Hybrid Modeling and Simulation of Genetic Regulatory Networks: A Qualitative Approach

Hidde de Jong[1], Jean-Luc Gouzé[2], Céline Hernandez[3], Michel Page[1,4], Tewfik Sari[5], and Johannes Geiselmann[6]

[1] Institut National de Recherche en Informatique et en Automatique (INRIA), Unité de recherche Rhône-Alpes, 655 avenue de l'Europe, Montbonnot, 38334 Saint Ismier Cedex, France,
{Hidde.de-Jong,Michel.Page}@inrialpes.fr,
[2] Institut National de Recherche en Informatique et en Automatique (INRIA), Unité de recherche Sophia Antipolis, France,
Jean-Luc.Gouze@sophia.inria.fr,
[3] Swiss Institute of Bioinformatics (SIB), Geneva, Switzerland,
Celine.Hernandez@isb-sib.ch,
[4] École Supérieure des Affaires, Université Pierre Mendès France, Grenoble, France,
[5] Laboratoire de Mathématiques, Université de Haute Alsace, Mulhouse, France,
T.Sari@uha.fr,
[6] Laboratoire Plasticité et Expression des Génomes Microbiens, Université Joseph Fourier, Grenoble, France,
Hans.Geiselmann@ujf-grenoble.fr.

Abstract. The study of genetic regulatory networks has received a major impetus from the recent development of experimental techniques allowing the measurement of patterns of gene expression in a massively parallel way. This experimental progress calls for the development of appropriate computer tools for the modeling and simulation of gene regulation processes. We present a method for the hybrid modeling and simulation of genetic regulatory networks, based on a class of piecewise-linear (PL) differential equations that has been well-studied in mathematical biology. Distinguishing characteristics of the method are that it makes qualitative predictions of the behavior of regulatory systems and that it deals with discontinuities in the right-hand side of the differential equations. The simulation method has been implemented in Java in the computer tool Genetic Network Analyzer (GNA). The method and the tool have been used to analyze several networks of biological interest, including the network underlying the initiation of sporulation in *Bacillus subtilis*.

1 Introduction

The study of *genetic regulatory networks* has received a major impetus from the recent development of experimental techniques allowing the measurement of patterns of gene expression in a massively parallel way. This experimental progress calls for the development of appropriate computer tools for the modeling

and simulation of gene regulation processes. A variety of approaches for the modeling and simulation of genetic regulatory networks has been proposed in the past three decades [4, 17, 21, 25].

A particularly interesting approach towards the computational analysis of genetic regulatory networks, well-adapted to state-of-the-art measurement techniques in genomics, is based on a class of *piecewise-linear (PL)* differential equations originally proposed by Glass and Kauffman [10, 14]. The state variables in the PL models correspond to the concentrations of proteins encoded by genes in the network, while the differential equations represent the interactions arising from the regulatory influence of some proteins on the synthesis and degradation of others. The regulatory interactions are modeled by means of step functions, which gives rise to the piecewise-linear structure of the system. More precisely, the use of step functions divides the phase space into hyperrectangular regions, in each of which the system evolves according to a set of linear, uncoupled differential equations. On the boundaries between these regions, the system description switches from one set of linear, uncoupled equations to another.

The dual, continuous and discrete, nature of the PL models of genetic regulatory networks has attracted the interest of researchers in hybrid systems [1, 13]. In this paper, we present a modeling and simulation method [7–9] that extends the above work in two respects. First, the PL models being used are *qualitative*, in the sense that numerical values for parameters and initial conditions, which are usually not available, need not be specified. Instead, the models are supplemented by qualitative constraints in the form of algebraic inequalities. Second, the method is able to deal with discontinuities in the right-hand side of the differential equations, resulting from the use of step functions. The discontinuities give rise to non-trivial mathematical problems that are solved through the use of a Filippov generalization of the PL models [12, 15]. On a formal level, the PL models are related to a class of asynchronous logical models proposed by Thomas and colleagues [28].

The qualitative simulation method is supported by the publicly-available computer tool *GNA (Genetic Network Analyzer)* [6], which has been used to analyze several genetic regulatory networks of biological interest. We will illustrate the use of GNA by summarizing the results obtained in the modeling and simulation of the large and complex network underlying the initiation of sporulation in *Bacillus subtilis* [5].

2 PL Models of Genetic Regulatory Networks

The dynamics of genetic regulatory networks can be modeled by a class of piecewise-linear differential equations of the following general form [14, 22, 26]:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{g}(\boldsymbol{x})\,\boldsymbol{x},\ \boldsymbol{x} \geq \boldsymbol{0}, \tag{1}$$

where $\boldsymbol{x} = (x_1, \ldots, x_n)'$ is a vector of cellular protein concentrations, and $\boldsymbol{f} = (f_1, \ldots, f_n)'$, $\boldsymbol{g} = \mathrm{diag}(g_1, \ldots, g_n)$. The rate of change of each concentration x_i, $1 \leq i \leq n$, is defined as the difference of the rate of synthesis $f_i(\boldsymbol{x})$ and the rate of degradation $g_i(\boldsymbol{x})\,x_i$ of the protein.

The function $f_i : \mathbb{R}^n_{\geq 0} \to \mathbb{R}_{\geq 0}$ is defined as

$$f_i(\boldsymbol{x}) = \sum_{l \in L} \kappa_{il}\, b_{il}(\boldsymbol{x}), \tag{2}$$

where $\kappa_{il} > 0$ is a rate parameter, $b_{il} : \mathbb{R}^n_{\geq 0} \to \{0, 1\}$ a *regulation function*, and L a possibly empty set of indices of regulation functions. A regulation function b_{il} is the arithmetic equivalent of a Boolean function expressing the logic of gene regulation [22, 28]. The function g_i expresses the regulation of protein degradation. It is defined analogously to f_i, except that we demand that $g_i(\boldsymbol{x})$ is strictly positive. In addition, in order to formally distinguish degradation rates from synthesis rates, we will denote the former by γ instead of κ.

Fig. 1 gives an example of a simple genetic regulatory network. Genes a and b, transcribed from separate promoters, encode proteins A and B, each of which controls the expression of both genes. More specifically, proteins A and B repress gene a as well as gene b at different concentrations. Repression of the genes is achieved by binding of the proteins to regulatory sites overlapping with the promoters.

