynamical properties. For example, the trace T represented in Figure 1a satisfies the formula $\phi_1=\mathbf{F}(\texttt{[A]}>7\wedge\mathbf{F}\,\texttt{[A]}<3)$, expressing that at some time point, protein A concentration exceeds 7 and later goes below 3. Because negations can be pushed to atomic propositions with the usual duality properties of operators, and the set of atomic propositions is closed by negation, we consider without loss of generality only negation-free LTL formulae.

The standard semantics of LTL formulae is generally defined with respect to infinite executions, i.e. infinite traces. Because in our case, the traces are finite, the usual semantics of LTL has to be adapted. Let $T=(s_0,s_1,\ldots,s_n)$ be a finite numerical trace, $\pi\in\Pi$ be an atomic proposition and ϕ, ψ be LTL formulae. Then the semantics of LTL formulae with respect to finite traces is defined inductively as $T\models\phi$ iff $s_0\models\phi$, and

- $s_i\models\pi$ iff $s_i=(t_i,\mathbf{x}_i,\dot{\mathbf{x}}_i)$ satisfies π with the usual semantics,
- $s_i\models\phi\wedge\psi$ iff $s_i\models\phi$ and $s_i\models\psi$,
- $s_i\models\phi\vee\psi$ iff $s_i\models\phi$ or $s_i\models\psi$,
- $s_i\models\mathbf{X}\phi$ iff $i<n$ and $s_{i+1}\models\phi$, or $i=n$ and $s_n\models\phi$,
- $s_i\models\mathbf{F}\phi$ iff $\exists j\in[i,n]$ such that $s_j\models\phi$,
- $s_i\models\mathbf{G}\phi$ iff $\forall j\in[i,n]$, $s_j\models\phi$,
- $s_i\models\phi\mathbf{U}\psi$ iff $\exists j\in[i,n]$ s. t. $s_j\models\psi$ and $\forall k\in[i,j-1]$, $s_k\models\phi$.

Our semantics of LTL coincides with the standard semantics used on finite traces completed by a self-loop on the last state (Fages and Rizk, 2008). This semantics differs from the neutral semantics of Eisner *et al.* (2003) for finite traces only for the next operator, which in their definition is always false on the last state, whereas in our case it enjoys the duality property $\neg\mathbf{X}\phi=\mathbf{X}\neg\phi$ and either $\mathbf{X}\phi$ or $\mathbf{X}\neg\phi$ holds. In practice, the next operator being mainly used to detect local extrema, this difference of interpretation is not significant.

It is worth noticing that when the numerical trace corresponds to a discrete representation of a continuous process, the discrete time semantics that we use may cause that particular events are 'missed' independently of the numerical errors that can be made by the numerical integration method. For example, the formula $\mathbf{F}(\texttt{[A]}\geq 10)$ interpreted on trace T of Figure 1a and expressing that eventually [A] exceeds 10 might be found true or false depending

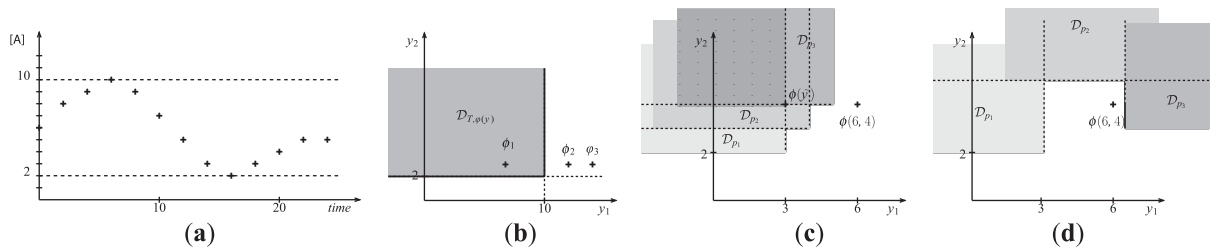


Fig. 1. (**a**) Numerical trace depicting the time evolution of a protein concentration. (**b**) Satisfaction domain $\mathcal{D}_{T,\phi(\mathbf{y})}$ of QFLTL formula $\phi(\mathbf{y})=\mathbf{F}(\texttt{[A]}>y_1\wedge\mathbf{F}\,\texttt{[A]}<y_2)$ and trace T, and LTL formulae ϕ_1, ϕ_2 and ϕ_3, represented in formula space. (**c** and **d**) Representation of satisfaction domains for three perturbations p_1, p_2, and p_3. $\mathcal{D}_{p_i}$ denotes $\mathcal{D}_{T_{p_i},\phi(\mathbf{y})}$. In (c), the intersection of satisfaction domains (shaded) is not empty and $Rsd^s_{\phi,P}=3$. The property $\phi(\tilde{\mathbf{y}})=\phi(3,4)=\mathbf{F}(\texttt{[A]}>3\wedge\mathbf{F}\,\texttt{[A]}<4)$ is satisfied for all perturbations. In (d), the intersection of satisfaction domains is empty and $Rsd^s_{\phi,P}=\infty$.

on the integration step and precision. So care must be taken when checking temporal properties on finite discrete time traces [for a discussion, see Eisner *et al.* (2003); Fainekos and Pappas (2007) and Maler *et al.* (2008), and references therein].

2.2 From model checking to constraint solving

The Boolean interpretation of temporal logic is not well adapted to defining a quantitative notion of robustness. Indeed, neither of the two formulae $\phi_2 = \mathbf{F}(\texttt{[A]} > 12 \wedge \mathbf{F}\ \texttt{[A]} < 3)$ and $\phi_3 = \mathbf{F}(\texttt{[A]} > 14 \wedge \mathbf{F}\ \texttt{[A]} < 3)$ hold for the trace T of Figure 1a. However, intuitively ϕ_2 is closer to satisfaction than ϕ_3, since it only requires that `[A]` reaches 12 instead of 14.

To provide a formal definition of a continuous degree of satisfaction of LTL formulae, we first consider quantifier-free LTL (QFLTL) formulae with free (non-state) real-valued variables $\mathbf{y}$ (Fages and Rizk, 2008). Then, the original model checking problem is transformed into the following *constraint solving problem*: for which values $\mathbf{y}$ does $\phi(\mathbf{y})$ hold on T? Accordingly, we define for any trace T the *satisfaction domain* of $\phi(\mathbf{y})$ as the set of values $\mathbf{y}$ for which $\phi(\mathbf{y})$ holds:

$$\mathcal{D}_{T,\phi(\mathbf{y})} = \{\mathbf{y} \in \mathbb{R}^q \mid T \models \phi(\mathbf{y})\} \tag{1}$$

In the sequel, $\phi(\mathbf{y})$ will denote the QFLTL formula obtained by variable abstraction from a (QF)LTL formula ϕ.

Interestingly, an LTL formula can be seen as an instance of a QFLTL formula obtained by abstracting the constants appearing in the formula by new variables $\mathbf{y} \in \mathbb{R}^q$. For example, to $\phi_1 = \mathbf{F}(\texttt{[A]} > 7 \wedge \mathbf{F}\ \texttt{[A]} < 3)$, we associate the formula $\phi(\mathbf{y}) = \phi(y_1, y_2) = \mathbf{F}(\texttt{[A]} > y_1 \wedge \mathbf{F}\ \texttt{[A]} < y_2)$. Then we have $\phi_1 = \phi(7,3)$. Moreover, one can easily check that for our example trace T, $\mathcal{D}_{T,\phi(y_1,y_2)} = \{y_1 \leq 10 \wedge y_2 \geq 2\}$, 10 and 2 being, respectively, the maximal and minimal values of `[A]` in T.

More generally, this variable abstraction/instantiation process allows us to view a LTL formula as a point in the QFLTL *formula space* $\mathbb{R}^q$, where q is the number of constants appearing in ϕ (or the number of constants that are replaced by variables, if not all constants are abstracted away). In Figure 1b, ϕ_1, ϕ_2, ϕ_3 and $\mathcal{D}_{T,\phi}$ are represented in this formula space.

Given any trace $T = (s_0, s_1, \ldots, s_n)$ and formula $\phi(\mathbf{y})$ we showed in (Fages and Rizk, 2008) that the satisfaction domain $\mathcal{D}_{T,\phi(\mathbf{y})}$ can be computed by induction on T and the subformulae of $\phi(\mathbf{y})$ using the equalities of Proposition 1.

PROPOSITION 1 [Computation of satisfaction domains (Fages and Rizk, 2008)].

- $\mathcal{D}_{T,\phi(\mathbf{y})} = \mathcal{D}_{s_0,\phi(\mathbf{y})}$,
- $\mathcal{D}_{s_i,\pi(\mathbf{y})} = \{\mathbf{y} \in \mathbb{R}^q \mid \pi(\mathbf{y}) \text{ holds with the usual semantics}\}$,
- $\mathcal{D}_{s_i,\phi(\mathbf{y})\wedge\psi(\mathbf{y})} = \mathcal{D}_{s_i,\phi(\mathbf{y})} \cap \mathcal{D}_{s_i,\psi(\mathbf{y})}$,
- $\mathcal{D}_{s_i,\phi(\mathbf{y})\vee\psi(\mathbf{y})} = \mathcal{D}_{s_i,\phi(\mathbf{y})} \cup \mathcal{D}_{s_i,\psi(\mathbf{y})}$,
- $\mathcal{D}_{s_i,\mathbf{X}\phi(\mathbf{y})} = \begin{cases} \mathcal{D}_{s_{i+1},\phi(\mathbf{y})}, & \text{if } i < n, \\ \mathcal{D}_{s_i,\phi(\mathbf{y})}, & \text{if } i = n, \end{cases}$
- $\mathcal{D}_{s_i,\mathbf{F}\phi(\mathbf{y})} = \cup_{j\in[i,n]} \mathcal{D}_{s_j,\phi(\mathbf{y})}$,
- $\mathcal{D}_{s_i,\mathbf{G}\phi(\mathbf{y})} = \cap_{j\in[i,n]} \mathcal{D}_{s_j,\phi(\mathbf{y})}$,
- $\mathcal{D}_{s_i,\phi(\mathbf{y})\mathbf{U}\psi(\mathbf{y})} = \cup_{j\in[i,n]} (\mathcal{D}_{s_j,\psi(\mathbf{y})} \cap \cap_{k\in[i,j-1]} \mathcal{D}_{s_k,\phi(\mathbf{y})})$.

The atomic propositions in $\phi(\mathbf{y})$ being linear constraints on free variables $\mathbf{y}$, the satisfaction domains are finite unions and intersections of polytopes that can be computed with standard polyhedra libraries. Although generally efficient, these operations require in the worst case a time exponential in the formula space dimension. They are, however, independent on the number of state variables.

2.3 Violation degree

To quantify how far from satisfaction a system's behavior is, we introduce the *violation degree* $vd(T,\phi)$ of a formula ϕ with respect to trace T as the *distance* between the actual specification and validity domain $\mathcal{D}_{T,\phi(\mathbf{y})}$ of the QFLTL formula $\phi(\mathbf{y})$ obtained by variable abstraction:

$$vd(T,\phi) = dist(\phi, \mathcal{D}_{T,\phi(\mathbf{y})}).$$

The violation degree has thus a simple interpretation, since it *quantifies by how much a given LTL formula must be changed to hold on a given numerical trace*.

Considering again our example in Figure 1b and using the Euclidean distance, we have that $vd(T,\phi_1) = 0$, meaning that the formula is satisfied by T, and $vd(T,\phi_2) = 2$ and $vd(T,\phi_3) = 4$, reflecting that T is further from satisfaction of ϕ_3 than of ϕ_2.

We would like to emphasize that abstracting constants by variables in temporal logic formulae is a means to define a *metric* on the set of formulae. All set operations and distance computations are made in the corresponding metric space, known as the formula space. It might seem more intuitive to define distances directly between traces. For example, Fainekos and Pappas (2006) use with a similar aim—defining a continuous interpretation of temporal logic formulae on traces—the distance between a given trace T and the set of traces satisfying a formula ϕ. One major advantage of our approach is that the dimensionality of the formula space (number of abstracted constants) is generally much lower than the dimensionality of the trace space (trace length). Performing set operations and distance computation in low-dimensional spaces may strongly affect the practical applicability of these methods. In Donaldson and Gilbert (2008) a similar notion of violation degree has been recently proposed, also based on the definition of a satisfaction domain of temporal logic formulae. However, the computation of the (finite) satisfaction domain is made by sampling the formula space rather than by constraint solving. In this article, we will use the Euclidean distance. However, many other distances can be used (e.g. Manhattan or Chebyshev), depending on the desired interpretation of distance and, as we will see in the next paragraph, on the desired interpretation of robustness.

To define the robustness of a behavior, it is more convenient to reason with a positive notion that describes how good the (possibly perturbed) system performs, i.e. satisfies a dynamical property. To do so, we introduce the notion of continuous *satisfaction degree* of a formula with respect to a trace T:

$$sd(T,\phi) = \frac{1}{1 + vd(T,\phi)} \in [0,1], \tag{2}$$

where vd is the violation degree previously introduced. The satisfaction degree is normalized such that it ranges between 0 and 1, with a satisfaction degree equal to 1 when the property is true and tending toward 0 when the system is far from satisfying the expected property. For some applications, the satisfaction degree

might be normalized differently, using a given constant K instead of the ones in Equation (2).

3 ROBUSTNESS DEFINITIONS AND COMPUTATIONS

3.1 Absolute robustness

In this article, we mainly use Kitano's general definition of robustness. In Kitano (2007), the robustness of a property a of a system s with respect to a set of perturbations P is defined as the *average* of an evaluation function D_a^s of the system over all perturbations $p \in P$, weighted by the perturbation probabilities $prob(p)$:

$$R_{a,P}^s = \int_{p \in P} prob(p) D_a^s \, \mathrm{d}p \tag{3}$$

One should emphasize that this definition is very *general* and can be used in many cases. Unfortunately, Kitano does not provide much information on how to define the so-called *evaluation function* D_a^s of the system. This function should determine if the system still maintains its function under a perturbation and to what degree. The evaluation function needs to be defined for each specific problem in an *ad hoc* manner and re-implemented for the computation of the robustness. A central contribution of this article is to demonstrate that using the notion of satisfaction degree presented previously, one can provide a *general computational framework* based on temporal logic and Kitano's definition that can be used to evaluate the robustness of broad types of dynamical properties and perturbations.

Formally, the robustness of the system is defined as:

$$R_{\phi,P}^s = \int_{p \in P} prob(p) \, sd(T_p, \phi) \, \mathrm{d}p, \tag{4}$$

where ϕ is the specification of the functionality in temporal logic and T_p is the trace representing the behavior of the system under perturbation p. This notion of *robustness* corresponds to a mean functionality, that is, describes on average how the system behaves under perturbations. To illustrate this, consider the plots 1 and 2 of Figure 2 that describe the performance D_a^s—or equivalently in our case, the satisfaction degree—of two hypothetical systems in the face of perturbations p. Because these two plots have the same average, the robustness of these two systems would be equal for evenly distributed perturbations. For example, in a bioengineering context, if the 'property' reflects the quantity of some product exported by cells, these two systems will indeed produce on average the same quantity of the desired product.

This notion of robustness has been used in Ingolia (2004), Ma *et al.* (2006) and von Dassow *et al.* (2000) to study the influence of large parameter variations on the *Drosophila* segment polarity pattern formation. Ma *et al.* (2006) used a Boolean criteria requiring a 'pattern penality function' $pen(x(t))$ to be below a given threshold $\theta^* = 0.0125$ at 600 and 800 min. The QFLTL formula $\phi(\theta) = \mathbf{G}(time \in [600, 800] \rightarrow pen(x(t)) \leq \theta)$ states that the penalty function must be below θ in the entire time interval, and the satisfaction degree of a system's behavior T_p and $\phi(\theta^*)$ provides a quantitative measure of the distance between T_p and the reference behavior.

3.2 Relative robustness

When comparing plots 1 and 2 of Figure 2, it appears that the consequences of perturbations of the nominal behavior are not the same, with T_0 the nominal behavior. In System 1, perturbations degrade the system's performance more severely than in System 2. So, with a different meaning of robustness, one could say that System 2 is more robust than System 1. These two robustness interpretations (as average behavior or as impact of perturbations on nominal behavior) have been indiscriminately used in the literature (see e.g. Davidson *et al.*, 2003; von Dassow *et al.*, 2000). To reflect this second interpretation, let us define the *relative robustness* of a system with respect to a nominal behavior as the system's robustness divided by its nominal performance, that is, by the satisfaction degree of the reference behavior.

$$R_{\phi,P}^{s,p*} = R_{\phi,P}^s / sd(T_{p*}, \phi), \tag{5}$$

where T_{p^*} denotes the unperturbed, nominal behavior of the system. In Figure 2, one can distinguish the relative robustness of Systems 1 and 2 with respect to their nominal performance, reflecting that the performance is more impacted by perturbations in System 1 than in System 2. The performance function of System 3 equals half of the performance function of System 1. Consequently, these systems have the same relative robustness with respect to their nominal performance, although they have different absolute robustness.

Gonze *et al.* (2002) studied the influence of low molecule numbers on circadian oscillation periods. Stochastic simulation results are compared with the behavior of a corresponding deterministic model. The period is defined as the time interval separating two successive upward crossings of the mean value of protein or mRNA concentrations. One can study such oscillations with our approach using, for example, the QFLTL formula $\mathbf{F}(x < m \wedge \mathbf{X}(x > m) \wedge time = t_1 \wedge \mathbf{F}(x < m \wedge \mathbf{X}(x > m) \wedge time = t_2)) \wedge t_2 - t_1 < b$, expressing that the maximal time between two successive upwards crossing events is less than b, with m the mean value of x, that needs to be computed beforehand. More complex temporal behaviors, such as the existence

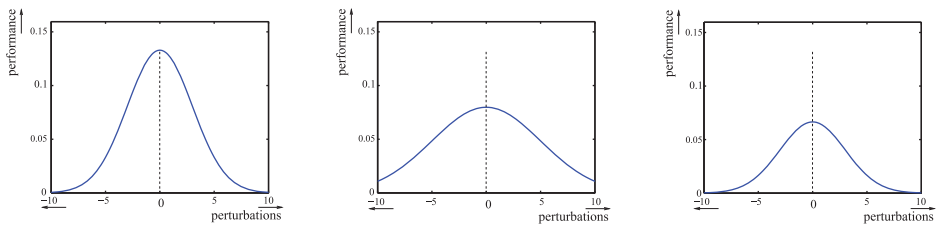


Fig. 2. Systems having same absolute robustness (1 and 2) or same relative robustness (1 and 3), assuming evenly distributed perturbations. Performance functions of Systems 1 and 2 have same the average, whereas the performance function of System 3 is half of the one of System 1.

of 13 mitotic cycles followed by G2 arrest (Calzone *et al.*, 2007), could be expressed similarly and subsequently analyzed using our approach.

3.3 Robust satisfaction degree

Using the notion of satisfaction domain, we can also define the *distance from robust satisfaction* of a property ϕ with respect to a set of perturbations P as $dist(\cap_p \mathcal{D}_{T_p}, \phi)$. This distance reflects the minimal change in the formula such that it holds for *all* perturbations. Then, we define the *robust satisfaction degree* as:

$$Rsd^s_{\phi,P} = \frac{1}{1+dist(\cap_p \mathcal{D}_{T_p,\phi})} \quad (6)$$

This notion allows us to distinguish whether it is possible to relax the specification to have it satisfied for all perturbations or not. In the case of Figure 1c, one can guarantee that the system always presents a (possibly suboptimal) behavior. Moreover, the closest property $\phi(\tilde{\mathbf{y}})$ robustly satisfied (i.e. such that $\tilde{\mathbf{y}} = argmin_{\mathbf{y}\in\cap_p \mathcal{D}_{T_p}} dist(\cap_p \mathcal{D}_{T_p}, \phi)$) can provide interesting hints for the system's design: because $\tilde{\mathbf{y}} = (3,4)$, it suggests that only the first value in ϕ (i.e. the maximum of `[A]` in T) needs to be modified.

In Batt *et al.* (2007), an approach is presented to check that a (model of a) synthetic transcriptional cascade satisfies a given input/output steady state property for sets of parameters. More precisely, it was required for all parameters in a given set, that if the inducer concentration is low ($u_{aTc} < 100$), then at steady state the fluorescence is low ($x_{eyfp} < 500$), and if the inducer concentration is high ($u_{aTc} > 400$), then so is the steady state fluorescence ($x_{eyfp} > 500000$). When considering the QFLTL formula $\phi(m,M) = u_{aTc} < 100 \rightarrow \mathbf{FG}(x_{eyfp} < m) \wedge u_{aTc} > 400 \rightarrow \mathbf{FG}(x_{eyfp} > M)$, it is additionally possible to find the set of *properties* satisfied by all parameters. This can be done by computing the intersection of all satisfaction domains $\cap_p \mathcal{D}_{T_p,\phi}$. The robust satisfaction degree of the property $\phi(m,M)$ provides an indication of how close to robust satisfaction our requirement is.

3.4 Implementation

For the computation of $R^s_{\phi,P}$, $R^{s,p*}_{\phi,P}$ and $Rsd^s_{\phi,P}$, one needs to distinguish whether the set of perturbations is finite (e.g. gene knockouts) or infinite (e.g. normally distributed parameter variations). In the first case, the values can be computed exactly, whereas in the second case, they can be estimated by sampling the perturbation set for sufficiently many perturbations.

The following algorithm is implemented in version 2.8 of the freely available tool BIOCHAM, a modeling environment for the analysis of biological systems (Calzone *et al.*, 2006). Given an ODE model f, a set P of perturbations of initial conditions or parameters, and (QF)LTL properties ϕ and $\phi(\mathbf{y})$, the tool computes the robustness, the relative robustness and the robust satisfaction degree of the property with respect to the given perturbations. The computation of the trace T_p is done by numerical integration. The computation of the satisfaction domain $\mathcal{D}_{T_p,\phi(\mathbf{y})}$ is made by induction on the formula structure, using for each subformula a direct implementation of the definition. Polytopes operations are implemented in BIOCHAM using a standard polyhedral library (Bagnara *et al.*, 2008).

Algorithm 1 Robustness computation

input: a model f, (QF)LTL formulae ϕ and $\phi(\mathbf{y})$, a set of perturbations P and their probabilities, a nominal behavior p^*
output: robustness estimates $R_{\phi,P}$, $R^{p*}_{\phi,P}$, and $Rsd_{\phi,P}$

1: **for** every perturbation $p \in P \cup \{p^*\}$ **do**
2: $T_p :=$ COMPUTE_TRACE(f,p)
3: $\mathcal{D}_{T_p,\phi(\mathbf{y})} :=$ COMPUTE_SAT_DOMAIN$(T_p, \phi(\mathbf{y}))$
4: **end for**
5: $R_{\phi,P} := \sum_{p\in P} prob(p)(1+dist(\mathcal{D}_{T_p,\phi(\mathbf{y})}, \phi))^{-1}$
6: $R^{p*}_{\phi,P} := R_{\phi,P} * (1+dist(\mathcal{D}_{T_{p^*},\phi(\mathbf{y})}, \phi))$
7: $Rsd_{\phi,P} := (1+dist(\cap_{p\in P} \mathcal{D}_{T_p,\phi(\mathbf{y})}, \phi)))^{-1}$

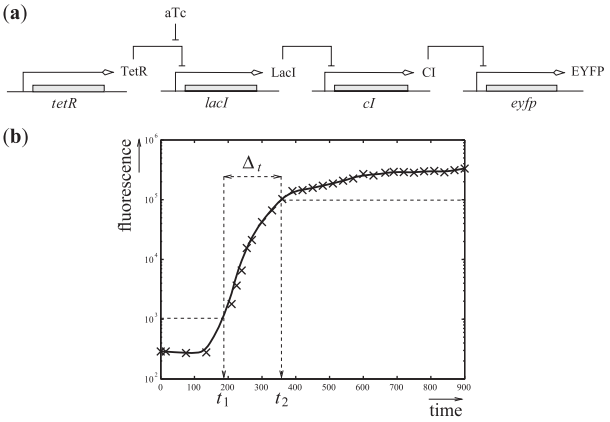


Fig. 3. Synthetic transcriptional cascade. (**a**) TetR represses *lacI*, LacI represses *cI* and CI represses *eyfp*. aTc controls the repression of *lacI* by TetR. The fluorescence of the protein EYFP is the output. (**b**) Graphical representation of a 'well-timed' behavior: fluorescence remains below 10^3 until time t_1, exceeds 10^5 after time t_2 and switches between low and high levels in Δ_t time. One expects that $t_1 > 150$, $t_2 < 450$ and $\Delta_t < 150$. Crosses represent experimental data from Hooshangi *et al.* (2005).