The network in Fig. 1 can be described by means of the following pair of state equations:

$$\dot{x}_a = \kappa_a\, s^-(x_a, \theta^2_a)\, s^-(x_b, \theta^1_b) - \gamma_a\, x_a \tag{3}$$

$$\dot{x}_b = \kappa_b\, s^-(x_a, \theta^1_a)\, s^-(x_b, \theta^2_b) - \gamma_b\, x_b. \tag{4}$$

Gene a is expressed at a rate $\kappa_a > 0$, if the concentration of protein A is below its threshold θ^2_a and the concentration of protein B below its threshold θ^1_b, that is, if $s^-(x_a, \theta^2_a)\, s^-(x_b, \theta^1_b) = 1$. Recall that $s^-(x, \theta)$ is a step function evaluating to 1, if $x < \theta$, and to 0, if $x > \theta$. Protein A is spontaneously degraded at a rate proportional to its own concentration ($\gamma_a > 0$ is a rate constant). The state equation of gene b is interpreted analogously.

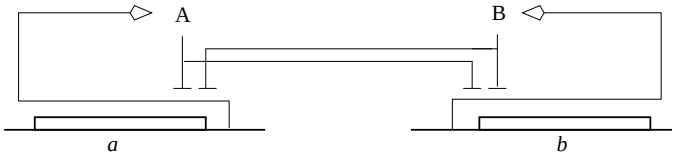

Fig. 1. Example of a genetic regulatory network of two genes (a and b), each coding for a regulatory protein (A and B).

3 Properties of PL Models

The dynamical properties of PL models of the form (1) can be analyzed in the n-dimensional phase space box $\Omega = \Omega_1 \times \ldots \times \Omega_n$, where $\Omega_i = \{x_i \in \mathbb{R}_{\geq 0} \mid$

$0 \leq x_i \leq max_i\}$, $1 \leq i \leq n$, and max_i is a parameter denoting a maximum concentration for the protein.

In general, a protein encoded by a gene is involved in different interactions at different threshold concentrations, which after ordering are denoted by $\theta_i^1, \ldots, \theta_i^{p_i}$. The $n-1$-dimensional hyperplanes $x_i = \theta_i^{k_i}$, $1 \leq k_i \leq p_i$, divide Ω into regions that are called *domains*. More precisely, a domain $D \subseteq \Omega$ is defined by $D = D_1 \times \ldots \times D_n$, where every D_i, $1 \leq i \leq n$, is defined by one of the equations below:

$$\begin{aligned} D_i =& \{x_i \mid 0 \leq x_i < \theta_i^1\}, \\ D_i =& \{x_i \mid x_i = \theta_i^1\}, \\ D_i =& \{x_i \mid \theta_i^1 < x_i < \theta_i^2\}, \\ & \ldots \\ D_i =& \{x_i \mid \theta_i^{p_i} < x_i \leq max_i\}. \end{aligned} \tag{5}$$

If for a domain D, there are some i, j, $1 \leq i \leq n$, $1 \leq j \leq p_i$, such that $D_i = \{x_i \mid x_i = \theta_i^j\}$, then D is called a *switching* domain. The *order* of a switching domain is a number between 1 and n, equal to the number of switching variables. A domain that is not a switching domain is called a *regulatory* domain.

In Fig. 2(a) the two-dimensional phase space box Ω for the example network is shown. As proteins A and B each have two thresholds, the phase space box is partitioned into 9 regulatory and 16 switching domains. For example, $D^1 = \{(x_a, x_b) \in \mathbb{R}^2 \mid 0 \leq x_a < \theta_a^1,\ 0 \leq x_b < \theta_b^1\}$ is a regulatory domain, whereas $D^4 = \{(x_a, x_b) \in \mathbb{R}^2 \mid 0 \leq x_a < \theta_a^1,\ x_b = \theta_b^2\}$ is a (first-order) switching domain.

When evaluating the step function expressions in (1) in a regulatory domain, f_i and g_i reduce to sums of rate constants. More precisely, in a regulatory domain D, f_i reduces to some $\mu_i^D \in M_i \equiv \{f_i(\boldsymbol{x}) \mid \boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{max}\}$, and g_i to some $\nu_i^D \in N_i \equiv \{g_i(\boldsymbol{x}) \mid \boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{max}\}$. M_i and N_i collect the synthesis and degradation rates of the protein in different domains of Ω. Inside D, the state equations thus simplify to linear, uncoupled differential equations

$$\dot{\boldsymbol{x}} = \boldsymbol{\mu}^D - \boldsymbol{\nu}^D \boldsymbol{x}, \tag{6}$$

where $\boldsymbol{\mu}^D = (\mu_1^D, \ldots, \mu_n^D)'$ and $\boldsymbol{\nu}^D = \mathrm{diag}(\nu_1^D, \ldots, \nu_n^D)$. For every regulatory domain D, we define the function $\phi_i(D) = \mu_i^D / \nu_i^D$. Analysis of (6) shows that all solution trajectories in D monotonically tend towards a *target equilibrium*, a stable equilibrium given by $\boldsymbol{x} = \boldsymbol{\phi}(D)$, with $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)'$ [14,22,26]. The target equilibrium level μ_i^D / ν_i^D of the protein concentration x_i gives an indication of the strength of gene expression in D. Call $\boldsymbol{\Phi}(D) = \{\boldsymbol{\phi}(D)\}$ the *target equilibrium set* of D. If $\boldsymbol{\Phi}(D) \cap D \neq \{\}$, then all trajectories will remain in D. If not, they will leave D at some point.

In the example, we have $M_a = \{0, \kappa_a\}$, $N_a = \{\gamma_a\}$ for protein A, and $M_b = \{0, \kappa_b\}$, $N_b = \{\gamma_b\}$ for protein B. In regulatory domain D^1 in Fig. 2(a), the trajectories tend towards the target equilibrium $\boldsymbol{\phi}(D^1) = (\kappa_a/\gamma_a, \kappa_b/\gamma_b)$. Since $\boldsymbol{\Phi}(D_1) \cap D_1 = \{\}$, the trajectories starting in D will leave this domain at some

point. Different regulatory domains generally have different target equilibria. For instance, in regulatory domain D^3, the target equilibrium is given by $(0, \kappa_b/\gamma_b)$.

The global solution of (1) could be obtained by piecing together the local solutions in regulatory domains, in such a way as to guarantee continuity of the global solution across the threshold hyperplanes [10, 26]. This works fine as long as trajectories arriving at a threshold hyperplane can be continued in another regulatory domain, *e.g.*, trajectories arriving at the switching domain D^2 from the regulatory domain D^1 (Fig. 2(a)). However, when the trajectories on both sides of a threshold hyperplane evolve towards this plane, as in the case of trajectories arriving from D^3 and D^5 at D^4, mathematical perplexities arise. There is no indication on how the local solutions in D^3 and D^5 can be continued.

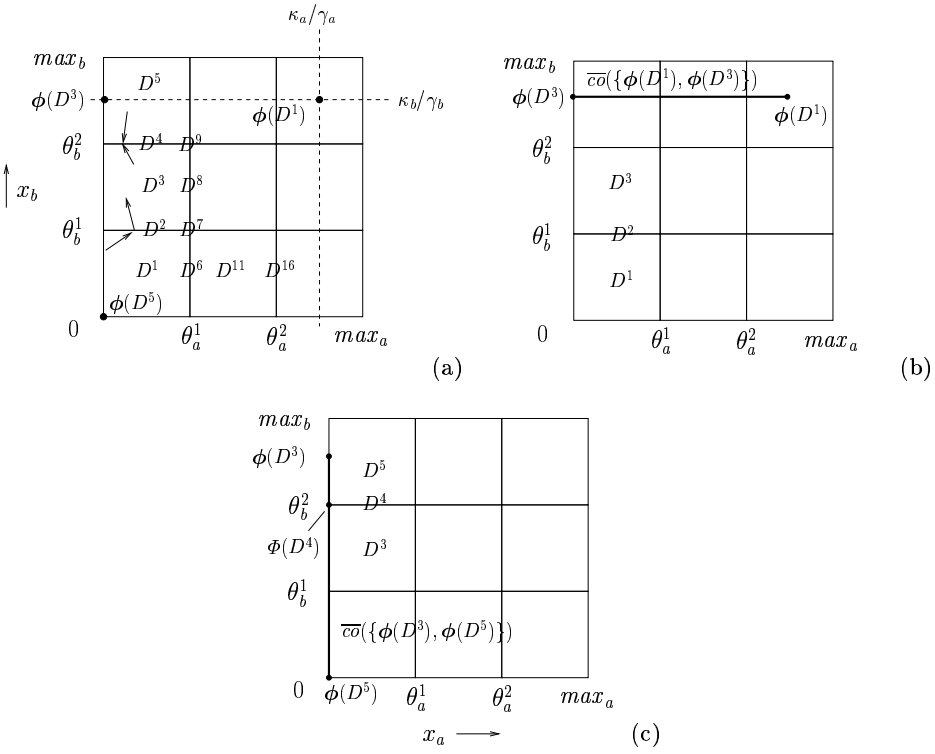

Fig. 2. (a) Phase space box for the example network in Fig. 1. $\phi(D^1)$, $\phi(D^3)$, $\phi(D^5)$ denote the target equilibria of the regulatory domains D^1, D^3, D^5. In addition, the figure shows the discontinuities at the switching domains D^2 and D^4. (b)-(c) Determination of the target equilibrium sets $\Phi(D^2)$ and $\Phi(D^4)$.

The troubles at the threshold hyperplanes are caused by discontinuities in the right-hand side of (1), due to the use of step functions. In order to deal with these discontinuities, we will use a method originally proposed by Filippov [12].

This method, recently applied by Gouzé and Sari [15] to PL systems of the form (1), consists of extending a system of differential equations with discontinuous right-hand sides into a system of differential inclusions. Using this generalization, it can be shown that, in the case of a switching domain D, the trajectories either traverse D instantaneously or tend towards a target equilibrium set $\Phi(D)$. As a summary of the analysis in [7, 8], consider a switching domain D of order k, contained in the $n-k$-dimensional hyperplane C. Moreover, let $R(D)$ be the set of regulatory domains that have D in their boundary. Then

$$\Phi(D) = C \cap \{\overline{co}\,(\{\phi(D') \mid D' \in R(D)\}). \tag{7}$$

That is, $\Phi(D)$ is the smallest closed convex set of the target equilibria of regulatory domains D' having D in their boundary, intersected with the hyperplane containing D. If $\Phi(D) = \{\}$, the solutions arriving at D will cross the switching domain instantaneously. On the other hand, if $\Phi(D) \neq \{\}$, then there exist solutions remaining in D for some interval of time, sliding along D towards a target equilibrium in $\Phi(D)$. If $\Phi(D) \cap D \neq \{\}$, then some trajectories may never leave D. If not, they will leave D at some point.

Consider the examples in Fig. 2(b)-(c). The target equilibrium set $\Phi(D^2)$ of the switching domain D^2 is defined, following (7), by the intersection of $\overline{co}(\{\phi(D^1), \phi(D^3)\})$ and the threshold hyperplane $x_b = \theta_b^1$. The smallest closed convex set consists of the linear segment connecting the points $(\kappa_a/\gamma_a, \kappa_b/\gamma_b)$ and $(0, \kappa_b/\gamma_b)$. Because $\overline{co}(\{\phi(D^1), \phi(D^3)\})$ and the threshold plane $x_b = \theta_b^1$ do not intersect in the figure, $\Phi(D^2) = \{\}$ and all solutions instantaneously cross D^2. This is different in the case of D^4. Here, the target equilibrium set $\Phi(D^4)$ is given by the intersection of $\overline{co}(\{\phi(D^3), \phi(D^5)\})$, the linear segment connecting the points $(0, \kappa_b/\gamma_b)$ and $(0, 0)$, and the threshold hyperplane $x_b = \theta_b^2$. Consequently, $\Phi(D^4)$ equals $\{(0, \theta_b^2)\}$, and all solutions arriving at D^4 from D^3 or D^5 slide along the threshold plane towards $(0, \theta_b^2)$. Because $\Phi(D^4)$ is included in D^4, $(0, \theta_b^2)$ is an equilibrium of the system. Closer analysis reveals that it is stable.