4 APPLICATION TO ROBUSTNESS ANALYSIS OF A TRANSCRIPTIONAL CASCADE

We consider the design of a synthetic transcriptional cascade that could be used for the temporal sequencing of events in synthetic biology applications. This cascade has been build by Hooshangi and colleagues (2005) and here we investigate the robustness of a desired behavior, and the possibilities to make it more robust. To do so, after having introduced the system, we formalize the expected behavior, develop a model of the system taking into account the observed variability and apply the method presented previously to investigate the robustness of the desired property.

4.1 System description

We consider a cascade of transcriptional inhibitions built in *E. coli* (Hooshangi *et al.*, 2005). The network is represented in Figure 3a. It is made of four genes: *tetR*, *lacI*, *cI* and *eyfp* that code, respectively, for three repressor proteins, TetR, LacI and CI, and the fluorescent protein EYFP. The fluorescence of the system, due to the protein EYFP, is the measured output. The system can be controlled by the addition or removal of a small diffusible molecule, aTc, in the growth media. More precisely, aTc binds to TetR and relieves the repression of *lacI*. The aTc concentration thus serves as a controllable input to the system. It is intuitively clear

$$\dot{x}_{lacI} = \kappa^0_{lacI} + \kappa_{lacI}\frac{u_{aTc}^{\eta_{aTc}}}{\theta_{aTc}^{\eta_{aTc}} + u_{aTc}^{\eta_{aTc}}} - \gamma_{lacI}\, x_{lacI},$$
$$\dot{x}_{cI} = \kappa^0_{cI} + \kappa_{cI}\frac{\theta_{lacI}^{\eta_{lacI}}}{\theta_{lacI}^{\eta_{lacI}} + x_{lacI}^{\eta_{lacI}}} - \gamma_{cI}\, x_{cI},$$
$$\dot{x}_{eyfp} = \kappa^0_{eyfp} + \kappa_{eyfp}\frac{\theta_{cI}^{\eta_{cI}}}{\theta_{cI}^{\eta_{cI}} + x_{cI}^{\eta_{cI}}} - \gamma_{eyfp}\, x_{eyfp},$$
$$\dot{u}_{aTc} = 0.$$

(a)

κ^0_{lacI}	3.0	θ_{aTc}	1100
κ^0_{cI}	5.1	θ_{lacI}	380
κ^0_{eyfp}	3.2	θ_{cI}	1000
κ_{lacI}	1150	η_{aTc}	3.0
κ_{cI}	538	η_{lacI}	2.1
κ_{eyfp}	3617	η_{cI}	4.0
γ_{lacI}	0.017		
γ_{cI}	0.015		
γ_{eyfp}	0.012		

(b)

$$\boldsymbol{P} \hookrightarrow LogN(ln(\boldsymbol{p}^*) - ln(1+\boldsymbol{\sigma}^2)/2, \sqrt{ln(1+\boldsymbol{\sigma}^2)}),$$
$$\text{with } \boldsymbol{\sigma} = \begin{cases} 0.064 & \text{for } \boldsymbol{\gamma}, \boldsymbol{\theta}, \boldsymbol{\eta}, \\ 0.128 & \text{for } \boldsymbol{\kappa}^0, \\ 0.256 & \text{for } \boldsymbol{\kappa} \end{cases}$$
$$\text{and such that } E(\boldsymbol{P}) = \boldsymbol{p}^* \text{ and } \sigma(\boldsymbol{P}) = \boldsymbol{\sigma}\boldsymbol{p}^*$$

(c)

Fig. 4. (**a**) ODE model of the transcriptional cascade. The concentrations of protein LacI, CI, EYFP and of aTc are denoted by x_{lacI}, x_{cI}, x_{eyfp} and u_{aTc}, respectively. The concentration of the constitutively expressed protein TetR is assumed constant. (**b**) Reference parameter values $\boldsymbol{p}^*$ and (**c**) parameter distributions modeling system's variability. $\boldsymbol{\sigma}$ is a noise intensity parameter vector.

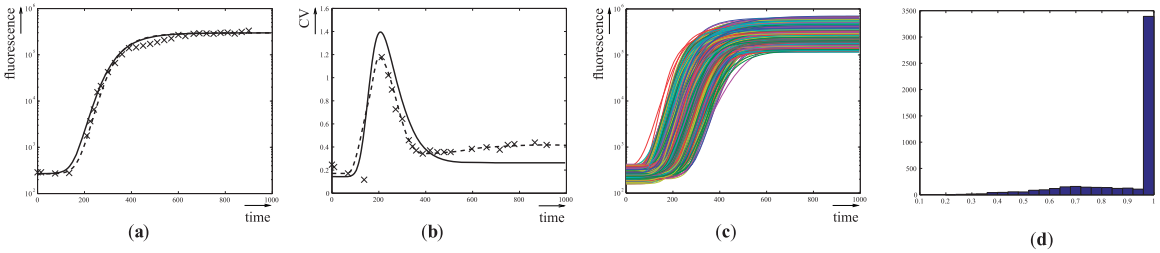


Fig. 5. (**a**) Temporal evolution of the fluorescence following addition of aTc. Crosses, dotted line and solid line represent experimental data from Hooshangi *et al.* (2005), predictions obtained using the ODE model with reference parameters $\boldsymbol{p}^*$, and average of 5000 numerical simulations of the ODE model with log-normally distributed parameters, respectively. (**b**) Temporal evolution of the coefficient of variation of the fluorescence following addition of aTc. Crosses and solid line represent coefficient of variations obtained from experimental data in Hooshangi *et al.* (2005) and from 5000 numerical simulations of the ODE model with log-normally distributed parameters with mean $\boldsymbol{p}^*$, respectively. (**c**) Numerical simulations of the ODE model with log-normally distributed parameters with mean $\boldsymbol{p}^*$. (**d**) Distribution of satisfaction degrees for 5000 numerical traces of the perturbed transcriptional cascade model. The corresponding robustness is $\hat{R}_{\phi,P} = 0.9$.

that the output (i.e. the fluorescence) of the system at steady state will be low for low inputs (i.e. aTc concentration), and high for high inputs. Moreover, it has been shown that the time response of the system to an inducer addition is characterized by a rapid increase of the fluorescence, preceded by a significant lag phase. Unfortunately, a high cell-to-cell variability has also been observed. The heterogeneity of the cell responses makes it difficult to use this system as a biological timer, for example for developmental programs as suggested in Hooshangi *et al.* (2005). In this context, as for many synthetic biology applications, having even a low proportion of cells sending a signal too early or too long might compromise the correct functioning of the whole system. Our goal here is to precisely investigate the possibilities to obtain a robustly 'well-timed' system, that is to ensure that all cells will indeed change state in a given time window.

4.2 Specifying the expected behavior

Here, we consider that the system is well-timed if the fluorescence remains below 10^3 for at least 150 min, then exceeds 10^5 after at most 450 min, and switches rapidly from low to high levels, that is, in less than 150 min. These specifications are consistent with the experimentally observed behavior of the cell population. These specifications are graphically represented in Figure 3b and can be formalized in temporal logic as follows:

$$\begin{aligned}\phi(t_1,t_2) = \quad & \mathbf{G}(time < t_1 \rightarrow [\texttt{EYFP}] < 10^3) \\ \wedge\ & \mathbf{G}(time > t_2 \rightarrow [\texttt{EYFP}] > 10^5) \\ \wedge\ & t_1 > 150 \wedge t_2 < 450 \wedge t_2 - t_1 < 150\end{aligned}$$

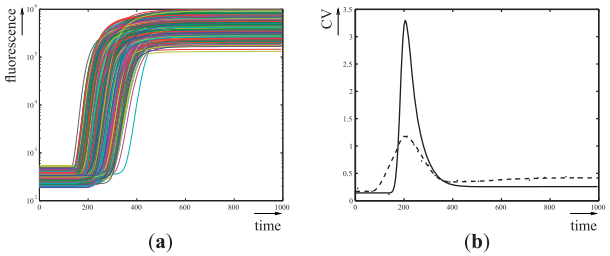
which is abstracted into

$$\begin{aligned}\phi(t_1,t_2,b_1,b_2,b_3) = \quad & \mathbf{G}(time < t_1 \rightarrow [\texttt{EYFP}] < 10^3) \\ \wedge\ & \mathbf{G}(time > t_2 \rightarrow [\texttt{EYFP}] > 10^5) \\ \wedge\ & t_1 > b1 \wedge t_2 < b_2 \wedge t_2 - t_1 < b_3\end{aligned}$$

for the computation of validity domains and satisfaction degree in a given trace.

4.3 Modeling the system's variability

There are many ways to model cell variability (see for example Manninen *et al.*, 2006). Our goal here is to construct a simple model such that the predicted behavior and standard deviation are in agreement with the available experimental data. We first develop a simple ODE model similar to Batt *et al.* (2007) but using Hill functions, with parameters fitted to experimental data (Figures 4a and b). These parameter reference values are denoted by $\boldsymbol{p}^*$ in the sequel. Second, we consider various ways to model cell variability, including stochastic differential equations with either additive or multiplicative noise and random parameter variations with (log)normal distributions. We have obtained a good qualitative and quantitative agreement between the predicted and observed mean and standard deviation for log-normally distributed parameters, as shown in Figure 5a and b. So we selected these log-normal parameter distributions as our 'perturbation model'. Using either stochastic differential equations or normally distributed parameters, we have not been able to find an agreement between model predictions and experimental observations (data not shown). This could be partially



(**a**) (**b**)

Fig. 6. Numerical simulations of the ODE model with 5000 log-normally distributed parameters with mean $\tilde{\boldsymbol{p}}$. (**a**) Temporal evolution of the fluorescence following addition of aTc. (**b**) Temporal evolution of the coefficient of variation of the fluorescence. Crosses and solid line represent coefficient of variations obtained from experimental data in Hooshangi *et al.* (2005) and from numerical simulations, respectively.

explained by the very high cell-to-cell variability. In particular, the observed coefficient of variation reaches 1.4 at some time point, meaning that the standard deviation is higher than the mean.

4.4 Improving robustness of the desired behavior

Having specified the 'well-timed' behavior and found an ODE model and a perturbation model, we wondered whether the system is robustly well-timed, and to what degree. When considering 5000 log-normally distributed parameter values in the 16 D parameter space, we estimated the robustness of the system as $\hat{R}_{\phi,P}=0.9$: the specification is not robustly satisfied. As expected, the property holds for the reference parameter values $\boldsymbol{p}^*$ (i.e. $sd(T_{\boldsymbol{p}^*},\phi)=1$), and consequently the robustness and absolute robustness are equal ($\hat{R}^{\boldsymbol{p}^*}_{\phi,P}=0.9$). The distribution of the satisfaction degree is represented in Figure 5d, showing that although the majority of timed traces satisfies the specification, this is not always the case. On average, each numerical simulation lasts 150 ms, and each satisfaction degree computation lasts 50 ms (~500 time points/trace; Dual Core, 2 GHz, 2 GB RAM). For this application and in all our computations, the limiting factor is numerical simulation.

As said earlier, for most synthetic biology applications, a more robust timer would be needed. Can we find other parameters so as to improve the robustness of the system with respect to similar parameter perturbations? To do so, we use the state-of the-art non-linear optimization tool CMAES that uses a covariance matrix adaptation evolution strategy (Hansen and Ostermeier, 2001), with the *robustness as optimization criteria* (i.e. as fitness function). We found the following parameter values: $\tilde{\boldsymbol{p}}=(\kappa^0,\kappa,\gamma,\theta,\eta)$, with

$$\tilde{\boldsymbol{p}}=((2.30,4.20,3.78),(1234.5,514.5,5174.3),(0.024,\\ 0.015,0.012),(1647.2,662.8,936.4),(4.8,3.7,8.4))$$

The comparison between original parameters $\boldsymbol{p}^*$ and so-called optimized parameters $\tilde{\boldsymbol{p}}$ reveals that the EYFP production rate and the Hill coefficients η have been significantly increased. Given that one wants to ensure a fast transition between the low and high states, these parameters were obvious targets for optimizations. Because tuning Hill coefficients is experimentally difficult, we looked for and found parameters with unchanged Hill coefficients that ensure a robust well-timed behavior.

Numerical integrations illustrate that the expected behavior is indeed more robustly obtained (compare Figures 5c and 6a). Interestingly, the coefficient of variation suggests that cell-to-cell variability will be significantly decreased when the time constraints hold (for $time<150$) and is significantly increased otherwise (for $150<time<450$, see Figure 6b). It would be interesting to study whether this feature appears systematically for parameter variations improving the robustness of the desired behavior. This could reveal trade-offs between robustness and fragility (Kitano, 2004).

4.5 Parameter influence on robust behavior

To obtain a more comprehensive picture of the variations of the robustness of the expected behavior, we sample the parameter space for large parameter variations, and for each parameter, we compute the robustness.

More precisely, we consider grids on the parameter space centered on the reference parameter values $\boldsymbol{p}^*$ and corresponding to $\pm$10-fold parameter variations of either two parameters (κ_{eyfp} and γ; 2D grids) or eight parameters ($\boldsymbol{\kappa^0}$, $\boldsymbol{\kappa}$, γ, and u_{aTc}; 8D grids). Then, for each grid point—taken as reference value for relative robustness computations—we estimate the robustness of the network behavior when all 16 parameters vary. Note that we consider the initial aTc concentration as a parameter. The γ parameter corresponds to a scaling factor of all degradation parameters γ_{lacI}, γ_{cI} and γ_{eyfp}, with $\gamma^*=1$. It is used to assess the impact of growth rate variations, affecting similarly all protein dilution rates, and consequently, all degradation rates. Robustness is estimated based on 50 perturbations (i.e. parameters), or less in case of fast convergence.

For the 2D grid, results can be visually displayed. In Figure 7, the satisfaction degree, the robustness and the relative robustness are represented in the (κ_{eyfp},γ) parameter space. It appears that the constraints on γ are much tighter than the constraints on κ_{eyfp}. Both for the satisfaction degree and for the robustness, γ has to remain in a narrow interval, whereas κ_{eyfp} simply has to exceed some value. This result can be explained by the fact that high production rates of the fluorescent protein helps the system to have a fast and marked response, whereas variations in protein degradation rates γ have subtle effects on the behavior, since it lowers the concentration of the fluorescent protein and of its repressor. It seems that the nominal behavior, and even more the average behavior, is rather fragile to growth rate variations.

The robustness landscape appears like a blurred version of the satisfaction degree landscape. This corresponds to the fact that parameter variations corresponding to cell-to-cell variability used for computing the robustness are generally smaller than parameter perturbations considered when exploring the parameter space. However, one should also stress that the robustness takes into account parameter variations in all dimensions and with particular distributions (here log-normal, with various noise intensities σ). Thus, Figure 7b is not merely a blurred version of Figure 7a.

In Figure 7c, it appears that the relative robustness that quantifies how different the average behavior is from the nominal one, efficiently identifies regions where the satisfaction degree changes significantly. In the context of system design, this information is of great interest. This could be compared with the sensitivity of satisfaction degree with respect to parameter perturbations. However, contrary to the sensitivities, the relative robustness takes into account a given perturbation model.

The preceding analysis is naturally not possible when considering parameter variations in higher dimensions. To carry the analysis on

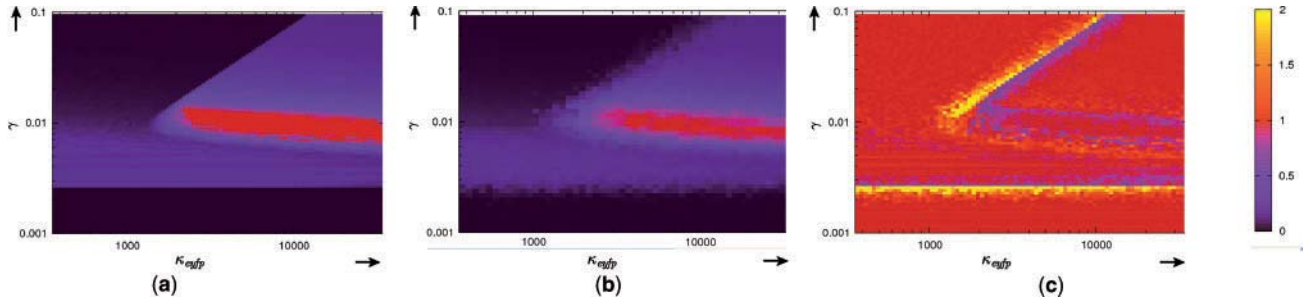


Fig. 7. (**a**) Satisfaction degree, (**b**) robustness and (**c**) relative robustness represented in the (κ_{eyfp}, γ) parameter space.

Table 1. First and most significant second order sensitivity indices defined for the robustness with respect to large parameter variations and computed on 8D grids.

First order sensitivity indices		Second order sensitivity indices	
S_{γ}	20.2%	$S_{\kappa_{eyfp},\gamma}$	8.7%
$S_{\kappa_{eyfp}}$	7.4%	$S_{\kappa_{cI},\gamma}$	6.2%
$S_{\kappa_{cI}}$	6.1%	$S_{\kappa^0_{cI},\gamma}$	5.0%
$S_{\kappa^0_{lacI}}$	3.3%	$S_{\kappa^0_{cI},\kappa_{eyfp}}$	2.8%
$S_{\kappa^0_{cI}}$	2.0%	$S_{\kappa_{cI},\kappa_{eyfp}}$	1.8%
$S_{\kappa_{lacI}}$	1.5%	$S_{\kappa^0_{eyfp},\gamma}$	1.5%
$S_{\kappa^0_{eyfp}}$	0.9%	$S_{\kappa^0_{cI},\kappa_{cI}}$	1.1%
$S_{u_{aTc}}$	0.4%	$S_{\kappa^0_{cI},\kappa_{lacI}}$	0.5%
Total first order	40.7 %	Total second order	31.2 %

8D grids, we use a variance-based global sensitivity method (Saltelli *et al.*, 2004). When a measure (in our case the robustness) is affected by variations of several parameters, one can statistically assess the importance of the variations of each parameter by computing its *sensitivity index*:

$$S_i = \frac{Var(E(R|P_i))}{Var(R)} \in [0,1]$$

These sensitivity indices and higher order sensitivity indices quantify how the variations of a parameter P_i or a group of parameters contribute to the variance of R.

We consider 8D grids defined as follows. Each grid is defined by three parameter values (p_i^1, p_i^2 and p_i^3) in each dimension. These values—or more precisely their log—are obtained by dividing evenly the parameter domain $[\ln(p_i/10), \ln(10p_i)]$ in three subintervals and by choosing randomly a value in each subinterval. The first and most significant second-order sensitivity indices are given in Table 1. They correspond to average values obtained on three similarly defined grids (∼20 hr per 8D grid).

The analysis of the first-order sensitivity indices corroborates our previous finding that γ variations have a very strong impact on the robust behavior of the cascade. The variations of this parameter alone are responsible for 20% of the robustness variations. In contrast, aTc variations seem to have a very low impact on the cascade behavior. Although it might seem in contradiction with the ultrasensitivity of the input/output behavior (Hooshangi *et al.*, 2005), it simply indicates that the aTc concentrations used for inducing the cascade are high enough to make the network insensitive to even large aTc variations.

A surprizing outcome of this analysis is the very different importance of variation in the basal and regulated EYFP production rates, κ^0_{eyfp} and κ_{eyfp} (Table 1). Given that the specification imposes similar constraints on the 'low' and 'high' EYFP levels, and that these levels are under mild approximations proportional to the ratio $\kappa^0_{eyfp}/\gamma_{eyfp}$ and $\kappa_{eyfp}/\gamma_{eyfp}$, respectively, one could have expected similar sensitivity indices for κ^0_{eyfp} and κ_{eyfp}. In fact the low EYFP levels also depend—and in a non-linear way—on the steady state value of CI, itself proportional to κ_{cI}/γ_{cI}. Because κ_{cI} variations have strong effects on robust behavior of the cascade, our results suggest that when uninduced, the basal production of EYFP is due to an incomplete repression of the promoter by CI, explaining the high effect of κ_{cI} variations, rather than a constitutive leakage of the promoter, explaining the low effect of κ^0_{eyfp} variations. This hypothesis is also consistent with the second-order sensitivity indices we found: $S_{\kappa_{cI},\gamma} > S_{\kappa^0_{eyfp},\gamma}$.

The analysis of second-order sensitivity indices indicates that joint variations of production and degradation rates play a significant role in robustness variations. This comes with no surprise, since as said earlier, the ratios κ_i/γ_i largely determine the steady state levels of the proteins.

5 DISCUSSION

We have presented a general and computational framework for the definition of the robustness of biological functions with respect to a set of perturbations. This framework is general because it applies (i) to any biological function expressible in the temporal logic LTL, an expressive language for specifying dynamical behaviors widely used in computer science and engineering, and (ii) to any perturbation set, provided that the behaviors of the perturbed system can be obtained as numerical timed traces, for example by numerical integration of ODEs. In this setting, the computation of robustness is fully automated and is implemented in the free software BIOCHAM (Calzone *et al.*, 2006). When formalizing the robustness notion, we found that several definitions can be proposed. One can notably distinguish absolute robustness, quantifying the average performance of a perturbed system, from relative

robustness, quantifying performance degradation/improvement due to perturbations.

To illustrate the applicability of our approach and demonstrate its biological relevance, we considered the possibility to improve the robustness of the timed response of a transcriptional cascade to an addition of inducer. The significant cell-to-cell variability makes it difficult to use this system as a reliable biological timer for synthetic biology applications. We found parameter modifications for which a desired timed behavior is robustly obtained. Moreover, we explored the impact of possibly large parameter variations on the robustness of the desired behavior. Using global sensitivity analysis, we obtained several interesting results that could potentially help for the optimization of the system.

Central to our approach is the notion of *satisfaction degree* of temporal logic formulae. In systems and synthetic biology, many computational approaches use a rather simple measure of the performance of the system, either for parameter searching, robustness computation or local and global sensitivity analysis. Finding a relevant measure of the system performance limits the applicability of the above-mentioned approaches. Examples of such measures are the gain of a response, and the perturbation of a steady state or of the period of oscillations (Felix and Wagner, 2008; Feng *et al.*, 2004; Gonze *et al.*, 2002). In contrast, using the satisfaction degree as a performance measure allows us to take advantage of the expressivity of temporal logics and consequently to significantly broaden the applicability of these techniques. In Rizk *et al.* (2008), we showed that using the satisfaction degree, one can efficiently find parameter values for which complex dynamical behaviors are observed. In this article, we show how using the same notion, one can define and estimate the robustness of any dynamical behavior expressible in LTL with respect to a set of perturbations, and how one can apply global sensitivity analysis to find the effect of parameter variations on the robustness of any LTL specification of an expected behavior. Other approaches have been proposed that use temporal logic to define robustness of biological systems (Batt *et al.*, 2007; Shen *et al.*, 2008). However, these approaches use a Boolean interpretation of temporal logic that is not well-adapted to defining a quantitative notion of robustness.

The relations between robustness and evolvability, and between robustness and modularity have been extensively studied in systems biology (Ciliberti *et al.*, 2007; Kitano, 2004). In synthetic biology, however, not much work has focused on robustness analysis. For obvious reasons, achieving a robust behavior despite cell variability and environmental fluctuations is a central issue in synthetic biology. Because large synthetic networks are very likely to be modular (Chin, 2006; McDaniel and Weiss, 2005), one could envision an approach in which each module is designed to robustly present a given behavior such that one has some guarantee that when included in a more complex system the module still functions as expected. In this context, input/output robustness (Shinar *et al.*, 2007) and insulation (Vecchio *et al.*, 2008) are notions of particular interest.

ACKNOWLEDGEMENTS

We thank Ron Weiss, Priscilla Purnick and other members of the Weiss lab for fruitful discussions and for providing experimental data. We acknowledge partial support from the European project Tempo, the INRIA Colage and INRA AgroBi projects.

Conflict of Interest: none declared.

REFERENCES

Antoniotti,M. *et al*. (2003) Model building and model checking for biochemical processes. *Cell Biochem. Biophys.*, **38**, 271–286.

Bagnara,R. *et al*. (2008) The Parma Polyhedra Library: toward a complete set of numerical abstractions for the analysis and verification of hardware and software systems. *Sci. Comput. Program.*, **72**, 3–21.

Barkai,N. and Leibler,S. (1997) Robustness in simple biochemical networks. *Nature*, **387**, 913–916.