4 Qualitative Constraints on Parameters

Most of the time, precise numerical values for the threshold and rate parameters in (1) are not available. Instead, we will specify qualitative constraints on the parameter values, as explained in [8]. These constraints, having the form of algebraic inequalities, can usually be inferred from biological data.

The first type of constraints is obtained by ordering the p_i threshold concentrations of the protein encoded by gene i, yielding the *threshold inequalities*:

$$0 < \theta_i^1 < \ldots < \theta_i^{p_i} < max_i, \tag{8}$$

The threshold inequalities determine the partitioning of Ω into regulatory and switching domains.

In the case of protein A, there are two threshold concentrations: θ_a^1 and θ_a^2. Assuming the first to be lower than the second, we obtain the threshold

inequalities $0 < \theta_a^1 < \theta_a^2 < max_a$. The ordering of the thresholds of protein B give rise to $0 < \theta_b^1 < \theta_b^2 < max_b$.

Second, the possible target equilibrium levels μ_i^D/ν_i^D of x_i in different regulatory domains $D \subseteq \Omega$ can be ordered with respect to the threshold concentrations. The resulting *equilibrium inequalities* define the strength of gene expression in the domain in a qualitative way, on the scale of ordered threshold concentrations. More precisely, for every $\mu_i \in M_i$, $\nu_i \in N_i$, and $\mu_i, \nu_i \neq 0$, we specify one of the following pairs of inequalities:

$$\begin{aligned} &0 < \mu_i/\nu_i < \theta_i^1, \\ &\theta_i^1 < \mu_i/\nu_i < \theta_i^2, \\ &\ldots \\ &\theta_i^{p_i} < \mu_i/\nu_i < max_i. \end{aligned} \tag{9}$$

The equilibrium inequalities constrain the relative position of D and its target equilibrium set $\Phi(D)$.

The equilibrium inequalities for x_a in the example are $\theta_a^2 < \kappa_a/\gamma_a < max_a$. In the absence of protein B, while protein A has not yet reached its highest level, gene a is expressed at a rate κ_a. The corresponding target equilibrium value κ_a/γ_a of x_a must be above the second threshold θ_a^2, otherwise the concentration of the protein would not be able to reach or maintain a level at which the observed negative autoregulation of gene a occurs. In a similar way, we set $\theta_b^2 < \kappa_b/\gamma_b < max_b$ for x_b.

A *quantitative* PL model of a genetic regulatory network consists of state equations (1) and numerical parameter values $\boldsymbol{\theta}, \boldsymbol{\kappa}, \boldsymbol{\gamma}$. In a *qualitative* PL model, on the other hand, the state equations are supplemented by threshold and equilibrium inequalities. Every quantitative PL model can be uniquely abstracted into a qualitative PL model.

5 Qualitative Simulation of Genetic Regulatory Networks

Let $\boldsymbol{x}$, defined on some time-interval $[0, \tau[$, be a solution of a quantitative PL model describing a genetic regulatory network. Furthermore, at some time-point t, $0 \leq t < \tau$, $\boldsymbol{x}(t) \in D$. A qualitative description of $\boldsymbol{x}$ at t consists of the domain D, supplemented by the relative position of D and $\Phi(D)$. We call this the *qualitative state* of the system. On $[0, \tau[$ the solution traverses a sequence of domains $D^0, \ldots, D^m$ in Ω. Whenever $\boldsymbol{x}$ enters a new domain, the system makes a transition to a new qualitative state. The sequence of qualitative states corresponding to the sequence of domains is called the *qualitative behavior* of the system on the time-interval. Every solution of a quantitative PL model can be uniquely abstracted into a qualitative behavior [8].

Given a qualitative PL model and initial conditions in a domain D^0, the aim of *qualitative simulation* is to determine the possible qualitative behaviors of the system [20]. More precisely, denoting by X the set of solutions $\boldsymbol{x}(t)$ of

all quantitative PL models corresponding to the qualitative model, such that $\boldsymbol{x}(0) = \boldsymbol{x}_0 \in D^0$, the aim of qualitative simulation is to find the set of qualitative behaviors abstracting from some $\boldsymbol{x} \in X$.

In [8] a simulation algorithm is described that generates a set of qualitative behaviors by recursively determining qualitative states and transitions from qualitative states, starting at the qualitative state associated with the initial domain D^0. Instead of performing extensive numerical calculations, the simulator reaches its goal through symbolic computation, by exploiting the parameter inequalities (8)-(9). The simulation results in a *transition graph*, a directed graph of qualitative states and transitions between qualitative states. The transition graph contains *qualitative equilibrium states* or *qualitative cycles*. These may correspond to equilibrium points or limit cycles reached by solutions in X, and hence indicate functional modes of the regulatory system.

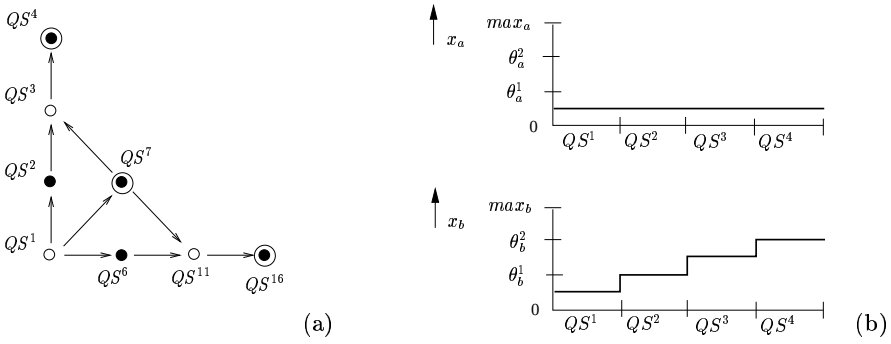

Fig. 3. (a) Transition graph resulting from a simulation of the example system starting in the domain D^1. Qualitative states associated with regulatory domains and switching domains are indicated by unfilled and filled dots, respectively. Qualitative states associated with domains containing an equilibrium point are circled [8]. (b) Detailed description of the qualitative behavior $\langle QS^1, QS^2, QS^3, QS^4 \rangle$.

Fig. 3(a) shows the transition graph for a qualitative simulation of the example system, starting in the regulatory domain D^1, where both x_a and x_b lie below their first threshold. As can be seen, the simulation results in five qualitative behaviors leading to different qualitative equilibrium states. In QS^{16}, associated with the switching domain D^{16} in Fig. 2(a), protein A is present at a high concentration ($x_a = \theta_a^2$), whereas protein B is present at a low concentration ($0 \leq x_b < \theta_b^1$). In QS^4, associated with D^4, protein A is present at a low concentration ($0 \leq x_a < \theta_a^1$) and protein B at a high concentration ($x_b = \theta_b^2$). In QS^7, associated with D^7, protein A and protein B are present at intermediate concentrations ($x_a = \theta_a^1$ and $x_b = \theta_b^1$). The qualitative equilibrium states QS^4 and QS^{16} correspond to stable equilibria of the system, whereas QS^7 corresponds to an unstable equilibrium.

A sequence of qualitative states in the transition graph represents a predicted qualitative behavior of the system. Fig. 3(b) gives a detailed description of one qualitative behavior, $\langle QS^1, QS^2, QS^3, QS^4 \rangle$. It shows for each qualitative state the corresponding domain, by indicating the (threshold) bounds for the concentration variables. In QS^1, for instance, x_a lies between 0 and θ_a^1, while x_b lies between 0 and θ_b^1. In the (instantaneous) state QS^2, x_b equals θ_b^1.

It has been demonstrated that the transition graph generated by the simulation algorithm covers all qualitative behaviors abstracting from some $\boldsymbol{x} \in X$ [8]. That is, whatever the exact numerical values for the parameters be, if these values are consistent with the threshold and equilibrium inequalities specified in the qualitative PL model, the qualitative shape of the solution is described by a sequence of states in the transition graph.

The qualitative simulation method has been implemented in Java 1.3 in the program *Genetic Network Analyzer (GNA)* [6]. GNA is available for non-profit academic research purposes at `http://www-helix.inrialpes.fr/gna`. The core of the system is formed by the simulator, which generates a transition graph from a qualitative PL model and initial conditions. The input of the simulator is obtained by reading and parsing text files specified by the user. A graphical user interface (GUI), named *VisualGNA*, assists the user in specifying the model of a genetic regulatory network as well as in interpreting the simulation results. Fig. 4 shows a screen capture of GNA for the example network.

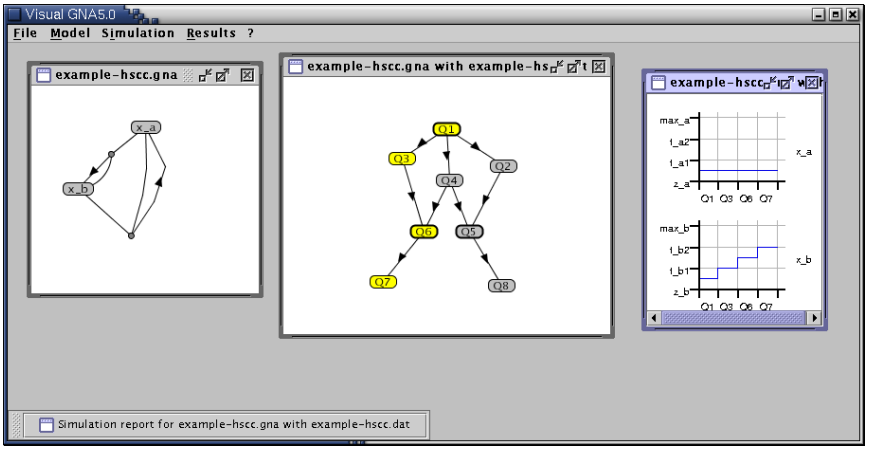

Fig. 4. Modeling and simulation of the genetic regulatory network of Fig. 1 by means of GNA. The window on the left shows the proteins and interactions of the sporulation network, the window in the middle part of the state transition graph resulting from simulation of the network under initial conditions inducing sporulation, and the window on the right the temporal sequence of qualitative states in one selected path in the state transition graph.

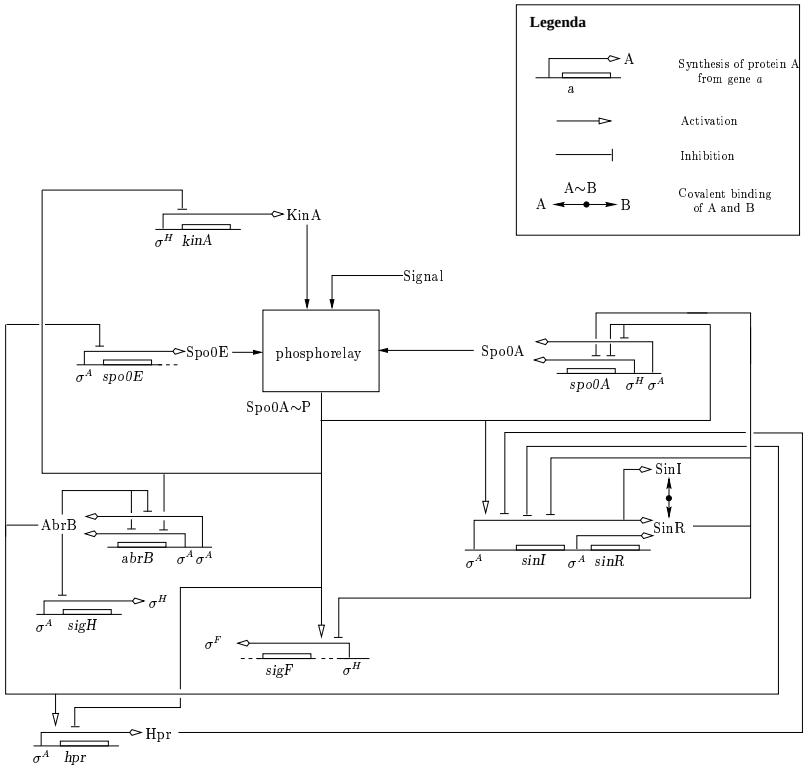

Fig. 5. Key genes, proteins, and regulatory interactions making up the network involved in *B. subtilis* sporulation. In order to improve the legibility of the figure, the control of transcription by the sigma factors σ^A and σ^H has been represented implicitly, by annotating the promoter with the corresponding sigma factor.