Batt,G. *et al*. (2005) 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.

Batt,G. *et al*. (2007) Robustness analysis and tuning of synthetic gene networks. *Bioinformatics*, **23**, 2415–2422.

Bernot,G. *et al*. (2004) Application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic. *J. Theor. Biol.*, **229**, 339–347.

Calzone,L. *et al*. (2006) BIOCHAM: an environment for modeling biological systems and formalizing experimental knowledge. *Bioinformatics*, **22**, 1805–1807.

Calzone,L. *et al*. (2007) Dynamical modeling of syncytial mitotic cycles in *drosophila* embryos. *Mol. Syst. Biol.*, **3**, 131.

Chabrier,N. and Fages,F. (2003) Symbolic model checking of biochemical networks. In C.Priami, (ed.) *Proceedings of Computational Methods in Systems Biology (CMSB'03)*. Vol. 2602 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 149–162.

Chaves,M. *et al*. (2007) Geometry and topology of parameter space: investigating measures of robustness in regulatory networks. *J. Math. Biol.*, **104**, 13591–13596.

Chin,J.W. (2006) Modular approaches to expanding the functions of living matter. *Nat. Chem. Biol.*, **2**, 304–311.

Ciliberti,S. *et al*. (2007) Innovation and robustness in complex regulatory gene networks. *Proc. Natl Acad. Sci. USA*, **104**, 13591–13596.

Davidson,E.H. and Levine,M.S. (2008) Properties of developmental gene regulatory networks. *Proc. Natl Acad. Sci. USA*, **105**, 20063–20066.

Davidson,E.H. *et al*. (2003) Reglatory gene networks and the properties of the developmental process. *Proc. Natl Acad. Sci. USA*, **100**, 1475–1480.

Donaldson,R. and Gilbert,D. (2008) A model checking approach to the parameter estimation of biochemical pathways. In Heiner,M. and Uhrmacher,A. (eds) *Proceedings of the Fourth International Conference on Computational Methods in Systems Biology (CMSB'08)*. Vol. 5307 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 269–287.

Eisner,C. *et al*. (2003) Reasoning with temporal logic on truncated paths. In *Proceedings of Fifteenth Computer-Aided Verification conference (CAV'O3)*. Vol. 2725 of *Lecture Notes in Computer Science*, Springer, pp. 27–39.

El-Samad,H. *et al*. (2005) Surviving heat shock: Control strategies for robustness and performance. *Proc. Natl Acad. Sci. USA*, **102**, 2736–2741.

Emerson,E.A. (1990) Temporal and modal logic. In van Leeuwen,J. ed. *Handbook of Theoretical Computer Science*. Vol. B: Formal Models and Sematics, MIT Press, Cambridge, pp. 995–1072.

Fages,F. and Rizk,A. (2008) On temporal logic constraint solving for the analysis of numerical data time series. *Theor. Comput. Sci.*, **408**, 55–65.

Fainekos,G.E. and Pappas,G.J. (2006) Robustness of temporal logic specifications. In Havelund,K. *et al*. (eds) *Proceedings of International the Workshop on Formal Approaches to Software Testing and Runtime Verification, (FATES/RV'06)*. Vol. 4262 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 178–192.

Fainekos,G.E. and Pappas,G.J. (2007) Robust sampling for MITL specifications. In Raskin,J.-F. and Thiagarajan,P. (eds) *Proceedings of the Fifth International Conference on Formal Modeling and Analysis of Timed Systems, (FORMATS'07)*. Vol. 4763 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 147–162.

Felix,M.-A. and Wagner,A. (2008) Robustness and evolution: concepts, insights and challenges from a developmental model system. *Heredity*, **100**, 132–140.

Feng,X.-J. *et al*. (2004) Optimizing genetic circuits by global sensitivity analysis. *Biophys. J.*, **87**, 2195–2202.

Gonze,D. *et al*. (2002) Robustness of circadian rhythms with respect to molecular noise. *Proc. Natl Acad. Sci. USA*, **99**, 673–678.

Hansen,N. and Ostermeier,A. (2001) Completely derandomized self-adaptation in evolution strategies. *Evol. Comput.*, **9**, 159–195.

Hooshangi,S. *et al*. (2005) Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proc. Natl Acad. Sci. USA*, **102**, 3581–3586.

Ingolia,N.T. (2004) Topology and robustness in the drosophila segment polarity network. *PLoS Biol.*, **2**, e123.

Kitano,H. (2004) Biological robustness. *Nat. Rev. Genet.*, **11**, 826–837.

Kitano,H. (2007) Towards a theory of biological robustness. *Mol. Syst. Biol.*, **3**, 137.

Ma,W. *et al*. (2006) Robustness and modular design of the drosophila segment polarity network. *Mol. Syst. Biol.*, **2**, 70.

Maler,O. *et al*. (2008) Checking temporal properties of discrete, timed and continuous behaviors. In Avron,A. *et al*. (eds) *Pillars of Computer Science, Essays Dedicated to B. Trakhtenbrot*. Vol. 4800 of *Lecture Notes in Computer Science*, Springer, pp. 475–505.

Manninen,T. *et al*. (2006) Developing Itô stochastic differential equation models for neuronal signal transduction pathways. *Comp. Biol. Chem.*, **30**, 280–291.

McDaniel,R. and Weiss,R. (2005) Advances in synthetic biology: on the path from prototypes to applications. *Curr. Opin. Biotechnol.*, **16**, 476–483.

Nickovic,D. and Maler,O. (2007) AMT: a property-based monitoring tool for analog systems. In Raskin,J.-F. and Thiagarajan,P.S. (eds) *Proceedings of conference on Formal Modeling and Analysis of Timed Systems (FORMAT'07)*. Vol. 4763 of *Lecture Notes in Computer Sciences*, Springer, Berlin, pp. 304–319.

Rizk,A. *et al*. (2008) On a continuous degree of satisfaction of temporal logic formulae with applications to systems biology. In Heiner,M. and Uhrmacher,A. (eds) *Proceedings of the Fourth International Conference on Computational Methods in Systems Biology (CMSB'08)*. Vol. 5307 of *Lecture Notes in Computer Science*, Springer, Berlin, pp. 251–268.

Saltelli,A. *et al*. (2004) *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*. Wiley Press, New York.

Shen,X. *et al*. (2008) Architecture and inherent robustness of a bacterial cell-cycle control system. *Proc. Natl Acad. Sci. USA*, **105**, 11340–11345.

Shinar,G. *et al*. (2007) Input-output robustness in simple bacterial signaling systems. *Proc. Natl Acad. Sci. USA*, **104**, 19931–19935.

Stelling,J. *et al*. (2004a) Robustness of cellular functions. *Cell*, **118**, 675–685.

Stelling,J. *et al*. (2004b) Robustness properties of circadian clock architectures. *Proc. Natl Acad. Sci. USA*, **101**, 13210–13215.

Vecchio,D.D. *et al*. (2008) Modular cell biology: retroactivity and insulation. *Mol. Syst. Biol.*, **4**, 161.

von Dassow,G. *et al*. (2000) The segment polarity network is a robust developmental module. *Nature*, **406**, 188–192.

Yi,T. *et al*. (2000) Robust perfect adaptation in bacterial chemotaxis through integral feedback control. *Proc. Natl Acad. Sci. USA*, **97**, 4649–4653.

OPEN ACCESS Freely available online

PLOS | COMPUTATIONAL BIOLOGY

STL-based Analysis of TRAIL-induced Apoptosis Challenges the Notion of Type I/Type II Cell Line Classification

Szymon Stoma[1☯], Alexandre Donzé[2☯¤], François Bertaux[1], Oded Maler[2], Gregory Batt[1]*

1 INRIA Paris-Rocquencourt, Le Chesnay, France, **2** VERIMAG, CNRS and the University of Grenoble, Gières, France

Abstract

Extrinsic apoptosis is a programmed cell death triggered by external ligands, such as the TNF-related apoptosis inducing ligand (TRAIL). Depending on the cell line, the specific molecular mechanisms leading to cell death may significantly differ. Precise characterization of these differences is crucial for understanding and exploiting extrinsic apoptosis. Cells show distinct behaviors on several aspects of apoptosis, including *(i)* the relative order of caspases activation, *(ii)* the necessity of mitochondria outer membrane permeabilization (MOMP) for effector caspase activation, and *(iii)* the survival of cell lines overexpressing Bcl2. These differences are attributed to the activation of one of two pathways, leading to classification of cell lines into two groups: type I and type II. In this work we challenge this type I/type II cell line classification. We encode the three aforementioned distinguishing behaviors in a formal language, called signal temporal logic (STL), and use it to extensively test the validity of a previously-proposed model of TRAIL-induced apoptosis with respect to experimental observations made on different cell lines. After having solved a few inconsistencies using STL-guided parameter search, we show that these three criteria do not define consistent cell line classifications in type I or type II, and suggest mutants that are predicted to exhibit ambivalent behaviors. In particular, this finding sheds light on the role of a feedback loop between caspases, and reconciliates two apparently-conflicting views regarding the importance of either upstream or downstream processes for cell-type determination. More generally, our work suggests that these three distinguishing behaviors should be merely considered as type I/II features rather than cell-type defining criteria. On the methodological side, this work illustrates the biological relevance of STL-diagrams, STL population data, and STL-guided parameter search implemented in the tool Breach. Such tools are well-adapted to the ever-increasing availability of heterogeneous knowledge on complex signal transduction pathways.

Citation: Stoma S, Donzé A, Bertaux F, Maler O, Batt G (2013) STL-based Analysis of TRAIL-induced Apoptosis Challenges the Notion of Type I/Type II Cell Line Classification. PLoS Comput Biol 9(5): e1003056. doi:10.1371/journal.pcbi.1003056

Editor: Stanislav Shvartsman, Princeton University, United States of America

Received November 16, 2012; **Accepted** March 26, 2013; **Published** May 9, 2013

Funding: This work was supported by the research grant Syne2arti ANR-10-COSINUS-007 from the French National Research Agency. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: gregory.batt@inria.fr

☯ These authors contributed equally to this work.

¤ Current address: EECS Department, University of California Berkeley, Berkeley, California, United States of America.

Introduction

Apoptosis, a major form of programmed cell death, plays a crucial role in shaping organs during development and controls homeostasis and tissue integrity throughout life [1,2]. Moreover defective apoptosis is often involved in cancer development and progression [3]. Apoptosis can be triggered by *intrinsic* or *extrinsic* stimuli. Intrinsic apoptosis is triggered in case of cell damage (e.g. stress, UV radiation) or cell malfunction (e.g. oncogene activation). Extrinsic apoptosis is initiated by the presence of extracellular death ligands, such as Fas ligand (FasL), Tumor Necrosis Factor (TNF), or TRAIL [2]. Because the latter has a unique ability to trigger apoptosis in various cancer cell lines without significant toxicity toward normal cells, TRAIL-induced apoptosis has been the focus of extensive studies [1].

The effects of TRAIL application can be significantly different from one cell line to another [4–6]. The current understanding is that cell death results from the activation of one of two parallel pathways, leading to the classification of cell lines into two distinct cell types. In type I cells, effector caspases are directly activated by initiator caspases. Mitochondria outer membrane permeabilization (MOMP) is not required to generate lethal levels of caspase activity. In type II cells, the activation of initiator caspases triggers MOMP that in turn triggers effector caspases activation. MOMP is required for cell death. This necessity of mitochondrial pathway activation to undergo apoptosis is often referred as *type II phenotype*, in contrast to *type I phenotype* where MOMP is a side effect of apoptosis.

Many models of apoptosis, based on different mathematical formalisms, ranging from logical models to differential equation systems, have been proposed so far [2,6–21]. To investigate the molecular origins of the two above-mentioned distinct phenotypes, Aldridge and colleagues developed a model describing key biochemical steps in TRAIL-induced apoptosis: extrinsic apoptosis reaction model (EARM1.4) [6]. EARM1.4 is an extension of a model developed to capture cell-to-cell variability in apoptosis of

Author Summary

Apoptosis, a major form of programmed cell death, plays a crucial role in shaping organs during development and controls homeostasis and tissue integrity throughout life. Defective apoptosis is often involved in cancer development and progression. Current understanding of externally triggered apoptosis is that death results from the activation of one out of two parallel signal transduction pathways. This leads to a classification of cell lines in two main types: type I and II. In the context of chemotherapy, understanding the cell-line-specific molecular mechanisms of apoptosis is important since this could guide drug usage. Biologists investigate the details of signal transduction pathways often at the single cell level and construct models to assess their current understanding. However, no systematic approach is employed to check the consistency of model predictions and experimental observations on various cell lines. Here we propose to use a formal specification language to encode the observed properties and a systematic approach to test whether model predictions are consistent with expected properties. Such property-guided model development and model revision approaches should guarantee an optimal use of the often heterogeneous experimental data.

HeLa cells [15,16]. In [6], the authors tested the hypothesis that the distinct cell behaviors can be explained solely by measured differences in protein concentrations before stimulation among different cell lines. Cell line models share the same set of ordinary differential equations and kinetic parameters, but possess specific protein contents at the initial state (i.e. before TRAIL application). These differences in the initial concentrations of a dozen of key apoptotic proteins are consistent with quantitative immunoblotting measurements. Then the authors use an abstract criterion that measures the influence of changes in initial protein concentrations on the future states of the system (i.e. divergence of trajectories): the direct finite-time Lyapunov exponent (DLE). They show that this criterion defines a partition of the state space that preserves known differences between phenotypes: type I and type II cells are associated to distinct regions in the state space [6]. The DLE-induced partition can be graphically represented as 2D slices of the high dimensional state space called DLE diagrams [6,22]. As shown in [6], DLE diagrams are intuitive tools to predict the effect of mutations on cell type. However, the connection between the abstract DLE notion and cell phenotypes remains elusive: why type I and type II cells correspond to two different regions separated by a third one having high DLE values? Understanding this relationship is important to evaluate the general applicability of the proposed approach. Moreover in [6], the authors also probed the functioning of the apoptotic pathways in different cell lines and for different mutants using three different experimental methods: clonogenic assays, microscopy imaging and flow cytometry measurements of immunostained cells. These experiments probe subtly different aspects of the interplay of different pathway components, and most notably on the role of MOMP in the apoptotic response: death/survival following TRAIL stimulation of derived cell lines overexpressing Bcl2 (Property 1), synchronous/sequential activation of initiator and effector caspases (Property 2), and effector caspase activation prior/posterior to MOMP (Property 3). However, the authors do not test the consistency of EARM predictions with the detailed experimental information they provide.

In this work we address the two above-mentioned problems by using a formal language, signal temporal logic (STL). STL was originally developed for monitoring purposes to specify the expected behavior of physical systems, including notably the order of physical events as well as the temporal distance between them [23]. Like other temporal logics and formal verification frameworks [24–31], it has been applied to the analysis of biomolecular networks [32,33]. In particular, because it allows expressing in a rigorous manner transient behaviors of dynamical systems, one can encode as STL properties various cellular responses observed with different experimental methods and associated to type I/II phenotypes. Because STL properties have a quantitative interpretation, describing how robustly behaviors of the system satisfy or violate the property, STL diagrams can be constructed analogously to DLE diagrams. However, since STL diagrams are each associated to a specific STL property their interpretations do not suffer from ambiguities. Moreover, one can benefit from the expressive power of the STL language to encode detailed experimental information and thoroughly test the consistency of EARM with the various observations (Figure 1).

We report three findings. Firstly, our results highlighted that the three experimental methods proposed in [6] to investigate the importance of MOMP for cell death from three different perspectives, each suggesting a type I/II distinguishing criterion, do not lead to consistent cell line classifications. For example the ΔXIAP HCT116 cell line should be classified as type II based on Properties 1 and 2, and as type I based on Property 3. This challenges the well-posedness of the type I/II notion. Secondly, using our systematic approach, we found several inconsistencies between model predictions and actual observations. Taking again advantage of the quantitative interpretation of STL properties, we searched for valid parameters using a cost function that is minimal when all properties are consistent with experimental data and state-of-the-art global optimization tools. Inconsistencies have been resolved simply by modifying a few parameters, thus showing that there is no need for structural changes in the model. Thirdly, our findings reconciliate the apparently contradictory views expressed by Scaffidi and colleagues [5] and Aldridge and colleagues [6] about the origins of type I and II phenotypes. Indeed, Scaffidi, Barnhart and colleagues suggest that the initiator caspase activation capabilities are the main determinants of the type I/II phenotype of a cell line [5,34], whereas Eissing and colleagues, Jost and colleagues, and Aldridge and colleagues suggest that the latter is mainly controlled by the relative abundance of downstream proteins, most notably XIAP and caspase-3 [4,6,7]. Our results suggest that, unlike downstream proteins, the modification of the concentration of upstream proteins within physiological range has a negligible effect on cellular responses. However, the critical effects of downstream protein concentration changes are fed back to upstream processes and are amplified via a positive feedback loop involving caspases 3, 6, and 8, leading to the activation of initiator caspases. Finally, the comparison of the STL and DLE diagrams showed that the DLE criterion essentially captures the notion of cell survival or cell death, like Property 1. This lead us to better understand why the fairly abstract DLE criterion induced biologically-relevant partitions in the work of Aldridge and colleagues [6]. A last contribution is that we extended the functionalities of the Breach tool [33] so that phase diagrams can be automatically computed given any differential equation model and STL property. Therefore, the methodology presented here can be applied to other complex biomolecular networks.

The first three sections of the Results part deal with the detailed analysis of three different observed phenotypes associated with

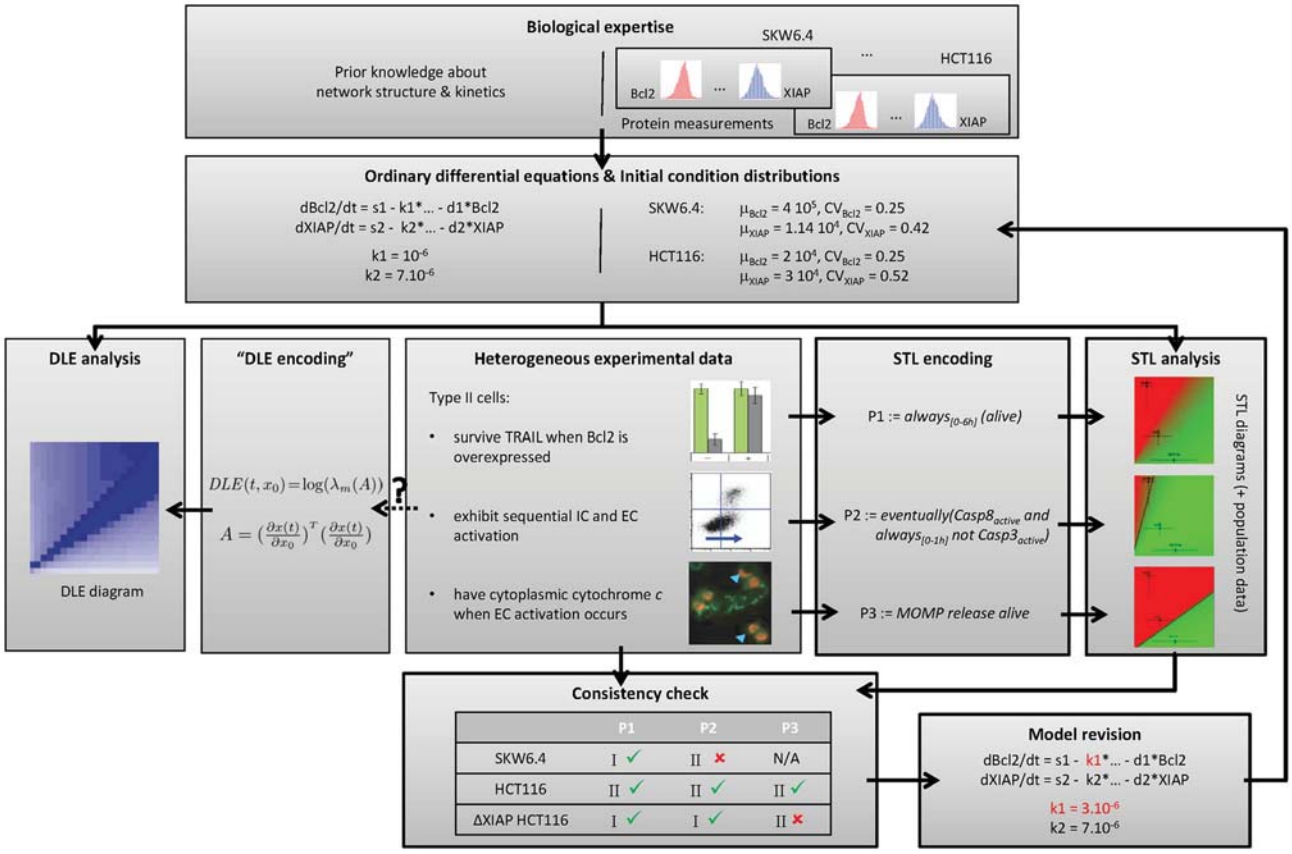


Figure 1. Property-based model analysis framework. Heterogeneous observations on the system are formalized as STL properties. Consistency between model and experimental observations is tested via STL diagrams and population data. Inconsistencies can be resolved via property-guided model revision. In contrast to DLE, STL properties explicitly encode specific aspects of cell's response, in our case, of the role of mitochondria in type I/II apoptosis. Bold boxes allows distinguishing our contribution with respect to the work of Aldridge and colleagues [6]. (Images reused with permission from Nature Publishing Group).
doi:10.1371/journal.pcbi.1003056.g001

type I/II behaviors, encoded in STL, and confronted with model predictions. In the last two sections, we study whether the EARM model can be reconciled with all the considered observations on all cell lines and search for the origins of cell type differences.

Results

Property 1: Type II cells survive if Bcl2 is over-expressed

STL encoding. Bcl2 over-expression is the standard experimental method for distinguishing type I and type II cells [5]. Type I cells overexpressing the anti-apoptotic protein Bcl2 die in the presence of death ligand but type II cells survive. The sequestration of Bax by high levels of Bcl2 prevents the formation of pores in the mitochondrial outer membrane (Figure 2). Therefore clonogenic survival of an OE-Bcl2 derived cell line reveals the need for MOMP to trigger cell death in type II cells. Clonogenic survival data is available in [6] for three cell lines, SKW6.4 (human B lymphoma cells), HCT116 (human colon carcinoma cells), T47D (human breast carcinoma cells), and for the ΔXIAP mutant of HCT116 cells [6,35,36]. Cells were exposed to a 50 ng/ml TRAIL treatment for 6 hours.

Here, we encode in STL the observations made in clonogenic assays on HCT116, SKW6, and T47D cells [6]. Effector caspases cleave essential structural proteins and inhibitors of DNase, leading eventually to cell death. PARP is a substrate of these effector caspases and its cleavage is often regarded as a marker of commitment to death by cells [15,16,37]. Therefore, we consider here that a cell is alive if less than a half of the PARP proteins is cleaved. In STL, this translates into: $alive := cPARP/PARP_{total} < 0.5$. Note that although the 50% threshold used here is somewhat arbitrary, we found that our conclusions are robust with respect to threshold changes in the range 10%–90% (see Figure S1). Then cell survival is simply expressed in STL as the cell is always alive: $Property1 := always_{[0-6h]}(alive)$. Here, *always* is an STL keyword (see Methods). Its scope is limited to the first 6 hours as in experiments.