6 Application: Initiation of Sporulation in *Bacillus subtilis*

Under conditions of nutrient deprivation, the Gram positive soil bacterium *Bacillus subtilis* can abandon vegetative growth and form a dormant, environmentally-resistant spore instead [3, 16, 19, 27]. During vegetative growth, the cell divides symmetrically and generates two identical cells. During sporulation, on the other hand, cell division is asymmetric and results in two different cell types: the smaller cell (the *forespore*) develops into the spore, whereas the larger cell (the *mother cell*) helps to deposit a resistant coat around the spore and then disintegrates. The decision to abandon vegetative growth and initiate sporulation involves a radical change in the genetic program, the pattern of gene expression, of the cell. The switch of genetic programme is controlled by a complex genetic regulatory network integrating various environmental, cell-cycle, and metabolic signals. Due to the ease of genetic manipulation of *B. subtilis*, it has been possible to identify and characterize a large number of the genes, proteins, and

interactions making up this network. Currently, more than 125 genes are known to be involved [11].

The qualitative simulation method based on PL models will be illustrated by analyzing the genetic regulatory network underlying the initiation of sporulation in *B. subtilis*. A graphical representation of the regulatory network controlling the initiation of sporulation is shown in Fig. 5, displaying key genes and their promoters, proteins encoded by the genes, and the regulatory action of the proteins. References to the experimental literature having been used to compile the network are given in [5].

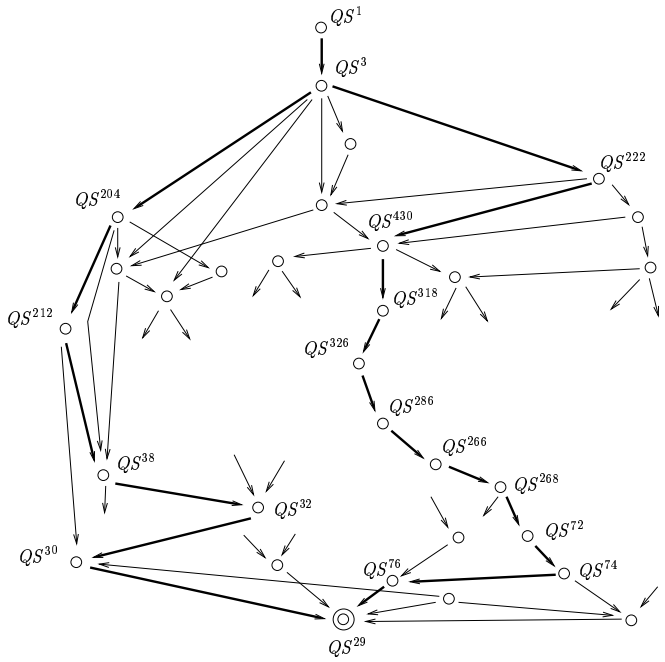

Fig. 6. Fragment of the state transition graph produced for vegetative growth conditions, when the sporulation signal is present.

The network is centered around a *phosphorelay*, which integrates a variety of environmental, cell-cycle, and metabolic signals. Under conditions appropriate for sporulation, the phosphorelay transfers a phosphate to the Spo0A regulator, a process modulated by kinases and phosphatases. The phosphorelay has been simplified in this paper by ignoring intermediate steps in the transfer of phosphate to Spo0A. However, this simplification does not affect the essential function of the phosphorelay: modulating the phosphate flux as a function of the competing action of kinases and phosphatases (here KinA and Spo0E). Under conditions conducive to sporulation, such as nutrient deprivation or high population density, the concentration of phosphorylated Spo0A (Spo0A$\sim$P) may reach

a threshold value above which it activates various genes that commit the bacterium to sporulation. The choice between vegetative growth and sporulation in response to adverse environmental conditions is the outcome of competing positive and negative feedback loops, controlling the accumulation of Spo0A$\sim$P [16, 19].

Notwithstanding the enormous amount of work devoted to the elucidation of the network of interactions underlying the sporulation process, very little quantitative data on kinetic parameters and molecular concentrations are available. The aim of the example is to show that GNA is able to reproduce the observed qualitative behavior of wild-type and mutant bacteria from a model that is a synthesis of available data in the literature. To this end, the graphical representation of the network has been translated into a PL model supplemented by qualitative constraints on the parameters. The resulting model consists of nine state variables and two input variables. The 49 parameters are constrained by 58 parameter inequalities, the choice of which is largely determined by biological data [5].

GNA has been used to simulate the response of a wild-type *B. subtilis* cell to nutrient depletion and high population density. Starting from initial conditions representing vegetative growth, the system is perturbed by a sporulation signal that causes KinA to autophosphorylate. Simulation of the network takes less than a few seconds to complete on a PC (500 MHz, 128 MB of RAM), and gives rise to a transition graph of 465 qualitative states. Many of these states are associated with switching domains that the system traverses instantaneously. Since the biological relevance of the latter states is limited, they can be eliminated from the transition graph. This leads to a reduced transition graph with 82 qualitative states, part of which is shown in Fig. 6.

The transition graph faithfully represents the two possible responses to nutrient depletion that are observed for *B. subtilis*: either the bacterium continues vegetative growth or it enters sporulation. A typical qualitative behavior for sporulation as well as for vegetative growth are shown in Fig. 7. The initiation of sporulation is determined by positive feedback loops acting through Spo0A and KinA, and a negative feedback loop involving Spo0E. When the rate of accumulation of the kinase KinA outpaces the rate of accumulation of the phosphatase Spo0E, we observe transient expression of *sigF*, *i.e.* a spo^+ phenotype (Fig. 7(a)). If the kinetics of these processes are inversed, *sigF* is never activated and we observe a spo^- phenotype (Fig. 7(b)). Deletion or overexpression of genes in the network of Fig. 5 may disable a feedback circuit, leading to specific changes in the observed sporulation phenotype. The results of the simulation of a dozen examples of sporulation mutants are discussed in [5].

7 Discussion

We have presented a method for the modeling and simulation of genetic regulatory networks. The method is based on a class of piecewise-linear (PL) differential equations that has been well-studied in mathematical biology. The PL models

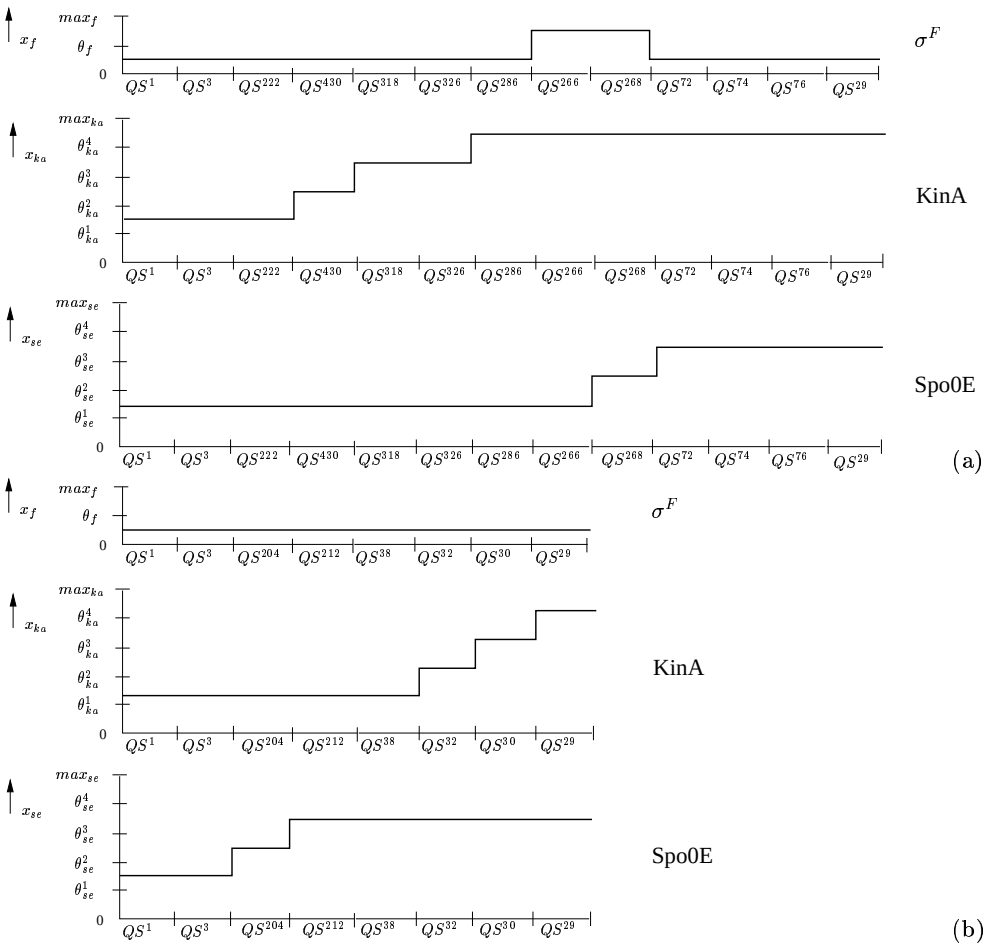

Fig. 7. (a) Temporal evolution of selected protein concentrations in a typical qualitative behavior corresponding to the spo^+ phenotype. (b) Idem, but for a typical qualitative behavior corresponding to the spo^- phenotype.

constitute a hybrid description of genetic regulatory networks, in the sense that they distinguish modes in which the system evolves continuously and discrete transitions between these modes. In the terminology of Mosterman and Biswas [23], the transitions are of one of two types. For example, *pinnacle mode transitions* occur when, upon reaching a switching domain D, the domain is traversed instantaneously ($\Phi(D) = \{\}$). On the other hand, *continuous mode transitions* occur when the system remains in D for an interval of time ($\Phi(D) \neq \{\}$).

Hybrid-system formalizations of genetic regulatory networks have been proposed by Alur *et al.* [1] and Ghosh and Tomlin [13]. The method presented in this paper extends these approaches in two respects. First, it provides a qualitative analysis of the behavior of the networks, generalized to higher-order systems. Second, it deals with discontinuities in the right-hand side of the differential equations in a mathematically proper and practically useful manner, by employing a Filippov generalization of the PL models. In order to handle discontinuities entailed by mode transitions, hybrid-system simulation methods based on Filippov solutions have been developed [24]. These methods are suitable for (semi-)quantitative, but not for qualitative PL models. In comparison with classical qualitative simulators like QSIM [20], the method presented here has been adapted to a particular class of systems, exploiting the favorable mathematical properties of (1). This allows it to scale up to large and complex genetic regulatory networks.

The PL models being used in this paper describe a genetic regulatory network as a set of genes encoding proteins that control the synthesis and degradation of other proteins. The models abstract from the precise molecular mechanisms involved, by expressing the underlying regulatory logic in terms of step functions. It would be possible to generalize the modeling framework so as to give a detailed description of the molecular mechanisms, for example the phosphorylation reactions in the phosphorelay in the sporulation network (Fig. 5). Mathematical biology offers the building blocks for achieving such a generalization, through well-established modeling approaches like mass-action and power-law kinetics [18, 29] (see [2] for an illustration in a hybrid-system context). However, this generalization would introduce nonlinearities that make it difficult to treat the dynamics of higher-order systems in a qualitative way. Moreover, due to differences in time-scale, it is often more appropriate to abstract away the dynamics of the molecular mechanisms by means of quasi-equilibrium assumptions, giving rise to step function approximations. This latter approach has been followed in the case of the phosphorelay (see [5] for details).

The simulation method has been implemented in Java in the computer tool Genetic Network Analyzer (GNA). The implementation has been used to study the network underlying the initiation of sporulation in *B. subtilis*. GNA is able to reproduce the observed qualitative behavior of wild-type and mutant bacteria from a model that is a synthesis of available data in the literature. Because sporulation in *B. subtilis* is one of the best-studied prokaryotic model systems, it is an excellent case study for the validation of the simulation tool. However, the real interest of tools like GNA comes from the simulation of genetic reg-

ulatory networks that are less understood and the use of the predictions thus obtained for guiding further experimental work. We are currently applying GNA in the context of studies of the global regulation of transcription in *E. coli* and *Synechocystis*.