STL phase diagrams. For each initial protein concentration, one can predict the behavior of the system after TRAIL stimulation and assess whether this behavior satisfies a given STL property, or more precisely, estimate the value of the STL property given the behavior (see Methods). One can then graphically represent the value of the property in the state space by so-called phase diagrams (see Methods). The placement of cell lines in the phase diagram, based on their initial protein concentrations, indicates whether the cell line satisfies the given property (see Methods). Since it has been shown that the ratio of XIAP to caspase-3 concentrations plays a key role for the determination of the apoptotic type [6], we first constructed diagrams associated with these two variables. The corresponding STL phase diagram associated to Property 1 is represented in Figure 3. The death/survival property is tested in derived cell lines where Bcl2 is overexpressed (OE-Bcl2 cells; 10-fold increase of Bcl2 initial concentrations). The presence of two distinct regions in

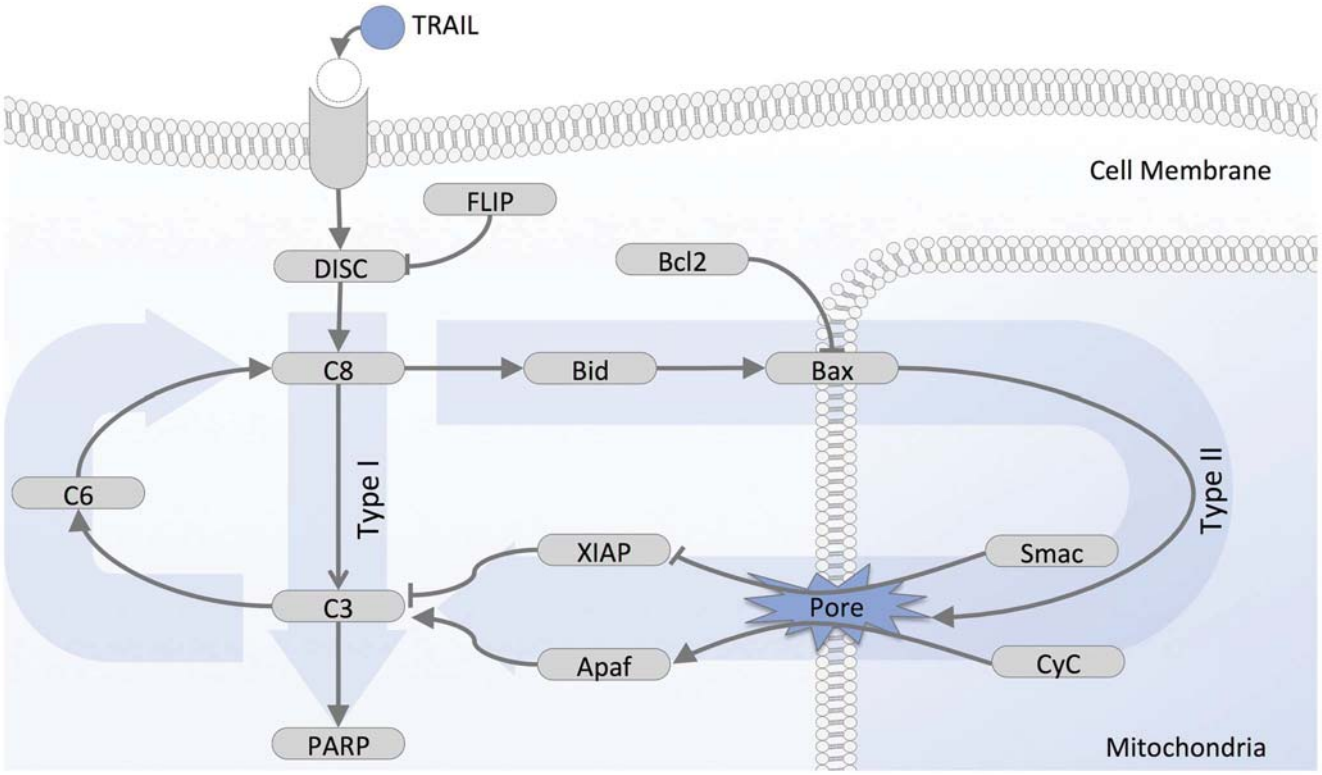


Figure 2. Simplified view on TRAIL-dependent apoptotic pathway. The activation of the membrane receptor by TRAIL binding promotes the assembly of the death-inducing signaling complexes (DISC), which recruit and activate initiator (pro-) caspases, including notably caspase-8 (C8) [53]. The recruitment of initiator caspases is modulated by FLIP. Once activated, initiator caspases cleave and activate effector caspases such as caspase-3 (C3). This effect is reinforced by a feedback loop involving caspase-6 (C6). Effector caspases cleave essential structural proteins, inhibitors of DNase, and DNA repair proteins (PARP), eventually leading to cell death. The cellular effect of effector caspase activation is regulated by factors such as the X-linked inhibitor of apoptosis protein (XIAP), which blocks the proteolytic activity of caspase-3 by binding tightly to its active site [54] and promotes its degradation via ubiquitination [55]. In addition to the direct activation of effector caspases, initiator caspases also activate Bid and Bax [56]. If not kept in check by inhibitors, most notably Bcl2, activated Bax directly contributes to the formation of pores in the mitochondria outer membrane, leading to MOMP [57]. Following MOMP, critical apoptosis regulators, such as Smac and cytochrome c (CyC), translocate into the cytoplasm. Smac binds to and inactivates XIAP, thus relieving the inhibition of effector caspases by XIAP [58]. Cytochrome c combines with Apaf-1 to form the apoptosome that in turn activates the initiator caspase-9 that activates effector caspases.
doi:10.1371/journal.pcbi.1003056.g002

the diagram, one where Property 1 is satisfied (positive values, green) corresponding to cell survival, typical of type II cells, and one where Property 1 is falsified (negative values, red) corresponding to cell death, typical of type I cells, suggests that the model correctly predicts the importance of the XIAP/caspase-3 ratio as a key factor to determine cell survival following TRAIL treatment. We then positioned cell lines in the diagram based on measured mean and standard deviations of protein concentrations (see Methods). In agreement with the observations (Figure 2B in [6]) and the known type of these cell lines, the STL diagram predicts that OE-Bcl2 HCT116 cells do satisfy Property 1, but OE-Bcl2 SKW6.4 cells do not. OE-Bcl2 T47D cells are located close to separatrix and most cells satisfy Property 1. This is only in partial agreement with the fact that only half of T47D cells were found to survive (Figure 7C in [6]). Interestingly, as noted by Aldridge and colleagues, one can immediately see the consequences of mutations [6]. For example, ΔXIAP cell lines are shifted to the leftmost part of the diagram (regions with low XIAP concentrations) and are thus predicted to violate Property 1. That is, all OE-Bcl2/ΔXIAP mutants of the HCT116, SKW6.4, and T47D cell lines are predicted to die in clonogenic experiments. This is again in accordance with experimental observations for HCT116 cells (Figure 2B in [6]). A detailed comparison of the Property 1 diagrams and the DLE diagrams used in [6] shows that the successful classification of cells provided by DLE diagrams implicitly relies on the snap-action, all-or-none aspect of apoptosis (Figure S2). Using the approach we propose here, the property of interest is explicitly stated and the interpretation of the resulting diagrams is not ambiguous. Moreover, since STL is a property specification language, this framework can be applied to analyze other properties of the system, not necessarily relying on snap-action responses.

STL population data. Using STL diagrams helps in understanding how cell behavior depends on its initial protein content and hence suggests why cell lines exhibit different phenotypes. However, DLE and STL diagrams suffer from some limitations. In both cases, only two initial protein concentrations are modified (XIAP and caspase-3 in our examples). Therefore, they fail to capture all the differences between cell lines. In more precise terms, DLE and STL diagrams represent the value of the DLE or of an STL property in a 2D slice of the high-dimensional state space, and cell line distributions are projected onto the slice. Therefore, even if they provide insight into the behavior of cells that are affected by changes in initial protein concentrations, DLE and STL diagrams must be interpreted with care. The information is exact for the cell line used to construct the diagram, called the

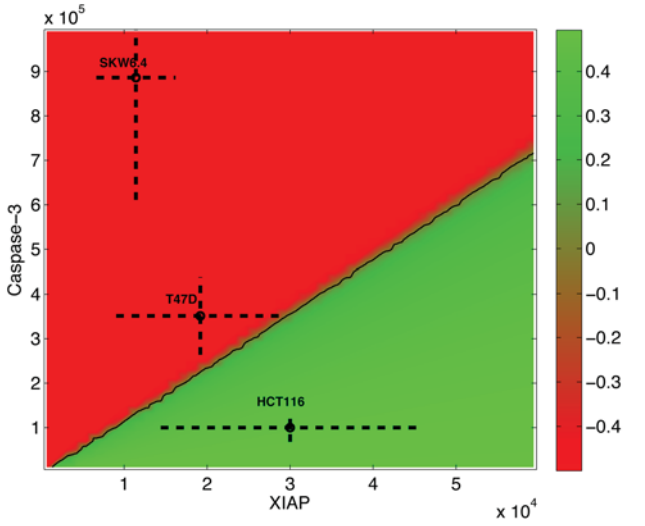


Figure 3. Property-1 phase diagram. Each point in the diagram represents a different initial concentration for XIAP and caspase-3 proteins, and its color represents the value of Property 1 evaluated on cell simulated behavior starting in these initial conditions ($p1:=always_{[0-6h]}(cPARP/PARP_{total}<0.5)$). Other protein concentrations correspond to nominal protein concentrations for HCT116 cells (Table 1). Green regions satisfy the property (positive values). Red regions do not (negative values). Cell lines can be positioned in this diagram, using crosses which center and size are determined by the mean and standard deviation of measured protein concentrations (Methods and [6]).
doi:10.1371/journal.pcbi.1003056.g003

reference cell line, but it is only approximate for other, projected cell lines. To investigate how the diagrams change when reference cell line change, we constructed the Property 1 diagrams with respect to OE-Bcl2 HCT116, OE-Bcl2 SKW6.4, and OE-Bcl2 T47D cell lines (Figure S3A–C). Although the conclusions based on Figure 3 are indeed valid for HCT116 and SKW6.4 cell lines (OE-Bcl2 HCT116 cells survive, OE-Bcl2 SKW6.4 cells die), they differ for T47D cells. When using a slice of the state space based on OE-Bcl2 T47D cells, it appears that these cells are classified as exhibiting mixed-type behaviors (Figure S3C), as experimentally observed, instead of mostly type I as suggested by Figure 3. This example illustrates that problems may arise when placing different cell lines on the same phase diagram. To obtain a less comprehensive but more accurate view of the value of STL properties, we propose to use *STL population data* in combination with phase diagrams. Population data are statistics describing the STL property values associated to whole cell populations (see Methods). For Property 1, these statistics are presented in Figure 4 (and Figure S4 for all cell lines). One can first check that indeed the mean values, distributions and satisfaction rates of Property 1 are qualitatively consistent with the predictions we obtained from the STL diagram in Figure 3 for the OE-Bcl2 HCT116, OE-Bcl2 SKW6.4, OE-Bcl2 T47D, and OE-Bcl2/ΔXIAP HCT116 cells. Moreover, the satisfaction rates in Figure 4 can be directly compared with the experimentally-measured survival rates in clonogenic assays (Figure 2B and Figure 7C in [6]). Strikingly, our data shows excellent quantitative agreement with observed cell behaviors for all but the parental T47D cell line. Like in clonogenic assays, we predict the survival of a large majority of OE-Bcl2 HCT116 cells, half of the OE-Bcl2 T47D cells and a minority of HCT116 cells, the death of all ΔXIAP HCT116 and SKW6.4 cells and their OE-Bcl2 variants. The sole discrepancy concern T47D cells that are predicted to be more resistant to apoptosis than experimentally-observed.

Property 2: Activations of initiator and effector caspases are sequential in Type II cells

STL encoding. In addition to survival of derived cell lines overexpressing Bcl2, Scaffidi and colleagues observed another important difference between type I and II cell lines: the dynamics of the activations of initiator and effector caspases by cleavage shows marked differences [5]. These are two critical events that can be considered as markers of the beginning and of the end of the apoptosis decision-making process. By using Western blots, Scaffidi and colleagues showed that in type I cells the activation of the effector caspase caspase-3 closely follows the activation of the initiator caspase caspase-8: caspase activations are gradual and near synchronous. In contrast, in type II cells the activation of initiator caspases is not closely followed by the activation of effector caspases [5]. Similar results have been obtained with a cellular resolution using FACS analysis (Figure 5 in [6]). The current understanding is that effector caspase activation is delayed until MOMP happens. Hence, the observed sequential activation is explained by a pre-MOMP delay in type II cells. Therefore this synchronous versus sequential activation is not only a robustly observed pattern but also relates to mechanistic interpretation of cell death.

We will consider that initiator and effector caspases activations are sequential if they are separated by more than one hour, in accordance with the low-temporal resolution of available observations in [5,6]. So to express sequential activation, we say that "at some time point, caspase-8 is active and for at least an hour, caspase-3 remains inactive". Hence, we have the following STL formula: $Property2:=eventually(Casp8_{active}\ and\ always_{[0-1h]}\ not\ Casp3_{active})$. We still have to set the threshold concentration for cleaved caspases that corresponds to a detectable activity. Since it has been shown that caspases are highly potent proteases (a few hundred caspases can cleave millions of substrate proteins within hours [38,39]), we set this threshold concentration to 1% of the total caspase concentration: $Casp8_{active}:=Casp8^*/Casp8_{total}>1\%$, and $Casp3_{active}:=Casp3^*/Casp3_{total}>1\%$ where $Casp8^*$ and $Casp3^*$ are the sum of the concentrations of all cleaved forms of caspase-8 and caspase-3, with the exclusion of caspase-8 bound to Bar and of caspase-3 bound to XIAP, respectively (the influence of the threshold is discussed in Figure S1).

STL phase diagrams and population data. Having formalized our property in STL, one can automatically construct the corresponding diagram (Figure 5, left). On this diagram one can clearly see two distinct, positive and negative, regions. HCT116 and T47D cells lie in the positive region and hence are predicted to satisfy Property 2, whereas SKW6.4 cells lie on the separatrix, and hence are predicted to show a mixed phenotype with respect to Property 2. Note that in the case of SKW6.4 cells, it is important to consider the diagram computed with respect to this cell line to have an accurate representation (compare S3D and E). The diagram also predicts that ΔXIAP mutants violate Property 2 (i.e. lie in the negative region). The predicted phenotypes of HCT116 and of ΔXIAP HCT116 cells are consistent with observations: whereas HCT116 cells show a clear sequential activation of caspases, this behavior is lost in ΔXIAP HCT116 cells [6]. Diagram shows that EARM1.4 is also compatible with the hypothesis that high levels of XIAP control caspase activation and substrate cleavage, and may promote apoptosis resistance and sublethal caspase activation *in vivo* [13]. However, the predicted phenotype of SKW6.4 cells is in contradiction with the observed one (Figure S3E and Figure 5

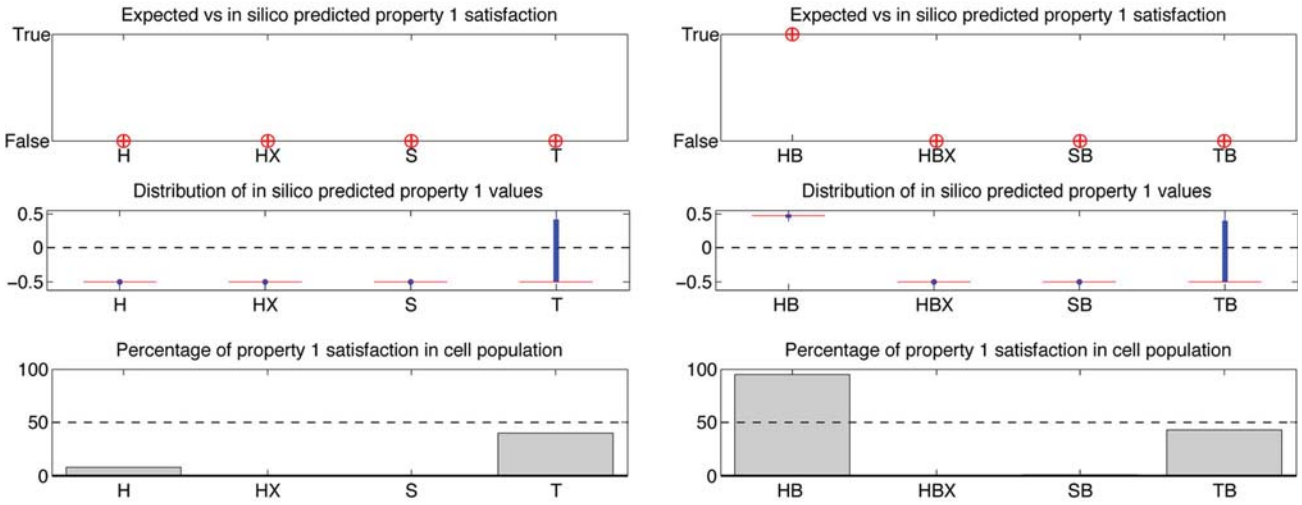


Figure 4. Property 1 population statistics. Plots indicate the satisfaction of Property 1 by the nominal cell (cross, top), the distribution of the values (middle), and the percentage of satisfaction (bottom) of Property 1 for populations of cells of different cell lines. For distributions, box boundaries and red line indicate first and third quartiles, and median, respectively. When experimental data is available, circles in the top plot represent the expected values. The following abbreviations are used in this and further figures: H is HCT116, HX is ΔXIAP HCT116, HB is OE-Bcl2 HCT116, HBX is OE-Bcl2/ΔXIAP HCT116, S is SKW6.4, SX is ΔXIAP SKW6.4, SB is OE-Bcl2 SKW6.4, SBX is OE-Bcl2/ΔXIAP SKW6.4, T is T47D, TX is ΔXIAP T47D, TB is OE-Bcl2 T47D and TBX is OE-Bcl2/ΔXIAP T47D.
doi:10.1371/journal.pcbi.1003056.g004

Right). As expected from type I cells, SKW6.4 cells clearly show synchronous activations of caspases and shoud therefore violate Property 2. The analysis of the OE-Bcl2 mutants of the HCT116, ΔXIAP HCT116, and SKW6.4 cell lines shows that consistent results are obtained in these cases (Figure 5, right). One should note that because caspase-3 is not activated in OE-Bcl2 HCT116 cells (they survive TRAIL treatment), Property 2 holds trivially in these cells. To investigate whether EARM1.4 can account for the observed phenotype of SKW6.4 cells, we slightly relaxed the timing constraint between the caspases activation times and found that by setting a slightly longer delay (e.g. 1h30 min), the mean value of Property 2 for the SKW6.4 cell population becomes negative as expected, and even more, that the percentage of cells satisfying Property 2 decreases to zero with longer delays. Therefore we conclude that the observed discrepancy results from EARM 1.4 limitations to capture quantitatively the elapse of time between events, rather than from severe modeling flaws.

Property 3: MOMP precedes caspase-3 activation in Type II cells

STL encoding. In type I and type II cells, MOMP happens during apoptosis with comparable kinetics [5]. This is in apparent contradiction with the very different role of MOMP in the two

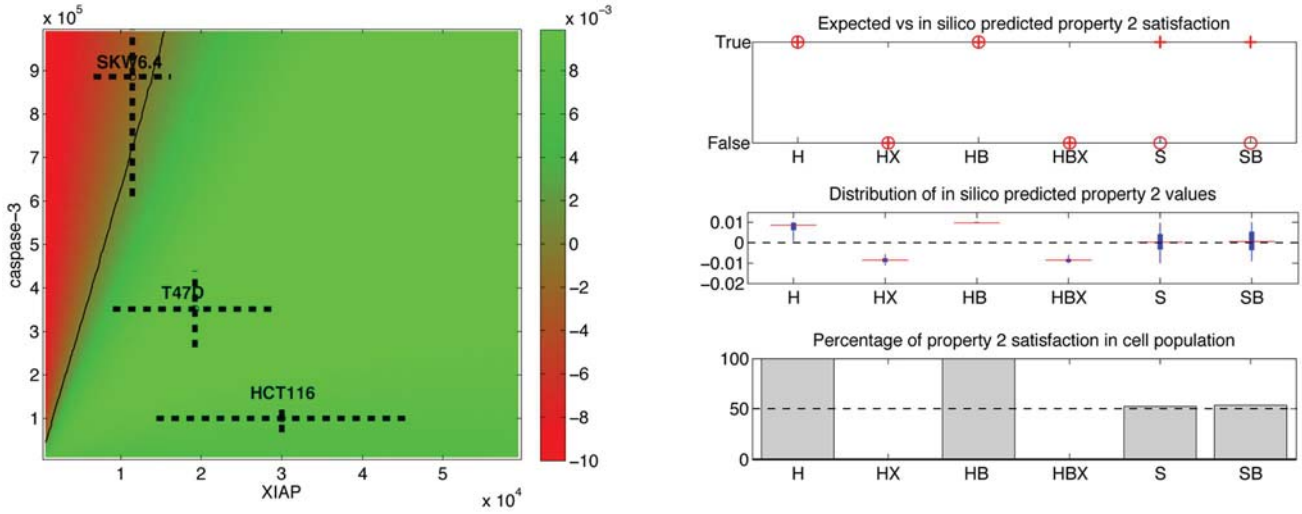


Figure 5. Property-2 phase diagram and statistics. Left: Values of Property 2 evaluated on cell simulated behaviors and represented as a function of the XIAP and caspase-3 initial concentrations ($p2: = eventually(Casp8_{active}$ *and* $always_{[0-1h]}$ *not* $Casp3_{active}$)). Other protein concentrations correspond to nominal protein concentrations for HCT116 cells. As in Figure 3, cell lines are positioned in this diagram according to their protein initial concentrations. Right: distributions of the values of Property 2 across populations of cells of different cell lines. Notations are identical to those used in Figure 4. Note the discrepancy between the predicted (cross) and expected (circle) values for Property 2 in SKW6.4 cells.
doi:10.1371/journal.pcbi.1003056.g005

pathways, as revealed by Bcl2 overexpression experiments (Property 1), and with the different kinetics of caspases activations (Property 2). The current understanding is that in type I cells MOMP is a consequence of effector caspases activation, whereas in type II cells, MOMP is the cause of effector caspases activation [5,34,40]. Under this assumption one should observe that in the first case MOMP follows effector caspases activation, and in the second case, that MOMP precedes effector caspases activation. This question has been directly investigated by Aldridge and colleagues by staining cells with anti-cytochrome c and anti-cPARP antibodies [6]. The authors demonstrate that most of HCT116 cells showing effector caspases activation also show cytoplasmic cytochrome c localization indicating that MOMP has happened. Stated differently, caspase-3 is not active until MOMP happens. This is not always true for ΔXIAP HCT116 cells, or for OE-Bcl2 ΔXIAP HCT116 cells: a significant proportion (respectively 20% and 80% of the cells) shows effector caspase activation in absence of cytoplasmic cytochrome c (Figure 3 in [6]). The same experiment was made for T47D and OE-Bcl2 T47D cells, showing that these cells behave like HCT116 cells: caspase-3 is not active until MOMP happens (Figure 7 in [6]).

To test the consistency of EARM1.4 with these observations, we express in STL the property, typical of type II behaviors, that cells remain alive until MOMP happens. We simply write *Property3: =MOMP release alive*. The *release* operator states that the second property (*alive*) must hold until the first property holds for the first time (*MOMP*). The occurrence of MOMP is detected by the titration of Apaf-1 by the released cytochrome c to form the apoptosome. We say that MOMP happened when more than 50% of Apaf-1 is bound to cytochrome c: $MOMP: = Apaf_{free}/Apaf_{total} < 0.5$ (see Figure S1 for discussion of threshold).

STL phase diagrams and population data. We used Breach to compute the STL diagram associated with Property 3 with respect to HCT116 cells, and the Property 3 population data (Figure 6). One should note that like in the *in vitro* setup of [6], only cells in which MOMP happened were taken into account: we excluded surviving cells to compute statistics.

The diagram presented in Figure 6 is consistent with the observation that HCT116 cells satisfy the property. However, it suggests that the ΔXIAP mutant falsify the property, since negative values are found in the region where XIAP concentration is null. This is in contradiction with the observation that caspase-3 is active before MOMP happened in a majority (80%) of these cells. In summary, ΔXIAP HCT116 cells present a type I phenotype with respect to clonogenic survival and caspases activation dynamics, and a type II phenotype with respect to the need for MOMP for cell death, but the model classifies them as type I cells for all properties. In fact, the fact that ΔXIAP HCT116 cells have been observed to satisfy Property 3 but not Property 2 imposes strong constraints on the kinetics of the apoptosis process. In these cells, caspase-3 activation is precocious, since it follows by less than one hour the activation of caspase-8, implying that death (i.e. PARP cleavage) is rapid since it follows shortly after caspase-3 activation. But then Property 3 implies that MOMP happened even before this time instant. Given the efficient caspase-3 activation in EARM1.4, the model fails to capture the need for MOMP in these cells. Lastly, one can note that contradictions are also found with T47D, and OE-Bcl2 T47D cells (Figure 6, right).

Improving EARM 1.4: Property-guided parameter search

In summary, we found that EARM1.4 satisfies the majority of the observed behaviors encoded in STL (Figure 7). This is commendable for a model of this size and complexity, given that EARM1.4 has not been tuned with respect to these properties, even if the model and the specific observations we used to state our STL properties have been published in the same paper [6]. However, few discrepancies were identified. It is important to test whether the proposed model is structurally not capable of accounting for all the observed properties. If not, this would call for significant model revision.

We first tried to resolve inconsistencies by minor adjustments of the thresholds we used in formulae. However, property satisfaction values proved robust to threshold changes (Figure S1).