Acknowledgments The authors would like to thank Grégory Batt for contributions to this paper.

References

1. R. Alur, C. Belta, F. Ivančíc, V. Kumar, M. Mintz, G.J. Pappas, H. Rubin, and J. Schlug. Hybrid modeling and simulation of biomolecular networks. In M.D. Di Benedetto and A. Sangiovanni-Vincentelli, editors, *Hybrid Systems: Computation and Control (HSCC 2001)*, volume 2034 of *LNCS*, pages 19–32. Springer-Verlag, 2001.
2. C. Belta, L.C.G.J.M. Habets, and V. Kumar. Control of multi-affine systems on rectangles with an application to gene transcription control. In *Proc. IEEE 2002 Conf. Decision and Control, CDC 2002*, Las Vegas, NV, 2002.
3. W.F. Burkholder and A.D. Grossman. Regulation of the initiation of endospore formation in *Bacillus subtilis*. In Y.V. Brun and L.J. Shimkets, editors, *Prokaryotic Development*, chapter 7, pages 151–166. ASM, 2000.
4. H. de Jong. Modeling and simulation of genetic regulatory systems: A literature review. *J. Comput. Biol.*, 9(1):69–105, 2002.
5. H. de Jong, J. Geiselmann, G. Batt, C. Hernandez, and M. Page. Qualitative simulation of the initiation of sporulation in *B. subtilis*. Technical Report RR-4527, INRIA, 2002.
6. H. de Jong, J. Geiselmann, C. Hernandez, and M. Page. Genetic Network Analyzer: Qualitative simulation of genetic regulatory networks. *Bioinformatics*, 2003. In press.
7. H. de Jong, J.-L. Gouzé, C. Hernandez, M. Page, T. Sari, and H. Geiselmann. Dealing with discontinuities in the qualitative simulation of genetic regulatory networks. In F. van Harmelen, editor, *Proc. Fifteenth Europ. Conf. Artif. Intell., ECAI 2002*, pages 412–416. IOS Press, 2002.
8. H. de Jong, J.-L. Gouzé, C. Hernandez, M. Page, T. Sari, and H. Geiselmann. Qualitative simulation of genetic regulatory networks using piecewise-linear models. Technical Report RR-4407, INRIA, 2002.
9. H. de Jong, M. Page, C. Hernandez, and J. Geiselmann. Qualitative simulation of genetic regulatory networks: Method and application. In B. Nebel, editor, *Proc. Seventeenth Int. Joint Conf. Artif. Intell., IJCAI-01*, pages 67–73. Morgan Kaufmann, 2001.
10. R. Edwards, H.T. Siegelmann, K. Aziza, and L. Glass. Symbolic dynamics and computation in model gene networks. *Chaos*, 11(1):160–169, 2001.
11. P. Fawcett, P. Eichenberger, R. Losick, and P. Youngman. The trancriptional profile of early to middle sporulation in *Bacillus subtilis*. *Proc. Natl. Acad. Sci. USA*, 97(14):8063–8068, 2000.
12. A.F. Filippov. *Differential Equations with Discontinuous Righthand Sides*. Kluwer Academic Publishers, 1988.

13. R. Ghosh and C.J. Tomlin. Lateral inhibition through Delta-Notch signaling: A piecewise affine hybrid model. In M.D. Di Benedetto and A. Sangiovanni-Vincentelli, editors, *Hybrid Systems: Computation and Control (HSCC 2001)*, volume 2034 of *LNCS*, pages 232–246. Springer-Verlag, Berlin, 2001.
14. L. Glass and S.A. Kauffman. The logical analysis of continuous non-linear biochemical control networks. *J. Theor. Biol.*, 39:103–129, 1973.
15. J.-L. Gouzé and T. Sari. A class of piecewise linear differential equations arising in biological models. *Dynam. Syst.*, 2003. To appear.
16. A.D. Grossman. Genetic networks controlling the initiation of sporulation and the development of genetic competence in *Bacillus subtilis*. *Ann. Rev. Genet.*, 29:477–508, 1995.
17. J. Hasty, D. McMillen, F. Isaacs, and J.J. Collins. Computational studies of gene regulatory networks: *in numero* molecular biology. *Nat. Rev. Genet.*, 2(4):268–279, 2001.
18. R. Heinrich and S. Schuster. *The Regulation of Cellular Systems*. Chapman & Hall, New York, 1996.
19. J.A. Hoch. Regulation of the phosphorelay and the initiation of sporulation in *Bacillus subtilis*. *Ann. Rev. Microbiol.*, 47:441–465, 1993.
20. B. Kuipers. *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge*. MIT Press, 1994.
21. H.H. McAdams and A. Arkin. Simulation of prokaryotic genetic circuits. *Ann. Rev. Biophys. Biomol. Struct.*, 27:199–224, 1998.
22. T. Mestl, E. Plahte, and S.W. Omholt. A mathematical framework for describing and analysing gene regulatory networks. *J. Theor. Biol.*, 176:291–300, 1995.
23. P.J. Mosterman and G. Biswas. A comprehensive methodology for building hybrid models of physical systems. *Artif. Intell.*, 121:171–209, 2000.
24. P.J. Mosterman, F. Zhao, and G. Biswas. Sliding mode model semantics and simulation for hybrid systems. In P.J. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, and S. Sastry, editors, *Hybrid Systems V*, pages 19–32. Springer-Verlag, Berlin, 1999.
25. P. Smolen, D.A. Baxter, and J.H. Byrne. Modeling transcriptional control in gene networks: Methods, recent results, and future directions. *Bull. Math. Biol.*, 62:247–292, 2000.
26. E.H. Snoussi. Qualitative dynamics of piecewise-linear differential equations: A discrete mapping approach. *Dyn. Stabil. Syst.*, 4(3-4):189–207, 1989.
27. P. Stragier and R. Losick. Molecular genetics of sporulation in *Bacillus subtilis*. *Ann. Rev. Genet.*, 30:297–341, 1996.
28. 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. *Bull. Math. Biol.*, 57(2):247–276, 1995.
29. E.O. Voit. *Computational Analysis of Biochemical Systems: A Practical Guide for Biochemists and Molecular Biologists*. Cambridge University Press, Cambridge, 2000.