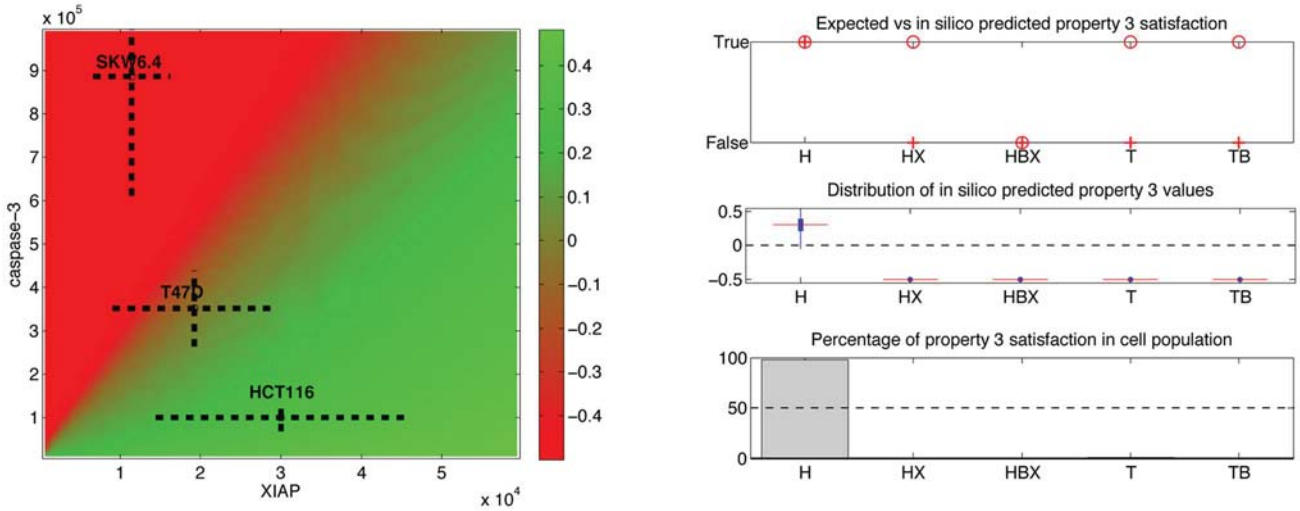


Figure 6. Property-3 phase diagram and statistics. The value of Property 3 (*p3: =MOMP release alive*) is represented as a function of the XIAP and caspase-3 initial concentrations. Other protein concentrations correspond to nominal protein concentrations for HCT116 cells. As in Figure 3, cell lines are positioned in this diagram according to their protein initial concentrations. Note that HCT116 cells depleted from all XIAP ([XIAP] = 0) are predicted not to satisfy Property 3. This is in contradiction to experimental observations [6]. (Right) Distributions of the values of Property 3 across populations of cells of different cell lines. Notations are identical to those used in Figure 4. One can note that discrepancies between predicted mean values (crosses) and observed phenotypes (circles) exist for ΔXIAP HCT116, T47D and OE-Bcl2 T47D cells.
doi:10.1371/journal.pcbi.1003056.g006

Cell line \ Property	Property 1	Property 2	Property 3
HCT116	False (Fig 2B in [6]) ✓	True (Fig 5A in [6]) ✓	True (Fig 3A in [6]) ✓
ΔXIAP HCT116	False (Fig 2B in [6]) ✓	False (Fig 5A in [6]) ✓	True (Fig 3A in [6]) ×
OE-Bcl2 HCT116	True (Fig 2B in [6]) ✓	True (Fig 5A in [6]) ✓	N/A
OE-Bcl2 ΔXIAP HCT116	False (Fig 2B in [6]) ✓	False (Fig 5A in [6]) ✓	False (Fig 2B in [6]) ✓
SKW6.4	False (Fig 2B in [6]) ✓	False (Fig 5B in [6]) ×	N/A
OE-Bcl2 SKW6.4	False (Fig 2B in [6]) ✓	False (Fig 5B in [6]) ×	N/A
T47D	False (Fig 7C in [6]) ✓	N/A	True (Fig 7E in [6]) ×
OE-Bcl2 T47D	Mixed (Fig 7C in [6]) ✓	N/A	True (Fig 7E in [6]) ×

p1:= always$_{[0\text{-}6h]}$(alive); p2:= eventually(Casp8$_{active}$ and always$_{[0\text{-}1h]}$ not Casp3$_{active}$); p3:= MOMP release alive.

Figure 7. Summary of findings. Truth values of the three properties based on observations in [6] for the HCT116, SKW6.4, and T47D cell lines and some mutants. N/A indicates that the experimental information is not available. Experiments showed that OE-Bcl2 T47D cells present clonogenic survival rates close to 50%, hence their "mixed" behavior. Consistency or discrepancy with predictions from the original EARM 1.4 model obtained using our approach are indicated by green or red marks. Because of their ambiguous phenotypes, T47D cell data (in grey) were not used for parameter search.
doi:10.1371/journal.pcbi.1003056.g007

We therefore resorted to search for better parameter values using global optimization methods [41]. We defined a cost function that indicates for any given parameter how far the model is from satisfying all its constraints. More precisely, the cost function aggregates three measures: how many properties are consistent with the observations, how robustly satisfied or falsified they are, and how large are the deviations of the parameters with respect to their reference values. Then, we used the global optimization tool CMA-ES [42] to search automatically for parameters minimizing this cost function (see Methods). Here, one should note that the real-valued semantics of STL properties is critical: continuous optimization tools take advantage of the graded interpretation of STL properties, whose values indicate their "distances from satisfaction". The sole use of traditional Boolean-valued interpretations of temporal logic formulae would have made this search impractical. Because of their ambiguous phenotypes, T47D cell data were not used for parameter search. We started with 43 parameters, that is, all catalytic and forward reaction rates (see Method section). After applying our optimization procedure we found that the modification of only 2 parameters was sufficient to achieve full agreement with experimental data. The parameters found by the search procedure are a parameter regulating the strength of the caspases feedback loop (2.71 fold increase) and a parameter regulating the kinetics of PARP cleavage (55.6 fold decrease) (Table S1). Given the usually large uncertainties in actual parameter values, such changes can still be considered as acceptable. New parameter values lead to satisfaction of Property 1–3 in *nominal* cells corresponding to all HCT116 and SKW6.4 normal and derived cell lines. To test whether property values are corrected at the *cell population* level, we recomputed the population data with these new parameter values. As shown in Figure S5, all inconsistencies were indeed resolved at the cell population level for all cell types (again with the exception of T47D cells).

Origins of type I/II behaviors: Key role of downstream proteins and of a positive feedback loop

It is important to note that the significantly different phenotypic responses of the different cell lines are in the model solely explained by observed differences in the initial concentrations of a dozen of key proteins. Therefore one can use EARM1.4 with new parameter values (Table S1) and STL diagrams to investigate the *origins* of the different behaviors shown by cell lines. One important question is to distinguish whether the different behaviors can be explained exclusively by differences in upstream protein concentrations or exclusively by differences in downstream protein concentrations, or whether a combination of upstream and downstream changes is needed [6]. Indeed, it has been proposed that the main differences between type I and II behaviors are essentially due to differences in the efficiency of initiator caspase activation by the DISC [5,34,43]. It has also been proposed that the main determinant is the concentration of XIAP relative to caspase-3 [4,6]. These questions can easily be answered using STL diagrams. Figure 8 shows the XIAP/caspase-3 and FLIP/caspase-8 diagrams for Property 2, computed with respect to the HCT116 cell line. It is apparent that the sole change of the concentrations of XIAP and capsase-3 from their original values to values corresponding to SKW6.4 cells is sufficient to alter the behavior of those cells from a type II to a type I phenotype. A similar change but for FLIP and caspase-8 proteins has no effect: cells remains with a type II phenotype. As illustrated in Figure S6 and S7, this is true for all properties and both directions (i.e., modifying protein concentrations from HCT116 to SKW6.4 values and vice-versa). This lack of influence of any upstream protein concentration is in apparent contradiction with the markedly different profiles for caspase-8 activation observed experimentally in [5] between type I and type II cells, and in EARM1.4 between HCT116 and SKW6.4 cell lines, and even more between normal and ΔXIAP cells that differ only in XIAP concentration (Figure 8C and 8D). The latter comparison suggests that differences in downstream protein concentrations feed back on upstream protein activities. To test this, we created feedback mutants (denoted ΔFB cells) by zeroing the cleaveage rate of caspase-8 by caspase-6. Similar activation profiles for caspase-8 in HCT116 and SKW6.4 cell lines were then obtained showing that in normal conditions differential activation of upstream processes is the consequence of differential downstream processes activation (Figure 8C and 8D). So, one can reconciliate the different views expressed by Scaffidi and colleagues and by Aldridge and colleagues [5,6]. There are indeed functionally significant differences in upstream protein activities (e.g. caspase-8) in type I and II cells. However, according to EARM1.4 model, these differences do not result from differences in upstream protein levels but rather from downstream differences that feed their influence back on upstream processes. The feedback loop is required to preserve synchronous initiator and effector caspase activation in type I cells. Note that using STL was instrumental here. Indeed, because all cells die in these

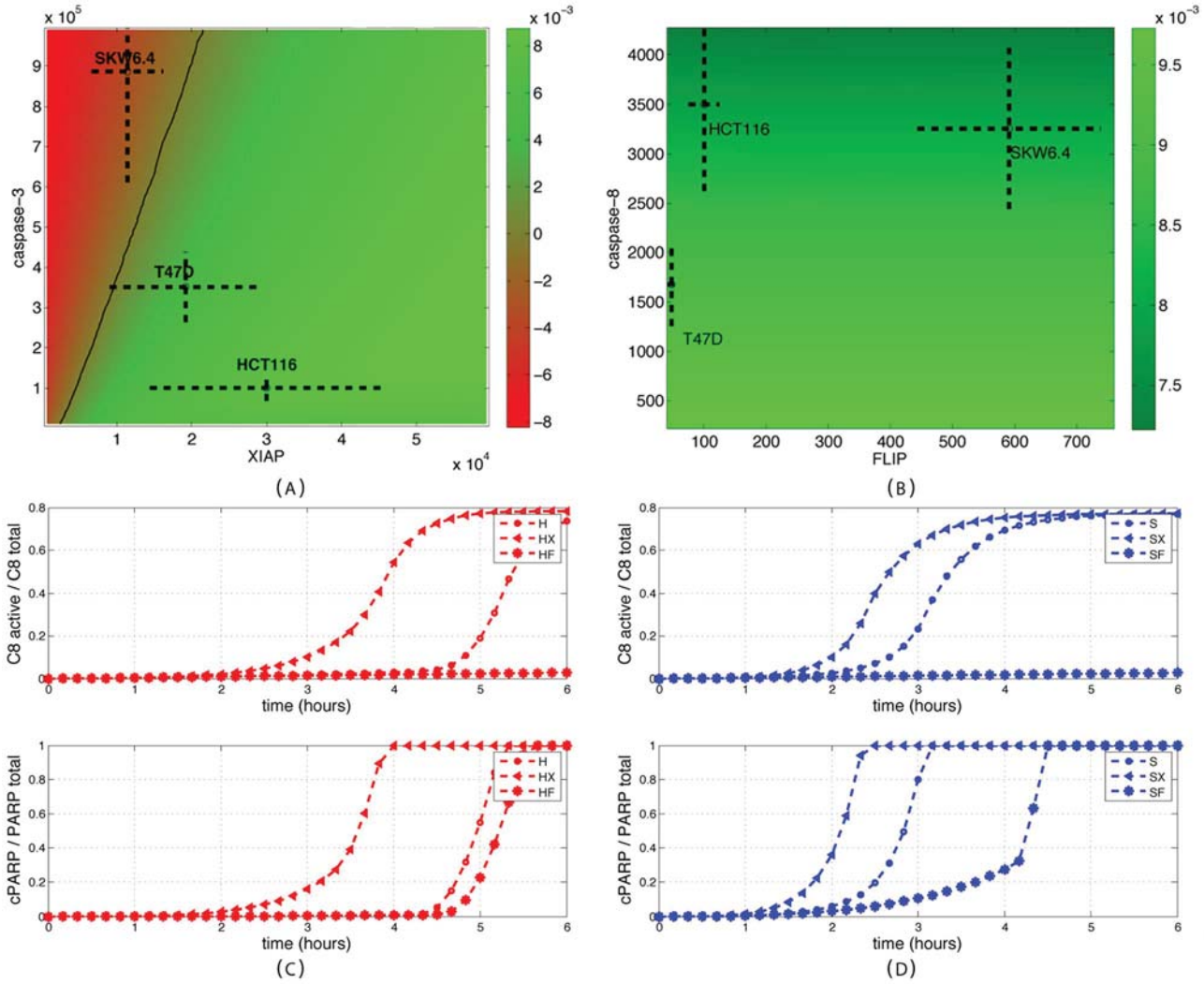


Figure 8. Investigating the role of upstream proteins. (A) XIAP/caspase-3 and (B) FLIP/caspase-8 Property 2 diagrams using HCT116 as reference cell line. Changes in XIAP/caspase-3 levels in HCT116 cells to match levels found in SKW6.4 cells change the original type II phenotype into a type I phenotype. This is not the case for FLIP/caspase-8 changes. (B) The corresponding DLE diagram does not offer intuitive interpretation. (C and D) Temporal evolution of active caspase-8 (top) and cleaved PARP (bottom) in HCT116 cells (red circle), SKW6.4 cells (blue circle), ΔXIAP HCT116 cells (red triangle), ΔXIAP SKW6.4 cells (blue triangle), ΔFB HCT116 (red star) and ΔFB SKW6.4 cells (blue star). The comparison of HCT116 and SKW6.4 cell lines with their ΔXIAP mutant shows important differences in the caspase-8 activation profile. Downstream proteins change upstream protein activation.
doi:10.1371/journal.pcbi.1003056.g008

simulations (no Bcl2 overexpression), DLE diagrams were not offering relevant information.

Interestingly, the analysis of the FLIP/caspase-8 STL diagrams for Property 2 and 3 reveals that moderate inhibition of caspase-8 levels (e.g., by one third) in SKW6.4 cells would transform them into cells showing mixed type behaviors (Figure S7 D and F). Indeed the model predicts that these cells would present a sequential activation of caspases (Property 2 satisfied; a type II feature) and a MOMP-independent death (Property 3 violated; a type I feature). This mutant would show exactly the opposite behavior of ΔXIAP HCT116 cells, a combination of behaviors that has not yet been observed. Therefore, the detailed analysis of this cell line could possibly provide valuable information on the interplay between the two apoptotic pathways. Similarly, the partial inhibition of caspase-3 levels in SKW6.4 cells would also lead to cells showing mixed type behaviors (Property 1 remains false whereas Property 2 and 3 change to true; Figure S6 B, D, and F).

Discussion

In this work, we expressed in a formal language, STL, a number of observed properties on molecular details of extrinsic apoptosis in several mammalian cell lines, and systematically tested their consistency with a previously-proposed model developed to capture the same process in the same cell lines, EARM1.4. It is important to note that even if model and experimental data have been published in the same article [6], the model has not been tuned to comply with the various observed properties we tested on the different cell lines. Indeed, we found several inconsistencies between model predictions and experimental observations. These inconsistencies can be resolved by model reparametrization involving a limited number of parameter changes. However, these needed changes were affecting key processes, namely the PARP cleavage rate and the strength of the caspases-3, -6 and -8 feedback loop. It is remarkable that the model was able to explain a number

of experiments probing different aspect of apoptosis made on different cell lines and mutants, simply by taking into account observed differences in protein concentrations but keeping the same model structure and reaction rates for all cell lines. This makes it a valuable tool to investigate the *origins* of the two different cell responses. Unlike in *in vivo* experiments, the number of factors that could explain these differences is limited in EARM1.4. Using STL diagrams, we showed that observed differences in the concentrations of upstream proteins in different cell lines could not account for the observed cell type changes. This finding is consistent with the observation based on *in vivo* and *in silico* works that downstream proteins, most notably XIAP and caspase-3, play a key role [6], but is in apparent contradiction with the observation that upstream protein activities are markedly different in type I and II cell lines [5]. Detailed analysis showed that the effects of downstream protein concentration differences are in fact fed back to upstream processes and amplified via the positive feedback loop involving caspases 3, 6, and 8. This finding reconciliates the views expressed by Scaffidi and colleagues and by Aldridge and colleagues [5,6].

Based on experimental observations, we defined three properties associated with type II behaviors: (*1*) the cell survives if Bcl2 is over-expressed, (*2*) the activations of initiator and effector caspases are sequential, and (*3*) MOMP precedes caspase-3 activation. They all assess the role of mitochondria for cell death and differ only in subtle means. However, they are not always equivalent. For example, ΔXIAP HCT116 cells satisfy Property 3 but not Property 2, leading to interpretations like ΔXIAP HCT116 being type I cells while exhibiting a type II phenotype. Based on our work, there is no evidence that one property could be considered as a defining criterion, excepted maybe for historical or practical reasons (cell types were originally defined based on caspase activation kinetics whereas Bcl2 overexpression is considered as the standard method for cell type classification). This challenges the consensual understanding that there exists (implicitly) well-defined type I and type II phenotypes. It should be noted that here we go beyond the notion of mixed cell type introduced by Aldridge and colleagues for describing T47D cells. The authors implicitly assume that cell types are well defined but that within a population of cells a mixture of both phenotypes can be observed, coming from cell-to-cell variability [16,44,45]. Here we propose that these three properties are considered as *type II features*. Then the ΔXIAP HCT116 cells would be more consistently qualified as possessing some type I and some type II features. With the accumulation of more detailed characterizations of apoptosis in more cell lines, it is likely that the use of the loosely-defined notion of cell types will otherwise become more and more problematic.

Like the DLE diagrams introduced by Aldridge and Haller [22], STL diagrams are a convenient and intuitive way to represent the influence of various factors on complex dynamical properties. However, STL diagrams are superior on several counts. Firstly, one can benefit from the expressive power of temporal logics to express different observed properties of the dynamics of the cell response. It allows us to test in which respect are the cell lines different. Secondly, although the evaluation of STL properties and of the DLE returns continuous values, the fact that STL values are signed – positive values indicate satisfaction and negative values indicate falsity – allows for a more direct interpretation of the diagrams. Moreover, it allows defining statistics over populations of cells. Thirdly, DLE generates well-defined partitions if in some regions a small change in the initial state has a mild effect on the future system's state, thus generating low DLE values, and in other regions, similar changes have drastic effects, thus generating high DLE values. Although this is clearly the case in cell lines overexpressing Bcl2 since some cells die, whereas others survive (Figure 3), this is not generally true.

DLE and STL diagrams are particularly useful to have a rapid view of the consequences of changing a few factors, initial concentrations in our case. This feature allows us to foresee the consequences of mutations (e.g. ΔXIAP mutants in XIAP/caspase-3 diagrams), to investigate the (lack of) influence of given factors (e.g. FLIP changes in FLIP/caspase-8 diagrams), and to assess the influence of cell-to-cell heterogeneity by representing graphically the means and standard values of populations in diagrams. However, heterogeneity in diagrams is limited to two dimensions. Moreover, since the cell lines differ in more than two dimensions, only one cell line can be correctly mapped in the state space slice of the diagram. Other cell lines are projected onto it, making their interpretations subject to caution. To solve this issue, we introduced population property values for describing the behavior of cell populations. These values and their statistics, notably means, standard deviations, and percentage of satisfaction, offer a more accurate view than phase diagrams. Indeed, even if we found that the rapid picture offered by STL diagrams are often consistent with population property values, a few cases illustrated the need to compute these statistics as well (e.g. T47D cells manifesting *in silico* a clear mixed-type behavior with respect to Property 1, that is not present in the phase diagram in Figure 3).

In addition to computing diagram and population statistics, STL properties also enable model revision based on *experimental observations*. Observed properties are encoded in STL and the continuous semantics of STL is used to search for valid parameter values. Traditional model revision methods based on curve fitting could not be adapted here by lack of well-defined time series data. The non-standard use of continuous semantics for temporal logic formula interpretation is essential to allow for an effective search [46–48]. Using global optimization methods, we found that the few discrepancies we had identified in earlier steps can be resolved by modifying only a restricted set of model parameters. Remarkably, one of the two selected parameters is regulating the strength of the caspases feedback loop, a process that is predicted to play an important role in the genesis of type I or type II behaviors.

The development of experimental methods to probe quantitatively subtle aspects of the dynamics of biological processes has spurred the development of large and complex quantitative models [49,50]. However the available experimental data is seldom in the form of time series data directly usable by standard model validation and model calibration techniques. Therefore tools allowing for the exploration of model properties, the comparison between predictions and observations and the revision of models that are adapted to the available experimental data are increasingly needed. Temporal logics offer a flexible means to encode for a broad range of experimentally-observed properties. Moreover they are also formal languages that allow automating model analysis. Because it supports STL and uses by default distributions for parameter and initial concentrations, Breach naturally allows the exploration of properties of cell populations. We expect that Breach will become a valuable tool for the computational biologists to explore model properties, and more importantly, to get tight connections between experimental data and model predictions [51].

Methods

Modeling extrinsic apoptosis and cell line differences

We used the model of extrinsic apoptosis proposed by Aldridge and colleagues [6] named EARM1.4. This model is

an extension and adaptation of a previous model, EARM1.0, proposed in [15]. EARM1.0 has been calibrated on HeLa cells using live and fixed cell imaging, flow cytometry of caspases substrates and biochemical analysis. EARM1.4 has been adapted to HCT116, SKW6.4 and T47D cells, and has been shown to capture their capacity to die or survive in OE-Bcl2 clonogenic experiments. It is a mass-action ODE model based on nearly 70 reactions and involving 17 native proteins, 40 modified proteins or protein complexes, and TRAIL. For each cell line, the model assumes different nominal initial protein concentrations. Nominal concentrations refer here to concentrations found in a hypothetical mean cell within the cell population. More precisely, out of the 17 native proteins, 12 have been quantified by immunoblotting and the relative differences between cell lines have been used to set nominal initial protein concentrations for HCT116, SKW6.4 and T47D cells (see Table 1). Besides initial concentrations, the 3 models are identical. ΔXIAP and OE-Bcl2 mutant cell lines are defined with respect to their parent cell line. In ΔXIAP cells, the XIAP concentration is set to 0. In OE-Bcl2 cells, the initial Bcl2 concentration is 10 times higher than in the parent cell line. For cells with modified feedback (ΔFB cells) we set the cleavage rate of caspase 8 by caspase 6, k7, to 0. To represent cell-to-cell variability within cell lines, we assumed that protein concentrations are log-normally distributed. The means of protein concentrations were the nominal values. The coefficient of variation were either measured, for caspase-3 and XIAP [6], or assumed to be 25% as in [16]. The complete model together with Breach is available in Supplementary Materials as MATLAB files (S9). The names of the variables, constants and reactions used in the model are the same as in [6].

STL semantics and property evaluation

STL is an intuitive yet formal language for specifying the properties of continuous dynamical systems. It allows us to express in a (pseudo-) natural language hypothesis on the mechanistic functioning of the system taken from available biological knowledge in a formal way so that model consistency can be precisely and systematically tested. Given a model of the system, expected properties are expressed using predicates describing constraints on protein concentrations, like $cPARP<10^5$, traditional logical operators, like *and*, *or* and *implies*, and temporal operators, like $eventually_{[a,b]}$, $always_{[a,b]}$, and $until_{[a,b]}$. Time intervals [a,b] limit the scope of temporal operators. These operators can be combined to create properties of arbitrary complexities. For example, $always_{[0-6h]}$ $(XIAP>10^3$ *and* $cPARP<10^5)$ is a valid STL formula. The formal syntax is given in Table S2 (top). STL properties are then interpreted with respect to so-called signals. In this context, signals are functions from time to the reals representing the evolution of the different concentrations in the system. Computationaly speaking, they often come from (discrete) time-series data obtained by numerical simulation of the ODE model. The semantics is defined such that it captures a notion of distance from satisfaction. For example, the interpretation at time t of the predicate $XIAP>10^3$ is simply the value of $XIAP(t) - 10^3$. Trivially it is positive if $XIAP>10^3$, and negative if $XIAP<10^3$. The interpretation of $XIAP>10^3$ *and* $cPARP<10^5$ at time t is the minimum of the value of the two operands at time t. Note that it is positive if and only if both operands have positive values. Similarly, the interpretation of $always(XIAP>10^3)$ is the minimum of the value of $XIAP>10^3$ at all future time instants. It is positive if $XIAP$ is always greater than 10^3. The interpretation of STL formulas is also illustrated on Figure 9. More generally, the continuous interpretation of STL properties ensures that if the value of a property is positive (resp. negative), then the property holds (resp. is violated) in a more usual Boolean interpretation. Moreover, it captures a notion of "distance from satisfaction": a large positive value indicates a robustly satisfied property, whereas a large negative value indicates a property that is far from satisfaction. The semantics is formally defined in Table S2 (bottom). Note that property values are relative to the formula, in the sense that values obtained for different STL formulas are not directly comparable between each other.

Table 1. Initial concentrations of proteins in HCT116, SKW6.4, and T47D cells.

Protein\Cell line	HCT116	SKW6.4	T47D
FLIP	100 (0.25)	591 (0.25)	48 (0.25)
caspase-8	3500 (0.25)	3255 (0.25)	1680 (0.25)
caspase-6	10000 (0.25)	6700 (0.25)	22500 (0.25)
caspase-3	100000 (**0.32**)	886000 (**0.31**)	351000 (**0.25**)
XIAP	30000 (**0.52**)	11400 (**0.42**)	19200 (**0.53**)
PARP	1000000 (0.25)	1120000 (0.25)	1040000 (0.25)
Bid	40000 (0.25)	74800 (0.25)	53600 (0.25)
Mcl1	1000 (0.25)	1250 (0.25)	4640 (0.25)
Bax	80000 (0.25)	786400 (0.25)	113600 (0.25)
Bcl2	20000 (0.25)	400000 (0.25)	104000 (0.25)
Smac	100000 (0.25)	177000 (0.25)	139000 (0.25)

Nominal values and coefficients of variations for initial protein concentrations that differ between cell lines (see [6] for concentrations of other proteins). Protein concentrations are assumed log-normally distributed across the cell populations.

doi:10.1371/journal.pcbi.1003056.t001

Computation of property diagrams

Given an STL property, the associated STL phase diagram is a representation of the value of the property as a function of the system's initial configuration. More precisely diagrams represent property values in 2D slices of a high dimensional state space. Each point in the diagram is associated to the value of the STL formula evaluated on the system's trajectory starting at this point. Boundaries were set so as to enclose the variability observed between cell lines. Diagrams are defined with respect to a particular cell line: with the exception of the two variables of the diagram, all other variables assume their nominal values for the given cell line. Other cell lines are placed on the diagram based on the initial concentrations of the selected proteins. Unless mentioned otherwise, the HCT116 cell line is used as a reference. In practice, a 50×50 grid of linearly-spaced points is used for the computation of each diagram. For each point on the grid, we computed the cell behavior predicted by the model and then the value of the STL property associated with this behavior (see Program S1). The ranges for caspase-3, XIAP, caspase-8 and FLIP are [0, 10^6], [0, $6*10^4$], [1200, 4500] and [0, 800], respectively. Similarly, DLE diagrams represent the direct finite-time Lyapunov exponent for all points in (a 2D slice of) the state space. This value captures the sensitivity to initial conditions: $DLE(t,x_0) = \log \lambda_{\max}((\frac{\partial x(t)}{\partial x_0})^T(\frac{\partial x(t)}{\partial x_0}))$, where $\lambda_{\max}(M)$ denotes the maximum eigenvalue of the matrix M and t is some future time instant (here 6 or 4 hours). Again, a 50×50 grid of linearly-spaced points is used for the computation of DLE diagrams.

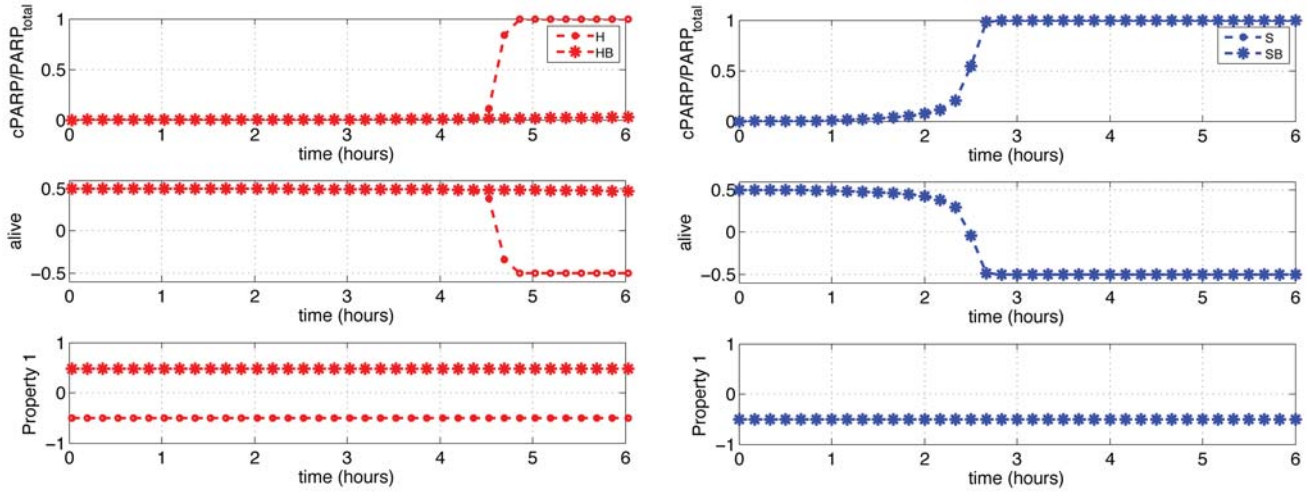


Figure 9. Evaluation of STL properties. Temporal evolution of the ratio *cPARP/PARP$_{total}$*, of the value of the property *alive*: = *cPARP/PARP$_{total}$*<*0.5*, and of *Property 1*: = *always$_{[0-6h]}$*(*cPARP/PARP$_{total}$*<*0.5*) for HCT116 and OE-Bcl2 HCT116 cells (left), and SKW6.4 and OE-Bcl2 SKW6.4 cells (right). When the concentration of cleaved PARP increases, the value of the *alive* property gradually decreases from a positive value ("true") to a negative value ("false"). *Property 1* at time *t* evaluates to the minimal value of *alive* at all future times. So, Property 1 simply captures whether at all times *alive* holds.
doi:10.1371/journal.pcbi.1003056.g009

Computation of STL population data

Given an STL property, the STL population data correspond to the evaluation of this property on all the simulated individual cell behaviors among a population of cells of a given cell line. Based on these property values, statistics are computed. For STL population data, 5000 different initial conditions are obtained for each cell line by sampling around its nominal initial conditions from lognormal distributions. Mean values, value distributions and percentages of satisfaction of the property (i.e. the percentage of cells in the population satisfying a given property) are then computed.

Parameter search procedure

The search procedure has two phases. In the first phase we search for new parameters for EARM1.4 that lead to full agreement with experimental data (Figure 7). In the second phase, when a solution is found, we minimize the number of modified parameters. We use a cost function composed of three different components: continuous, Boolean, and parameter penalties. The continuous penalties correspond to the (negation of) the continuous values of STL properties, and the Boolean penalties correspond to their Boolean value multiplied by a (negative) constant. These costs decrease when more properties are consistent with observations ($B_{penalty}$), and when they are more robustly consistent with observations ($C_{penalty}$). In the continuous component, weights are used to balance the importance of all properties, given their typical range. The last component penalizes parameter deviations from their original values ($P_{penalty}$). The overall cost is the weighted sum of these three components.

$$cost(p) := \alpha\, P_{penalty}(p) + \sum_{i \in CellLines} \sum_{\pi \in Properties} (\beta\, B_{penalty}(p,i,\pi) + \gamma\, C_{penalty}(p,i,\pi))$$

In the first phase, we selected 43 parameters (14 catalytic rates of enzymatic reactions and 29 forward rates) out of approximately 80 parameters in EARM1.4. Parameter modifications were limited to a 100-fold change. We set weights so that the Boolean, continuous, and parameter penalties contributed to approximately 50%, 30%, and 20% of the cost, respectively. After 10 hours of computations (2.2 GHz processor, 8GB RAM), the search converged to a state in which all expected properties were satisfied by the model (T47D cells excluded).

In the second phase, we selected the parameters that changed by more than 5 folds (there were 5 such parameters: kc9, kc25, kc20, k7 and k24) and run the search again for each pair of these parameters. The cost function was modified by setting the $C_{penalty}$ parameter to 0, and the beta parameter such as the Boolean penalty was responsible for approximately 90% of the cost. As a result, parameter deviations were minimized while preserving the agreement with the experimental data. We found that reparametrization of only one pair of parameters allowed for satisfaction of all properties for all cell lines.

Breach tool

All the computations have been made using Breach [33,48]. This MATLAB/C++ toolbox allows for efficient numerical simulation, for sensitivity computation, and for STL property and DLE evaluation. In particular, DLEs can efficiently be computed via forward sensitivity analysis [52]. Breach is particularly oriented towards the analysis of parametric systems, in the sense that it offers efficient routines for global sensitivity analysis and parameter search, and that the graphical user interface facilitates the modification of parameters and initial conditions, and the exploration of parameter spaces.

Supporting Information

Figure S1 Formula robustness. Number of matches between predicted and observed satisfaction values for Properties 1–3 in all HCT116 and SKW6.4 cell lines (Figure 7) as a function of the PARP-related threshold, α, defining the *alive* property, of the Apaf-related threshold, β, defining the *MOMP* occurrence and of the caspase-related threshold, γ, defining caspases activation when (A) α and β vary, and γ is fixed, or (B) γ varies, and α and β are fixed. Thresholds α, β, and γ *are defined as follows: p1: = always$_{[0-6h]}$(cPARP/PARP$_{total}$*<*α); p2: = eventually(Casp8$_{active}$ and always$_{[0-1h]}$ not Casp3$_{active}$); p3: = Apaf$_{free}$/Apaf$_{total}$*<*β release (cPARP/PARP$_{total}$*<*α)), where Casp8$_{active}$: = Casp8*/Casp8$_{total}$*>*γ and Casp3$_{active}$: = Casp3*/Casp3$_{total}$*>*γ*. Full consistency

with all experimental data corresponds to 16 matches (15 in Figure 7 and, additionally, p2(SKW6.4) = True). For original properties ($\alpha = \beta = 50\%$ and $\gamma = 1\%$), we found three mismatches (Figure 7). This number is robust with respect to changes of the PARP-related threshold, α, and of the Apaf-related threshold, β. It is also robust to the caspase-related threshold, γ, provided that this value remains low enough (i.e. $<2\%$).
(TIF)

Figure S2 Comparison between DLE and Property 1 STL diagrams. Diagrams representing the values of the DLE computed at time T (A,C) and of the STL Property: = $always_{[0\text{-}T]}(cPARP/PARP_{total}<0.5)$ (B,D) for T = 6 h (A–B) and T = 4 h (C–D). Strikingly, for the two time instants the separatrix is exactly at the same position, revealing that DLE and Property 1 capture precisely the same behavior: the existence of two different possible outcomes: survival or death. However, in full generality the DLE simply measures the influence of small changes in initial protein concentrations on the future state of the system. In fact, this similarity comes from the *snap-action* aspect of apoptotic cell death, captured by the EARM model: cell death is immediately preceded by a sudden activation of effector caspases (all-or-none behavior) [15]. Small changes in initial protein concentrations will result in dramatic differences in protein concentrations at the time of death and therefore in high DLE values. One should also note that the interpretation of low DLE values is ambiguous, since low values are found in regions corresponding to type I (SKW6.4) and to type II cell types (HCT116).
(TIF)

Figure S3 XIAP/capsase-3 STL diagrams for all properties and using HCT116, SKW6.4 or T47D as reference cell line. Diagrams representing the values of the STL properties p1 (A–C), p2 (D–F) and p3 (G–H) computed using HCT116 (A,D,G), SKW6.4 (B,E,H), or T47D (C,F,I) nominal protein concentrations. Bcl2 is overexpressed in Property 1 diagrams. In most cases, for a given property the satisfaction values associated with each cell type is similar irrespectively of the reference cell line used to construct the diagram. However, there are exceptions, like in the case of T47D cell line behavior (H and I). So care must be taken when interpreting STL diagrams. The same situation holds with DLE diagrams (not shown).
(TIF)

Figure S4 STL property values across all cell lines for Properties 1–3 for the EARM1.4. For each property, plots indicate the nominal cell value (top), the distribution (middle), and the percentage of satisfaction (bottom) of the property values for populations of cells of different cell lines. Notations are identical to those used in Figure 4.
(TIF)

Figure S5 Population statistics for Property 1, 2 and 3, computed with new parameter values. (see Table S1) This data should be compared with Figure 4, 5 (right), and 6 (right). The new parameter values allow resolving the inconsistencies found for SKW6.4, OEBcl2 SKW6.4 cells for Property 2, and for ΔXIAP HCT116 cells for Property 3. T47D cells still do not satisfy Property 3 as expected. Notations are identical to those used in Figure 4.
(TIF)

Figure S6 XIAP/Capsase-3 STL diagrams computed with new parameter values for all properties and using HCT116 or SKW6.4 as reference cell lines. Diagrams representing the values of the STL properties p1 (A–B), p2 (C–D) and p3 (E–F), computed using HCT116 (A,C,E) or SKW6.4 (B,D,F) nominal protein concentrations.
(TIF)

Figure S7 FLIP/Capsase-8 STL diagrams computed with new parameter values for all properties and using HCT116 or SKW6.4 as reference cell lines. Diagrams representing the values of the STL properties p1 (A–B), p2 (C–D) and p3 (E–F), computed using HCT116 (A,C,E) or SKW6.4 (B,D,F) nominal protein concentrations.
(TIF)

Program S1 Computation of STL diagrams using Breach [33]. The archive contains the freely-distributed Matlab tool Breach, an implementation of EARM1.4 in Breach, initial conditions for each of 12 cell lines used in this article, and example scripts illustrating how to generate STL phase diagrams.
(ZIP)

Table S1 Valid parameters. List of minimal parameter set leading to Property1–3 satisfaction for all but T47D cells, together with their new and original values, and the corresponding fold change.
(TIF)

Table S2 Syntax and semantics of STL [48]. The syntax of STL formulas is defined inductively. Here, ϕ, ϕ_1, ϕ_2 are STL formulas, $\mu(x)$ is an equality of type $f(x(t)) > 0$, with f a real-valued function on the state x, and $[a,b]$ is a time interval. The real-valued semantics $\rho(\phi,t)$ of an STL formula φ at time t is interpreted on a real-valued signal $x(t)$ defined on a time interval $[0,T_f]$, where T_f is typically the end time of a simulation. One additionally defines $\phi_1 \; or \; \phi_2$ as $not(not\,\phi_1 \; and \; not\,\phi_2)$, $eventually_{[a,b]}\,\phi$ as $\mathrm{True}\; until_{[a,b]}\,\phi$ and $always_{[a,b]}\,\phi$ as $not\; eventually_{[a,b]}\; not\,\phi$.
(TIF)

Acknowledgments

We thank Denis Thieffry, Magdalena Stepien and Xavier Duportet for their critical suggestions. We are grateful for reviewers' comments which significantly helped to improve our manuscript.

Author Contributions

Conceived and designed the experiments: SS AD FB GB. Performed the experiments: SS AD GB. Analyzed the data: SS AD FB OM GB. Contributed reagents/materials/analysis tools: SS AD. Wrote the paper: SS GB.

References

1. Gonzalvez F, Ashkenazi A (2010) New insights into apoptosis signaling by Apo2L/TRAIL. Oncogene 29: 4752–4765.
2. Spencer SL, Sorger PK (2011) Measuring and modeling apoptosis in single cells. Cell 144: 926–939.
3. Kasibhatla S, Tseng B (2003) Why Target Apoptosis in Cancer Treatment? Molecular Cancer Therapeutics 2: 573–580.
4. Jost PJ, Grabow S, Gray D, McKenzie MD, Nachbur U, et al. (2009) XIAP discriminates between type I and type II FAS-induced apoptosis. Nature 460: 1035–1039.
5. Scaffidi C, Fulda S, Srinivasan A, Friesen C, Li F, et al. (1998) Two CD95 (APO-1/Fas) signaling pathways. The EMBO Journal 17: 1675–1687.
6. Aldridge BB, Gaudet S, Lauffenburger DA, Sorger PK (2011) Lyapunov exponents and phase diagrams reveal multi-factorial control over TRAIL-induced apoptosis. Molecular Systems Biology 7: 553.
7. Eissing T, Conzelmann H, Gilles ED, Allgöwer F, Bullinger E, et al. (2004) Bistability analyses of a caspase activation model for receptor-induced apoptosis. The Journal of Biological Chemistry 279: 36892–36897.

8. Hoffmann A, Levchenko A, Scott ML, Baltimore D (2002) The IkappaB-NF-kappaB signaling module: temporal control and selective gene activation. Science 298: 1241–1245.
9. Bentele M, Lavrik I, Ulrich M, Stösser S, Heermann DW, et al. (2004) Mathematical modeling reveals threshold mechanism in CD95-induced apoptosis. The Journal of Cell Biology 166: 839–851.
10. Hua F, Cornejo MG, Cardone MH, Stokes CL, Lauffenburger DA (2005) Effects of Bcl-2 levels on Fas signaling-induced caspase-3 activation: molecular genetic tests of computational model predictions. Journal of Immunology 175: 985–995.
11. Bagci EZ, Vodovotz Y, Billiar TR, Ermentrout GB, Bahar I (2006) Bistability in apoptosis: roles of Bax, Bcl-2, and mitochondrial permeability transition pores. Biophysical Journal 90: 1546–1559.
12. Legewie S, Blüthgen N, Herzel H (2006) Mathematical modeling identifies inhibitors of apoptosis as mediators of positive feedback and bistability. PLoS Computational Biology 2: e120.
13. Rehm M, Huber HJ, Dussmann H, Prehn JHM (2006) Systems analysis of effector caspase activation and its control by X-linked inhibitor of apoptosis protein. The EMBO Journal 25: 4338–4349.
14. Chen C, Cui J, Lu H, Wang R, Zhang S, et al. (2007) Modeling of the Role of a Bax-Activation Switch in the Mitochondrial Apoptosis Decision. Biophysical Journal 92: 4304–4315.
15. Albeck JG, Burke JM, Spencer SL, Lauffenburger DA, Sorger PK (2008) Modeling a snap-action, variable-delay switch controlling extrinsic cell death. PLoS Biology 6: 2831–2852.
16. Spencer SL, Gaudet S, Albeck JG, Burke JM, Sorger PK (2009) Non-genetic origins of cell-to-cell variability in TRAIL-induced apoptosis. Nature 459: 428–432.
17. Fricker N, Beaudouin J, Richter P, Eils R, Krammer PH, et al. (2010) Model-based dissection of CD95 signaling dynamics reveals both a pro- and antiapoptotic role of c-FLIPL. The Journal of Cell Biology 190: 377–389.
18. Mai Z, Liu H (2009) Boolean network-based analysis of the apoptosis network: irreversible apoptosis and stable surviving. Journal of Theoretical Biology 259: 760–769.
19. Calzone L, Tournier L, Fourquet S, Thieffry D, Zhivotovsky B, et al. (2010) Mathematical modelling of cell-fate decision in response to death receptor engagement. PLoS Computational Biology 6: e1000702.
20. Saez-Rodriguez J, Alexopoulos LG, Zhang M, Morris MK, Lauffenburger DA, et al. (2011) Comparing signaling networks between normal and transformed hepatocytes using discrete logical models. Cancer Research 71: 5400–5411.
21. Schlatter R, Schmich K, Avalos Vizcarra I, Scheurich P, Sauter T, et al. (2009) ON/OFF and beyond–a boolean model of apoptosis. PLoS Computational Biology 5: e1000595.
22. Aldridge B, Haller G (2006) Direct Lyapunov exponent analysis enables parametric study of transient signalling governing cell behaviour. Systems Biology, IEE Proceedings 153: 425–432.
23. Maler O, Nickovic D (2004) Monitoring Temporal Properties of Continuous Signals. Proceedings of the Formal Techniques, Modelling and Analysis in Real-Time and Fault-Tolerant Systems Conference, FORMATS/FTRTFT 2004, LNCS 3253. Springer-Verlag. pp. 152–166.
24. Barnat J, Brim L, Cerna I, Drasan S, Safranek D (2008) Parallel Model Checking Large-Scale Genetic Regulatory Networks with DiVinE. Electronic Notes in Theoretical Computer Science. Springer-Verlag, Vol. 194. pp. 35–50.
25. Batt G, Yordanov B, Weiss R, Belta C (2007) Robustness analysis and tuning of synthetic gene networks. Bioinformatics 23: 2415–2422.
26. Batt G, Page M, Cantone I, Goessler G, Monteiro P, et al. (2010) Efficient parameter search for qualitative models of regulatory networks using symbolic model checking. Bioinformatics 26: i603–10.
27. Bernot G, Comet J-P, Richard A, Guespin J (2004) Application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic. Journal of Theoretical Biology 229: 339–347.
28. Donaldson R, Gilbert D (2008) A model checking approach to the parameter estimation of biochemical pathways. In: Proceedings of the 6th International Conference on Computational Methods in Systems Biology, CMSB 2008, LNCS 5307. Berlin: Springer-Verlag. pp. 269–287.
29. Fisher J, Piterman N, Hajnal A, Henzinger T a (2007) Predictive modeling of signaling crosstalk during C. elegans vulval development. PLoS Computational Biology 3: e92.
30. Jha SK, Clarke EM, Langmead CJ, Legay A, Platzer A, et al. (2009) A bayesian approach to model checking biological systems. In: Proceedings of the 7th International Conference on Computational Methods in Systems Biology, CMSB 2009, LNCS 5688. Berlin: Springer-Verlag. pp. 218–234.
31. Heath J, Kwiatkowska M, Norman G, Parker D, Tymchyshyn O (2008) Probabilistic model checking of complex biological pathways. Theoretical Computer Science 391: 239–257.
32. Donzé A, Fanchon E, Gattepaille LM, Maler O, Tracqui P (2011) Robustness analysis and behavior discrimination in enzymatic reaction networks. PloS One 6: e24246.
33. Donzé A (2010) Breach, a toolbox for verification and parameter synthesis of hybrid systems. In: Proceedings of the 22nd International Conference on Computer Aided Verification, CAV'10, LNCS 6174. Berlin: Springer-Verlag. pp. 167–170.
34. Barnhart BC, Alappat EC, Peter ME (2003) The CD95 Type I/Type II model. Seminars in Immunology 15: 185–193.
35. Algeciras-Schimnich A, Pietras EM, Barnhart BC, Legembre P, Vijayan S, et al. (2003) Two CD95 tumor classes with different sensitivities to antitumor drugs. Proceedings of the National Academy of Sciences of the United States of America 100: 11445–11450.
36. Ozören N, El-Deiry WS (2002) Defining characteristics of Types I and II apoptotic cells in response to TRAIL. Neoplasia 4: 551–557.
37. Rehm M, Dussmann H, Janicke RU, Tavare JM, Kogel D, et al. (2002) Single-cell fluorescence resonance energy transfer analysis demonstrates that caspase activation during apoptosis is a rapid process. Role of caspase-3. The Journal of Biological Chemistry 277: 24506–24514.
38. Albeck JG, Burke JM, Aldridge BB, Zhang M, Lauffenburger DA, et al. (2008) Quantitative analysis of pathways controlling extrinsic apoptosis in single cells. Molecular Cell 30: 11–25.
39. Agard NJ, Mahrus S, Trinidad JC, Lynn A, Burlingame AL, et al. (2012) Global kinetic analysis of proteolysis via quantitative targeted proteomics. Proceedings of the National Academy of Sciences of the United States of America 109: 1913–1918.
40. Maas C, Verbrugge I, De Vries E, Savich G, Van de Kooij LW, et al. (2010) Smac/DIABLO release from mitochondria and XIAP inhibition are essential to limit clonogenicity of Type I tumor cells after TRAIL receptor stimulation. Cell Death and Differentiation 17: 1613–1623.
41. Avriel M (2003) Nonlinear programming: analysis and methods. Courier Dover Publications.
42. Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation 9: 159–195.
43. Kober a MM, Legewie S, Pforr C, Fricker N, Eils R, et al. (2011) Caspase-8 activity has an essential role in CD95/Fas-mediated MAPK activation. Cell Death and Disease 2: e212.
44. Bhola PD, Simon SM (2009) Determinism and divergence of apoptosis susceptibility in mammalian cells. Journal of Cell Science 122: 4296–4302.
45. Rehm M, Huber HJ, Hellwig CT, Anguissola S, Dussmann H, et al. (2009) Dynamics of outer mitochondrial membrane permeabilization during apoptosis. Cell Death and Differentiation 16: 613–623.
46. Fainekos GE, Pappas GJ (2009) Robustness of temporal logic specifications for continuous-time signals. Theoretical Computer Science 410: 4262–4291.
47. Rizk A, Batt G, Fages F, Soliman S (2011) Continuous valuations of temporal logic specifications with applications to parameter optimization and robustness measures. Theoretical Computer Science 412: 2827–2839.
48. Donzé A, Maler O (2010) Robust satisfaction of temporal logic over real-valued signals. Formal Modeling and Analysis of Timed Systems: 92–106.
49. Klipp E, Herwig R, Kowald A, Wierling C, Lehrach H (2005) Systems Biology in Practice. Concepts, Implementation and Application. Wiley-VCH.
50. Szallasi Z, Stelling J, Periwal V (2010) System Modeling in Cellular Biology: From Concepts to Nuts and Bolts. Boston: MIT Press.
51. Ghosh S, Matsuoka Y, Asai Y, Hsin K-Y, Kitano H (2011) Software for systems biology: from tools to integrated platforms. Nature reviews Genetics 12: 821–832.
52. Serban R, Hindmarsh A (2005) CVODES, the sensitivity-enabled ODE solver in SUNDIALS. Proceedings of IDETC/CIE; Sept 2005; Long Beach, California, United States.
53. Peter ME, Krammer PH (2003) The CD95(APO-1/Fas) DISC and beyond. Cell Death and Differentiation 10: 26–35.
54. Huang Y, Park YC, Rich RL, Segal D, Myszka DG, et al. (2001) Structural basis of caspase inhibition by XIAP: differential roles of the linker versus the BIR domain. Cell 104: 781–790.
55. Chen L, Smith L, Wang Z, Smith JB (2003) Preservation of caspase-3 subunits from degradation contributes to apoptosis evoked by lactacystin: any single lysine or lysine pair of the small subunit is sufficient for ubiquitination. Molecular Pharmacology 64: 334–345.
56. Luo X, Budihardjo I, Zou H, Slaughter C, Wang X (1998) Bid, a Bcl2 interacting protein, mediates cytochrome c release from mitochondria in response to activation of cell surface death receptors. Cell 94: 481–490.
57. Kim H, Rafiuddin-Shah M, Tu H-C, Jeffers JR, Zambetti GP, et al. (2006) Hierarchical regulation of mitochondrion-dependent apoptosis by BCL-2 subfamilies. Nature Cell Biology 8: 1348–1358.
58. Du C, Fang M, Li Y, Li L, Wang X (2000) Smac, a mitochondrial protein that promotes cytochrome c-dependent caspase activation by eliminating IAP inhibition. Cell 102: 33–42.

Long-term model predictive control of gene expression at the population and single-cell levels

Jannis Uhlendorf[a,b], Agnès Miermont[b], Thierry Delaveau[c], Gilles Charvin[d], François Fages[a], Samuel Bottani[b], Gregory Batt[a,1,2], and Pascal Hersen[b,e,1,2]

[a]Contraintes Research Group, Institut National de Recherche en Informatique et en Automatique, INRIA Paris-Rocquencourt, 78150 Rocquencourt, France; [b]Laboratoire Matière et Systèmes Complexes, Unité Mixte de Recherche 7057 Centre National de la Recherche Scientifique and Université Paris Diderot, 75013 Paris, France; [c]Laboratoire de Génomique des Microorganismes, Unité Mixte de Recherche 7238 Centre National de la Recherche Scientifique and Université Pierre et Marie Curie, 75006 Paris, France; [d]Institut de Génétique et Biologie Moléculaire et Cellulaire, 67400 Illkirch, France; and [e]Mechanobiology Institute, National University of Singapore, Singapore 117411

Edited by David A. Weitz, Harvard University, Cambridge, MA, and approved July 16, 2012 (received for review April 23, 2012)

Gene expression plays a central role in the orchestration of cellular processes. The use of inducible promoters to change the expression level of a gene from its physiological level has significantly contributed to the understanding of the functioning of regulatory networks. However, from a quantitative point of view, their use is limited to short-term, population-scale studies to average out cell-to-cell variability and gene expression noise and limit the nonpredictable effects of internal feedback loops that may antagonize the inducer action. Here, we show that, by implementing an external feedback loop, one can tightly control the expression of a gene over many cell generations with quantitative accuracy. To reach this goal, we developed a platform for real-time, closed-loop control of gene expression in yeast that integrates microscopy for monitoring gene expression at the cell level, microfluidics to manipulate the cells' environment, and original software for automated imaging, quantification, and model predictive control. By using an endogenous osmostress responsive promoter and playing with the osmolarity of the cells environment, we show that long-term control can, indeed, be achieved for both time-constant and time-varying target profiles at the population and even the single-cell levels. Importantly, we provide evidence that real-time control can dynamically limit the effects of gene expression stochasticity. We anticipate that our method will be useful to quantitatively probe the dynamic properties of cellular processes and drive complex, synthetically engineered networks.

model based control | computational biology | high osmolarity glycerol pathway | quantitative systems biology

Understanding the information processing abilities of biological systems is a central problem for systems and synthetic biology (1–6). The properties of a living system are often inferred from the observation of its response to static perturbations. Time-varying perturbations have the potential to be much more informative regarding the dynamics of cellular functions (7–12). Currently, it is not possible to precisely perturb protein levels in an analogous manner, even though this perturbation would be instrumental in our understanding of gene regulatory networks. Indeed, despite the development of novel regulatory systems, including various RNA-based solutions (13), transcriptional control by means of inducible promoters is still the preferred method for manipulating protein levels (14, 15). Unfortunately, inducible promoters have several generic limitations. First, there is a significant delay between gene expression activation and effective protein synthesis. Second, many cellular processes can interfere with gene expression through internal feedback loops whose effects are hard to predict. Third, the process of gene expression shows significant levels of noise (16–18). Given these limitations, novel experimental strategies are required to gain quantitative, real-time control of gene expression in vivo.

Here, we see the problem of manipulating gene expression to obtain given temporal profiles of protein levels as a model-based control problem. More precisely, we investigate the effectiveness of computerized closed-loop control strategies to control gene expression in vivo. In model-based closed-loop control, a model of the system is used to constantly update the control strategy based on real-time observations. We propose an experimental platform that implements such an in silico closed loop in the budding yeast *Saccharomyces cerevisiae*. We show that gene expression can be controlled by repeatedly stimulating a native endogenous promoter over many cell generations (>15 h) for both time-constant and time-varying target profiles and at both the population and single-cell levels. Recently, Milias-Argeitis et al. (19) also proposed an approach for feedback control of gene expression in yeast. In contrast to their work, we propose a method that is effective at the single-cell level, for time-varying target profiles, and robust despite the presence of strong internal feedback loops. We start by describing the gene induction system and the experimental platform before discussing its efficiency.

Results and Discussion

Controlled System. We based our approach on the well-known response of yeast to an osmotic shock, which is mediated by the high osmolarity glycerol (HOG) signaling cascade. Its activation leads to the phosphorylation of the protein Hog1 (Fig. 1*A*), which orchestrates cell adaptation through glycerol accumulation. Phosphorylated Hog1 promotes glycerol production by activating gene expression in the nucleus as well as stimulating glycerol-producing enzymes in the cytoplasm. After they are adapted, the cells do not sense the hyperosmotic environment anymore, the HOG cascade is turned off, and the transcriptional response stops (20–22). In control terms, yeast cells implement several short-term (non transcriptional) and long-term (transcriptional) negative feedback loops that ensure perfect adaptation to the osmotic stress (10, 23). Because of these adaptation mechanisms, it is a priori challenging to control gene expression induced by osmotic stress. It is, thus, an excellent system to show that one can robustly control protein levels, even in the presence of internal negative feedback loops. Several genes are up-regulated in response to a hyperosmotic stress. These genes include the nonessential gene *STL1*, which codes for a glycerol proton symporter (24, 25). We decided to use its native promoter (pSTL1) to drive the expression of yECitrine, a fluorescent reporter. Applying an osmotic stress transiently activated the HOG cascade (Fig. 1*B*), and yECitrine levels reached modest values (600 fluorescence units) (Fig. 1*B*). Importantly, when short but repeated stresses were applied, pSTL1 could be repeatedly activated, and much higher levels could be reached (Fig. 1*C*).

Author contributions: J.U., G.B., and P.H. designed research; J.U. performed research; J.U., A.M., T.D., G.C., F.F., S.B., G.B., and P.H. contributed new reagents/analytic tools/software; J.U., G.B., and P.H. analyzed data; and J.U., G.B., and P.H. wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

Freely available online through the PNAS open access option.

[1]To whom correspondence may be addressed. E-mail: gregory.batt@inria.fr or pascal.hersen@univ-paris-diderot.fr.

[2]G.B. and P.H. contributed equally to this work.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1206810109/-/DCSupplemental.

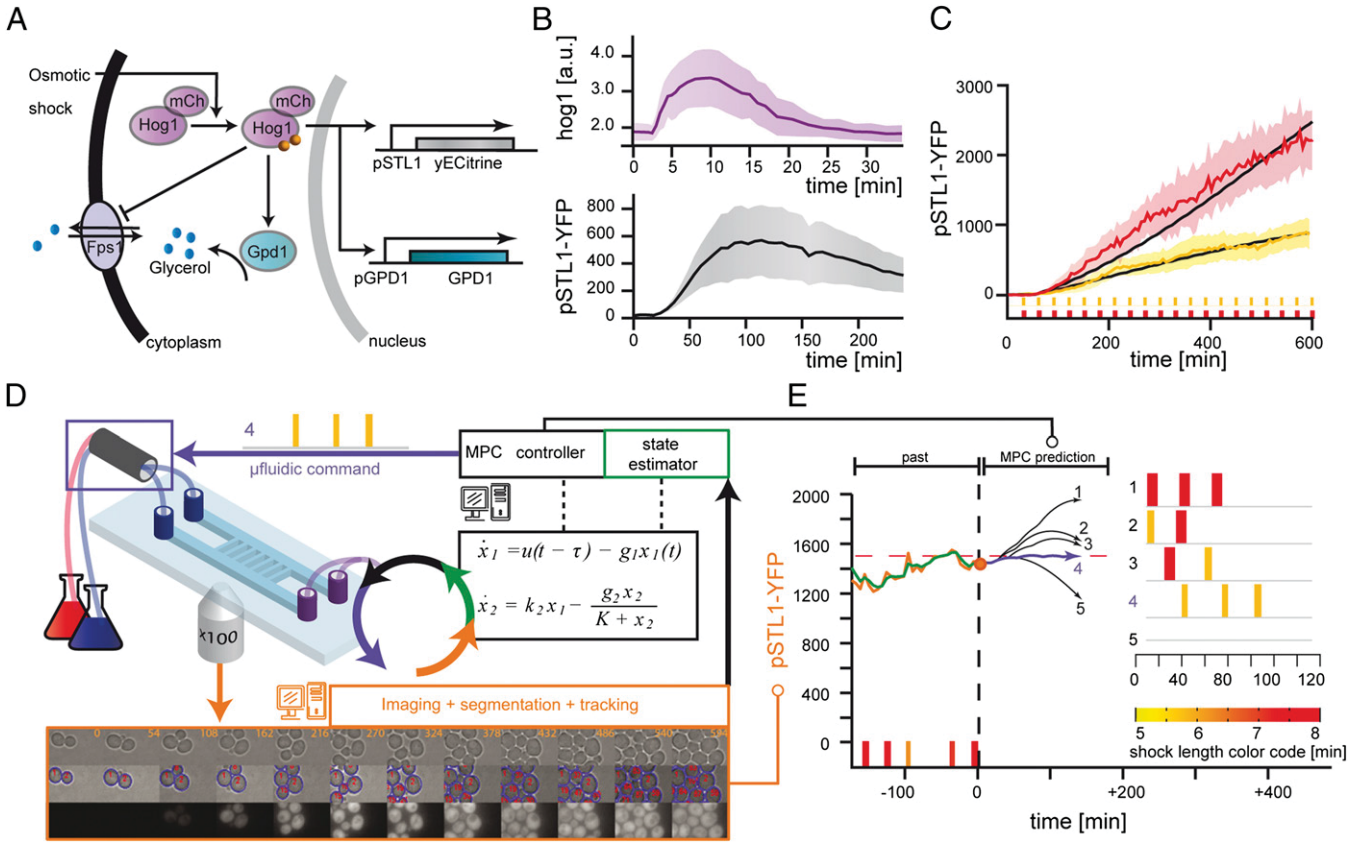


Fig. 1. A platform for real-time control of gene expression in yeast. (*A*) A hyperosmotic stress triggers the activation and nuclear translocation of Hog1. Short-term adaptation is mainly implemented by cytoplasmic activation of the glycerol-producing enzyme Gpd1 and closure of the aqua-glyceroporin channel Fps1. Long-term adaptation occurs primarily through the production of Gpd1. (*B*) When maintained in a hyperosmotic environment (1 M sorbitol), the HOG cascade was quickly activated, which is seen by Hog1 nuclear enrichment. This transient signaling response lasted typically <20 min. The expression level of pSTL1-yECitrine (YFP) increased after an ~20-min delay, peaked around 600 fluorescence units after 100 min, and then decayed. (*C*) In contrast, the fluorescence level showed a continuous increase when stimulated periodically (T = 30 min). The increase rate was larger for longer pulses (red, 8 min; yellow, 5 min). Black curves are the expected behaviors based on our model of the pSTL1 induction. Solid lines and their envelopes are the experimental means and SDs of the cells' fluorescence. (*D*) Yeast cells grew as a monolayer in a microfluidic device that was used to rapidly change the cells' osmotic environment (blue frame) and image their response. Segmentation and cell tracking were done using a Hough transform (orange frame). The measured yECitrine fluorescence, either of a single cell or of the mean of all cells, was then sent to a state estimator connected to an MPC controller. A model (black frame) of pSTL1 induction was used to find the best possible series of osmotic pulses to apply in the future so that the predicted yECitrine level follows a target profile. (*E*) At the present time point (orange circle), the system state is estimated (green), and the MPC searches for the best input (pulse duration and number of pulses) (see text and *SI Materials and Methods*), which minimizes the distance of the MPC predictions (black curves) to the target profile (red dashed line) for the next 2 h. Here, the osmotic series of pulses that corresponds to the blue curve (4) was selected and sent to the microfluidic command. This control loop is iterated every 6 min.

A closed-loop control of the pSTL1 activity requires the acquisition and analysis of live cell images, the computation of the input (i.e., osmolarity) to be applied in the near future, and the ability to change the cells osmotic environment accordingly (Fig. 1 *D* and *E*).

Experimental Platform. To observe the cells and control their environment, we designed a versatile platform made of standard microscopy and microfluidic parts. The microfluidic device contained several 3.1-μm-high chambers that were connected by both ends to large channels through which liquid media could be perfused (Fig. 1*D*). Because the typical diameter of an *S. cerevisiae* cell is 4–5 μm, the cells were trapped in the chamber and grew as a monolayer. Their motion was limited to slow lateral displacement due to cell growth (Fig. S1). This design allowed for long-term cell tracking (>15 h) and relatively rapid media exchanges (~2 min). The HOG pathway was activated by switching between normal and sorbitol-enriched (1 M) media.

Model of pSTL1 Induction. To decide what osmotic stress to apply at a given time, we used an elementary model of pSTL1 induction. Many models have been proposed for the hyperosmotic stress response in yeast (10, 26–30). We used a generic model of gene expression written as a two-variable delay differential equation system, where the first variable denotes the recent osmotic stress felt by the cell and the second variable is the protein fluorescence level (Fig. 1*D*, *Materials and Methods*, Table S1, and *SI Materials and Methods*). Because our goal was to show robust control, despite the presence of unmodeled feedback loops, the adaptation mechanisms described above were purposefully neglected. The choice of this model was also motivated by the tradeoff between its ability to quantitatively predict the system's behavior (favors complexity) and the ease of solving state estimation problems (favors simplicity). Despite its simplicity, we found a fair agreement between model predictions and calibration data corresponding to fluorescence profiles obtained by applying either isolated or repeated osmotic shocks of various durations (Fig. 1*C* and Fig. S2).

Closing the Loop. The fluorescence intensity of a single cell arbitrarily chosen at the start of the experiment, or the average fluorescence intensity of the cell population, was sent to a state estimator (extended Kalman filter discussed in *SI Materials and Methods*) connected to a model predictive controller (31). Model Predictive Control (MPC) is an efficient framework well-adapted

to constrained control problems. Schematically, given a model of the system and desired temporal profiles for the system's outputs, MPC aims at finding inputs to minimize the deviation between the outputs of the model and the desired outputs. The control strategy is applied for a (short) period. Then, the new state of the system is observed, and this information is used to compute the control strategy to be applied during the next time interval. This receding horizon strategy yields an effective feedback control. In practice, every 6 min, given the current estimate of the system state, past osmotic shocks, and our model of gene expression, the controller searched for the optimal number of osmotic pulses to apply within the next 2 h and their optimal start times and durations (Fig. 1*E*). If a shock had to be applied within the next 6 min, then it was applied. Otherwise, the same computation was reiterated 6 min later based on new observations, thus effectively closing the feedback loop. Here, we dealt with short-term cell adaptation by imposing a maximal stress duration of 8 min and a 20-min relaxation period between consecutive shocks. Under such conditions, cells stay responsive to osmotic stress at all times (Fig. 1*C*). This can be explained by the fact that, in absence of stress, the glycerol channel Fps1 opens (21, 32) and lets the glycerol leak out of the cell, thus effectively resetting the osmotic state of the cells (29).

Note that a proportional integral (PI) controller would have been an attractive alternative, because it would not have required the development of a model of the system. With PI controllers, the applied input (i.e., stress) is simply the weighted sum of the current error (deviation between target and measured outputs) and the integral of the (recent) past errors. Consequently, using a PI controller to reach high levels of fluorescence would lead to a control strategy in which high stress is maintained over extended periods of time. This condition would trigger cell adaptation and eventually lead to a stalled situation in which the maximal stress is applied without any effect.

Closed-Loop Population Control Experiments. Our first goals were to maintain the average fluorescence level of a cell population at a-given constant value (set-point experiment) and force it to follow a time-varying profile (tracking experiment). Both types of experiments lasted at least 15 h, starting with a few cells and ending with 100–300 cells in the field of view (Fig. S3). The control objective was to minimize the mean square deviations (MSDs) between the mean fluorescence of the population of cells and the target profile. We succeeded in maintaining the average fluorescence level at a given constant value or forcing it to follow several given time-varying profiles (Fig. 2 *A–D*, Figs. S3, S4, and S5, and Movies S1, S2, and S3). Admissible time-varying target profiles were obviously constrained by the intrinsic timescales of the system, such as the maximal protein production and degradation rates. However, within these constraints, graded responses could be obtained. In Fig. 2*C*, for example, the trapeze slope is less steep than what maximal pSTL1 induction can deliver (Fig. 2 *A* and *B*). Note that our control strategy opened the possibility to reach higher fluorescence levels than what full induction with a step shock would allow (compare with Fig. 1*B*). Indeed, because of cell perfect adaptation to hyperosmotic stresses, a sustained 1 M sorbitol shock triggers only a transient gene expression and fluorescence peaks at moderate levels (Fig. 1*B*). By using repeated, well-separated pulses,

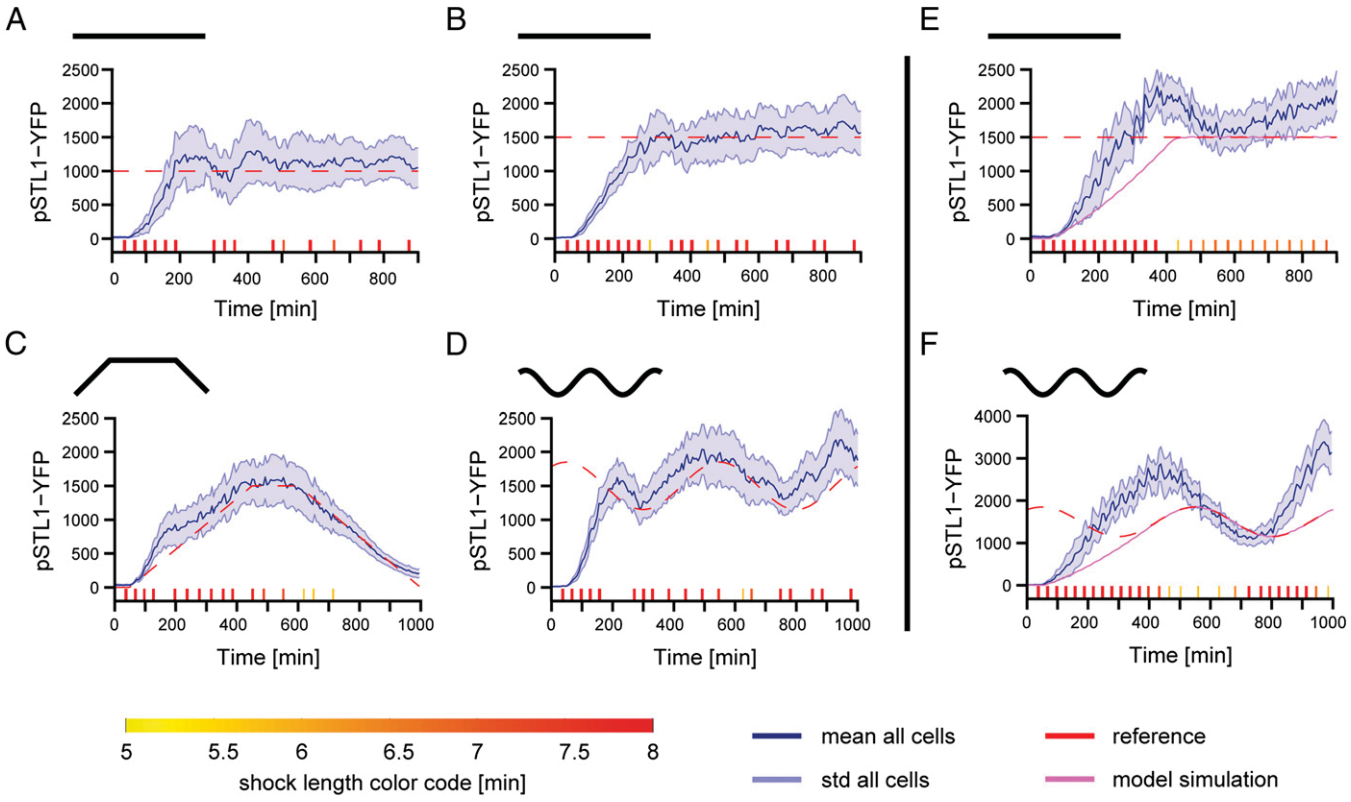


Fig. 2. Real-time control of gene expression can be achieved at the population level. (*A* and *B*) Set-point control experiments with target values 1,000 and 1,500 fluorescence units (f.u.; red dashed line). This unit is the same across all graphs (no renormalization). To avoid desensitizing the HOG pathway, the controller repeatedly applied short osmotic pulses (durations between 5 and 8 min). The timeline of osmotic events is shown at the bottom of each graph (color code along the bottom). Shock starting times and durations were computed in real time. The measured mean cell fluorescence is shown as solid blue lines. The envelopes indicate SD of the fluorescence distribution across the yeast population. (*C* and *D*) Tracking control experiments. In *C*, the target has a trapezoidal shape (maximum at 1,500 f.u.). In *D*, the target is sinusoidal (average value at 1,500 f.u.). In both cases, the mean level of fluorescence successfully follows the time-varying target profile. (*E* and *F*) Open-loop control experiments. Two examples of open-loop control (the osmotic inputs were computed using our model before starting the experiments) showing poor control quality. Errors accumulate over time. The simulated behavior of the system is represented in violet.

pSTL1 was iteratively activated (Fig. 1*C* and Movie S4). To assess the effective control range, we performed additional control experiments with target values spanning an order of magnitude (200–2,000 fluorescence units) (Fig. S5). Despite an initial overshoot for the lower target (200 fluorescence units), our results showed good control accuracy over time.

Quantitative limitations of our experimental platform can originate from the model, the state estimator, the control algorithm, and the intrinsic biological variability of gene expression. In silico analysis showed that applying the proposed control strategy to the (estimated state of the) system resulted in control performances that were significantly better than those obtained experimentally (Fig. 2 *E* and *F* and Fig. S4). Therefore, the control algorithm performed well, and future improvements should focus on system modeling and state estimation to better represent the experimental state of the system. To assess the importance of biological variability and modeling limitations, we carried out open-loop control experiments with the same objectives and the same model of the system. A time series of osmotic pulses was computed before the experiment and then sent to yeast cells without performing real-time corrections. Important deviations were found, indicating clear discrepancies between model predictions and the long-term system behavior (Fig. 2 *E* and *F*). As expected, open-loop strategies cannot result in a quantitative, robust control of gene expression. In contrast, closed-loop control performs well, despite significant biological variability and/or limited model accuracy.

Closed-Loop Single-Cell Control Experiments. In a second set of experiments, we focused on the real-time control of gene expression at the single-cell level. We tracked one single cell over at least 15 h and used its fluorescence to feed the MPC controller. As shown in Fig. 3, we obtained results with quality that is out of reach of any conventional gene induction system, both for constant and time-varying target profiles (Movies S5, S6, and S7). Because of intrinsic noise in gene expression, single-cell control was a priori more challenging than population control. Indeed, compared with the mean fluorescence levels in population control experiments, the fluorescence levels of controlled cells in single-cell control experiments showed larger fluctuations around the target values. However, at the cell level, the MSDs of controlled cells obtained in single-cell control experiments were significantly smaller than the MSDs of a cell in population control experiments (Fig. 4*B*, *SI Materials and Methods*, Table S2, and Fig. S6). For set-point control experiments in which fluctuations happen around a fixed reference value, we also defined the fluorescence noise level as the standard deviation (SD) over the mean. Again, we found that single-cell control significantly decreased noise at the cell level (Fig. 4*C*, *SI Materials and Methods*, Table S2, and Fig. S6). Taken together, these results show that real-time control effectively improves control quality and counteracts the effects of noise in gene expression when performed at the single-cell level. Interestingly, single-cell control experiments showed that, in few cases, the controlled cell behaved significantly differently from the rest of the population over extended periods of time (e.g., see Fig. 3*A*), suggesting long-term memory effects for gene expression spanning many cell generations. Lastly, the fact that, for different controlled cells but the same control objective, the decisions of the closed-loop controller were markedly different (Fig. 3*E*) highlights the fact that feedback control was critical to achieve good control performance at the single-cell level. This suggests that cell-to-cell variability and noise in gene expression fundamentally limit the quality of any open-loop inducible system.

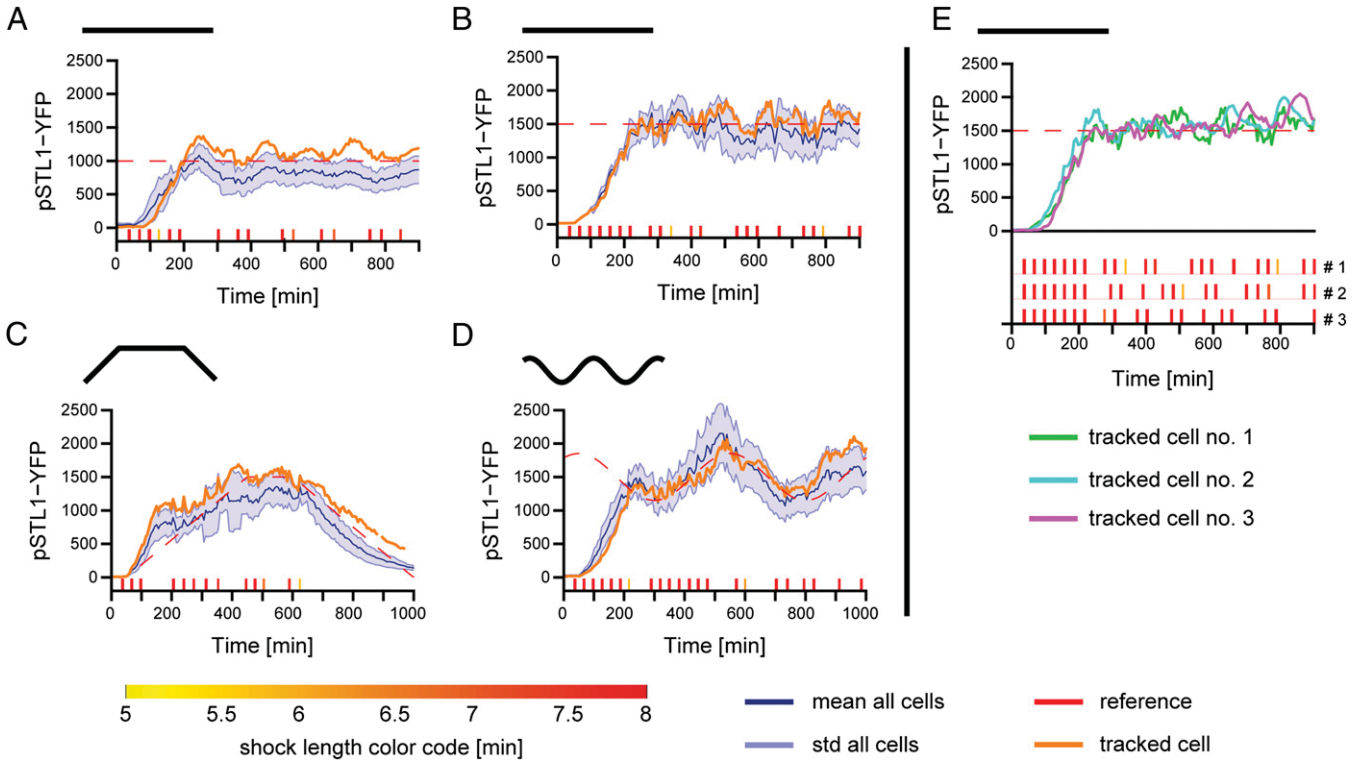


Fig. 3. Real-time control of gene expression can be achieved at the single-cell level. (*A* and *B*) Set-point control experiments at values 1,000 and 1,500 f.u. The yECitrine fluorescence of the controlled cells is shown as orange lines. The blue line and its envelope indicate the mean fluorescence and the SD of the fluorescence across the cell population. The population follows the target profile but with less accuracy than the controlled single cell. (*C* and *D*) Tracking control experiments. In *C*, the target has a trapezoidal shape (maximum at 1,500 f.u.). In *D*, the target is sinusoidal (average at 1,500 f.u.). (*E*) The fluorescence of the controlled cell in three different single-cell control experiments is represented together with the osmolarity profiles that were applied. Different experiments are labeled with different colors, and therefore, their corresponding osmotic inputs can be identified. It appears that, for each cell, the controller decisions were markedly different, showing that cell-to-cell variability was at play and that feedback control was critical when performing single-cell control.

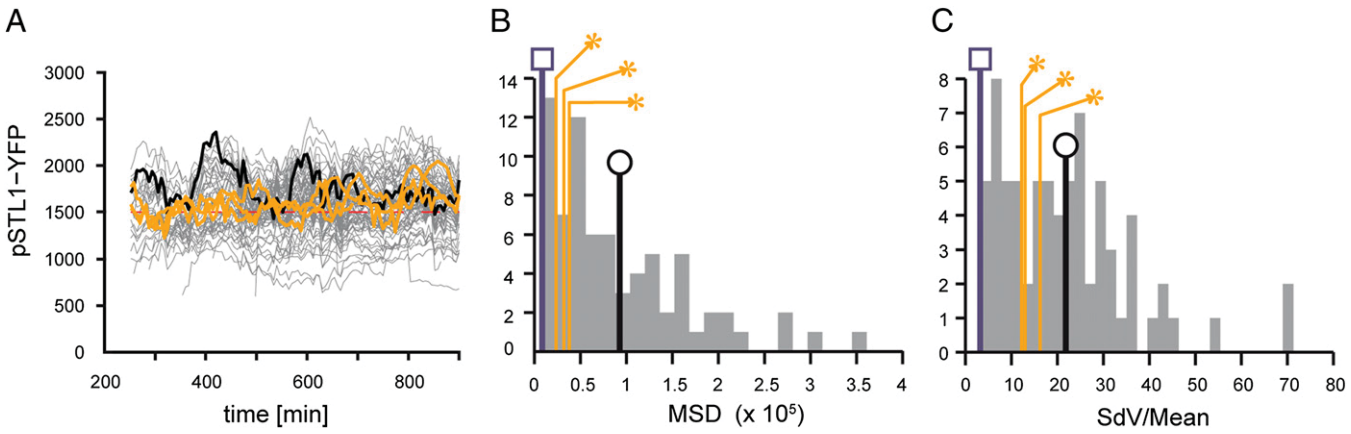


Fig. 4. Effectiveness of closed-loop control. (*A*) Single-cell fluorescence time profiles in two population control experiments (thin gray lines) and three single-cell control experiments (thick orange lines). One representative trace of a single cell in a population control experiment is shown in black. (*B*) Distribution of the MSDs of individual cells in population control experiments (gray). MSDs are defined with respect to the target profiles. The orange bars (stars) show the MSDs for the controlled cells in three single-cell control experiments. These data are compared with the mean MSD of single cells when controlling the population (black line, circle), which shows lower control quality. As expected, the control quality of the population is better (blue line, square), because noise in gene expression is averaged out. (*C*) Distribution of the noise levels defined as the ratio of the SD to the mean. Lower noise levels are observed for controlled cells in single-cell control experiments (orange, star) than a random cell in population control experiments (black line).

Conclusions

We showed that gene expression can be controlled in real time with quantitative accuracy at both the population and single-cell levels by interconnecting conventional microscopy, microfluidics, and computational tools. Importantly, we provided evidence that real-time control can dynamically limit the effects of gene expression stochasticity when applied at the single-cell level. This model predictive control framework overcame the presence of a significant delay between the environmental change and the fluorescent protein observation and the action of strong non-modeled endogenous negative feedback loops. The fact that good control results can be obtained in a closed-loop setting with a relatively coarse model of an endogenous promoter (compare with open-loop results) suggests that extensive modeling will not be required to transpose our approach to other endo- and exogenous induction systems (e.g., the galactose, methionine, or tetracycline inducible promoters). To appreciate the difficulty of the control problems that we addressed, one should keep in mind that the controlled system, a yeast cell, is an extremely complex and partially known dynamical system and that the controlled process, gene expression, is intrinsically stochastic.

Despite the fact that the importance of control theory for systems and synthetic biology has been widely recognized for more than a decade (33, 34), the actual use of in silico feedback loops to control intracellular processes has only been proposed recently. In 2011, we showed that the signaling activity in live yeast cells can be controlled by an in silico feedback loop (35). Using a PI controller, we controlled the output of a signal transduction pathway by modulating the osmotic environment of cells in real time. A similar framework has been proposed by Menolascina et al. (36). More recently, Toettcher et al. (37) used elaborate microscopy techniques and optogenetics to control (in real time and the single-cell level) the localization and activity of a signal transduction protein (PI3K) in eukaryotic cells. Interestingly, they were able to buffer external stimuli and clamp phosphatidylinositol (3,4,5)-triphosphate (PIP3) levels for short time scales. Because these two frameworks necessitate image acquisition at a high frequency, they are not suitable for long-term experiments. The most closely related work is the work by Milias-Argeitis et al. (19). Using optogenetic techniques, Milias-Argeitis et al. (19) managed to control the expression of a yeast gene to a constant target value over a few hours. Their approach is based on a chemostat culture and well-adapted for biotechnological applications, such as the production of biofuels or small-molecules. However, because it does not allow for single-cell tracking and control, it is less adapted to probe biological processes in single-cell quantitative biology applications. Controlling small cell populations, or even single cells, may be needed in multicellular systems, where cells differ by their genotype (38) or physical location (39).

Connecting living cells to computers is a promising field of research both for applied and fundamental research. By maintaining a system around specific operating points or driving it out of its standard operating regions, our approach offers unprecedented opportunities to investigate how gene networks process dynamical information at the cell level. We also anticipate that our platform will be used to complement and help the development of synthetic biology through the creation of hybrid systems resulting from the interconnection of in vivo and in silico computing devices.

Material and Methods

Yeast Strains and Plasmids. All experiments were performed using a pSTL1::yECitrine-HIS5, Hog1::mCherry-hph yeast strain derived from the S288C background. Cells were cultured overnight in synthetic complete (SC) medium at 30 °C; 4 h before loading them into the microfluidic chip, 60 μL overnight culture were diluted into 5 mL SC, thus obtaining an OD of ~0.19. During the experiment, cell growth continued, with a doubling time between 100 and 250 min (*SI Materials and Methods* and Movies S1, S2, S3, S4, S5, S6, and S7).

Microfluidics. We microfabricated a master wafer by standard soft lithography techniques. A microfluidic chip was made by casting polydimethylsiloxane (PDMS) (Sylgard 184 kit; Dow Corning) on the master wafer, curing it at 65 °C overnight, pealing it off, and bonding it to a glass coverslip after plasma activation. Cells were loaded into the imaging chamber by syringe injection. This created a positive pressure, which let the cells enter the trap. Liquid medium was flown by aspiration into the device using a peristaltic pump (IPC-N; Ismatec) placed after the microfluidic device. We used a flow rate of 230 μL/min. A computer-controlled three-way valve (LFA series; The Lee Company) was used to select between regular medium (SC) or the same medium supplemented with 1 M sorbitol. A switch of the valve state did not lead to an instantaneous change of the cells' environment inside the microfluidic device: a certain time (depending on the flow rate) was needed for the fluid to pass from the valve to the channels and the imaging chamber (Fig. S1).

Microscopy and Experimental Setup. We used an automated inverted microscope (IX81; Olympus) equipped with an X-Cite 120PC fluorescent illumination system (EXFO) and a QuantEM 512 SC camera (Roper Scientific). The YFP filters used were HQ500/20× (excitation filter; Chroma), Q515LP (dichroic; Chroma), and HQ535/30M (emission; Chroma). All these components were driven by the open-source software μManager (40), a plug-in of ImageJ (41), which we interfaced with Matlab using in-house–developed code. The temperature of the microscope chamber, which also contained the media reservoirs, was constantly held at a temperature of 30 °C by a temperature control

system (Life Imaging Services). Images were taken with a 100× objective (PlanApo 1.4 NA; Olympus). The fluorescence exposure time was 200 ms, with fluorescence intensity set to 50% of maximal power. The fluorescence exposure time was chosen such that the fluorescent illumination did not cause noticeable effects on cellular growth over extended periods of time. Importantly, illumination, exposure time, and camera gain were not changed between experiments, and no data renormalization was done. Therefore, the fluorescence intensities can be directly compared across experiments.

Image Analysis. The cellular boundaries were identified on the bright-field image using a circular Hough transform implemented in Matlab (42). For tracking, we compared the current image with the previous one, defined a cell-to-cell distance matrix, and used linear optimization to match pairs of cells. The tracking process was made more robust by also considering the last but one image if a gap was detected (caused by rare segmentation errors). The YFP fluorescence level in each cell was defined as the mean fluorescence level taken over the cell area after subtraction of the background fluorescent level. The signaling activity of the Hog1 cascade can be estimated by measuring the Hog1 nuclear enrichment. We defined the nuclear enrichment of Hog1::mCherry as the difference between the minimal and maximal fluorescence intensities within a cell. Maximal and minimal Hog1::mCherry intensities were computed by averaging the fluorescence of the 15 brightest and 15 dimmest pixels, respectively.

Modeling. The controller used a two dimensional ordinary differential equation (ODE) model to predict the behavior of the system:

$$\dot{x}_1 = u(t-\tau) - g_1 x_1$$

and

$$\dot{x}_2 = k_2 x_1 - g_2 \frac{x_2}{K + x_2},$$

where x_1 denotes the recent osmotic stress and x_2 denotes the protein fluorescence level. The osmotic input (u) is shifted by $\tau = 20$ min to account for the observed delay in the system. The remaining parameters have been estimated based on several calibration experiments: $g_1 = 4.02 \times 10^{-3}$, $k_2 = 0.58$, $g_2 = 37.5$, $K = 750$, and $\tau = 20$ (*SI Materials and Methods*, Table S1, and Fig. S2).

State Estimation. We implemented an extended Kalman filter, which estimates the system state based on fluorescent observations and the model of the system. The parameters of the filter (measurement noise R and process noise Q) were set to $R = 2{,}500$ and $Q = \mathrm{diag}(0.37, 925)$.

Model Predictive Control. The controller searches for osmolarity profiles that minimize the squared deviations between model output and target profile within the next 120 min, while fulfilling the input constraints (pulse duration of 5 to 8 min separated by at least 20 min). In practice, this problem is recast into a parameter search problem, in which parameters are used for encoding stress starting times and shock durations and solved using the global optimization tool CMAES. Because image analysis and parameter search may take up to 3 min, the input to be applied is not immediately available at the time of the measurement. Consequently, we apply at time t the input that was computed at time $t - 3$ min.

ACKNOWLEDGMENTS. The authors acknowledge discussions with D. di Bernardo (Tigem), P.-Y. Bourguignon (Max Planck), S. Léon (Institut Jacques Monod & Centre National de la Recherche Scientifique), F. Devaux, M. Garcia (Université Pierre et Marie Curie), J. M. di Meglio, A. Prastowo (Matière et Systèmes Complexes), R. Bourdais (Supélec), A. Kabla (Cambridge University), E. Cinquemani, H. de Jong, D. Ropers, J. Schaul, and S. Stoma (Institut National de Recherche en Informatique et en Automatique). We acknowledge the support of the Agence Nationale de la Recherche (under the references DiSiP-ANR-07-JCJC-0001 and ICEBERG-ANR-10-BINF-06-01), of the Région Ile de France (C'Nano-ModEnv), of the Action d'Envergure ColAge from INRIA/INSERM (Institut Nationale de la Santé et de la Recherche Médicale), of the MechanoBiology Institute, and of the Laboratoire International Associé CAFS (Cell Adhesion France-Singapour).

1. Bhalla US, Ram PT, Iyengar R (2002) MAP kinase phosphatase as a locus of flexibility in a mitogen-activated protein kinase signaling network. *Science* 297:1018–1023.
2. Hooshangi S, Thiberge S, Weiss R (2005) Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proc Natl Acad Sci USA* 102:3581–3586.
3. Cai L, Dalal CK, Elowitz MB (2008) Frequency-modulated nuclear localization bursts coordinate gene regulation. *Nature* 455:485–490.
4. Celani A, Vergassola M (2010) Bacterial strategies for chemotaxis response. *Proc Natl Acad Sci USA* 107:1391–1396.
5. Baumgartner BL, et al. (2011) Antagonistic gene transcripts regulate adaptation to new growth environments. *Proc Natl Acad Sci USA* 108:21087–21092.
6. O'Shaughnessy EC, Palani S, Collins JJ, Sarkar CA (2011) Tunable signal processing in synthetic MAP kinase cascades. *Cell* 144:119–131.
7. Walter E, Pronzato L (1997) *Identification of Parametric Models from Experimental Data* (Springer, Berlin).
8. Bennett MR, et al. (2008) Metabolic gene regulation in a dynamically changing environment. *Nature* 454:1119–1122.
9. Hersen P, McClean MN, Mahadevan L, Ramanathan S (2008) Signal processing by the HOG MAP kinase pathway. *Proc Natl Acad Sci USA* 105:7165–7170.
10. Muzzey D, Gómez-Uribe CA, Mettetal JT, van Oudenaarden A (2009) A systems-level analysis of perfect adaptation in yeast osmoregulation. *Cell* 138:160–171.
11. Shimizu TS, Tu Y, Berg HC (2010) A modular gradient-sensing network for chemotaxis in Escherichia coli revealed by responses to time-varying stimuli. *Mol Syst Biol* 6:382.
12. Pelet S, et al. (2011) Transient activation of the HOG MAPK pathway regulates bimodal gene expression. *Science* 332:732–735.
13. Rao CV (2012) Expanding the synthetic biology toolbox: Engineering orthogonal regulators of gene expression. *Curr Opin Biotechnol*, 10.1016/j.copbio.2011.12.015.
14. Voigt CA (2006) Genetic parts to program bacteria. *Curr Opin Biotechnol* 17:548–557.
15. Khalil AS, Collins JJ (2010) Synthetic biology: Applications come of age. *Nat Rev Genet* 11:367–379.
16. Elowitz MB, Levine AJ, Siggia ED, Swain PS (2002) Stochastic gene expression in a single cell. *Science* 297:1183–1186.
17. Balázsi G, van Oudenaarden A, Collins JJ (2011) Cellular decision making and biological noise: From microbes to mammals. *Cell* 144:910–925.
18. Munsky B, Neuert G, van Oudenaarden A (2012) Using gene expression noise to understand gene regulation. *Science* 336:183–187.
19. Milias-Argeitis A, et al. (2011) In silico feedback for in vivo regulation of a gene expression circuit. *Nat Biotechnol* 29:1114–1116.
20. de Nadal E, Alepuz PM, Posas F (2002) Dealing with osmostress through MAP kinase activation. *EMBO Rep* 3:735–740.
21. Hohmann S (2002) Osmotic stress signaling and osmoadaptation in yeasts. *Microbiol Mol Biol Rev* 66:300–372.
22. Miermont A, Uhlendorf J, McClean M, Hersen P (2011) The dynamical systems properties of the HOG signaling cascade. *J Signal Transduct* 2011:930940.
23. Yi TM, Huang Y, Simon MI, Doyle J (2000) Robust perfect adaptation in bacterial chemotaxis through integral feedback control. *Proc Natl Acad Sci USA* 97:4649–4653.
24. Ferreira C, et al. (2005) A member of the sugar transporter family, Stl1p is the glycerol/H+ symporter in Saccharomyces cerevisiae. *Mol Biol Cell* 16:2068–2076.
25. O'Rourke SM, Herskowitz I (2004) Unique and redundant roles for HOG MAPK pathway components as revealed by whole-genome expression analysis. *Mol Biol Cell* 15:532–542.
26. Klipp E, Nordlander B, Krüger R, Gennemark P, Hohmann S (2005) Integrative model of the response of yeast to osmotic shock. *Nat Biotechnol* 23:975–982.
27. Hao N, et al. (2007) A systems-biology analysis of feedback inhibition in the Sho1 osmotic-stress-response pathway. *Curr Biol* 17:659–667.
28. Mettetal JT, Muzzey D, Gómez-Uribe C, van Oudenaarden A (2008) The frequency dependence of osmo-adaptation in Saccharomyces cerevisiae. *Science* 319:482–484.
29. Zi Z, Liebermeister W, Klipp E (2010) A quantitative study of the Hog1 MAPK response to fluctuating osmotic stress in Saccharomyces cerevisiae. *PLoS ONE* 5:e9522.
30. Zechner C, et al. (2012) Moment-based inference predicts bimodality in transient gene expression. *Proc Natl Acad Sci USA* 109:8340–8345.
31. Findeisen R, Imsland L, Allgower F, Foss BA (2003) State and output feedback nonlinear model predictive control: An overview. *Eur J Control* 9:179–195.
32. Tamás MJ, et al. (1999) Fps1p controls the accumulation and release of the compatible solute glycerol in yeast osmoregulation. *Mol Microbiol* 31:1087–1104.
33. Csete ME, Doyle JC (2002) Reverse engineering of biological complexity. *Science* 295:1664–1669.
34. Iglesias PA, Ingalls BP (2009) *Control Theory and Systems Biology* (MIT Press, Cambridge, MA).
35. Uhlendorf J, Bottani S, Fages F, Hersen P, Batt G (2011) Towards real-time control of gene expression: Controlling the hog signaling cascade. *Pac Symp Biocomput* 2011:338–349.
36. Menolascina F, di Bernardo M, di Bernardo D (2011) Analysis, design and implementation of a novel scheme for in-vivo control of synthetic gene regulatory network. *Automatica* 47:1265–1270.
37. Toettcher JE, Gong D, Lim WA, Weiner OD (2011) Light-based feedback for controlling intracellular signaling dynamics. *Nat Methods* 8:837–839.
38. Regot S, et al. (2011) Distributed biological computation with multicellular engineered networks. *Nature* 469:207–211.
39. Sprinzak D, et al. (2010) Cis-interactions between Notch and Delta generate mutually exclusive signalling states. *Nature* 465:86–90.
40. Edelstein A, Amodaj N, Hoover K, Vale R, Stuurman N (2010) Computer control of microscopes using μManager. *Curr Protoc Mol Biol*, 10.1002/0471142727.mb1420s92.
41. Rasband WS (1997–2012) *ImageJ* (US National Institutes of Health, Bethesda, MD). Available at http://imagej.nih.gov/ij/. Accessed July 29, 2012.
42. Ballard DH (1981) Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognit* 13(2):111–122.