



## Genetic Network Analyzer: qualitative simulation of genetic regulatory networks

Hidde de Jong<sup>1,\*</sup>, Johannes Geiselmann<sup>2</sup>, Céline Hernandez<sup>1</sup>  
and Michel Page<sup>1,3</sup>

<sup>1</sup>INRIA Rhône-Alpes, 655 avenue de l'Europe, Montbonnot, 38334 Saint Ismier  
CEDEX, France, <sup>2</sup>PEGM, Université Joseph Fourier, Grenoble and <sup>3</sup>ESA, Université  
Pierre Mendès France, Grenoble

Received on February 1, 2002; revised on June 5, 2002; accepted on August 12, 2002

### ABSTRACT

**Motivation:** 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.

**Results:** We present Genetic Network Analyzer (GNA), a computer tool for the modeling and simulation of genetic regulatory networks. The tool is based on a qualitative simulation method that employs coarse-grained models of regulatory networks. The use of GNA is illustrated by a case study of the network of genes and interactions regulating the initiation of sporulation in *Bacillus subtilis*.

**Availability:** GNA and the model of the sporulation network are available at <http://www-helix.inrialpes.fr/gna>.

**Contact:** Hidde.de-Jong@inrialpes.fr

### INTRODUCTION

The study of *genetic regulatory networks* has taken a qualitative leap through the use of modern genomic techniques that allow simultaneous measurement of the expression levels of all genes of an organism. In addition to experimental tools, computer tools for the *modeling* and *simulation* of gene regulation processes will be indispensable. As most networks of interest involve many genes connected through interlocking positive and negative feedback loops, an intuitive understanding of their dynamics is difficult to obtain and may lead to erroneous conclusions. Modeling and simulation tools, with a solid foundation in mathematics and computer science, allow the behavior of large and complex systems to be predicted in a systematic way (e.g. de Jong, 2002).

Several computer tools for the simulation of biochemical reaction networks by means of differential equations are currently available, such as DBsolve (Goryanin *et al.*,

1999), GEPASI (Mendes, 1993), and PLAS (Voit, 2000). These tools can be used to simulate genetic, metabolic, and signal transduction networks described by differential equations. In addition, they allow the user to perform tasks like the stability analysis of steady states and the estimation of parameter values. The currently available tools are essentially restricted to *quantitative* models of reaction networks, in the sense that numerical values for the kinetic parameters and molecular concentrations need to be specified. However, since this information is usually absent, especially in the case of systems that are not well understood, the above-mentioned tools may be difficult to apply.

This paper presents *Genetic Network Analyzer (GNA)*, a computer tool for the qualitative simulation of genetic regulatory networks. GNA employs piecewise-linear (PL) differential equation models that have been well studied in mathematical biology (Glass and Kauffman, 1973; Mestl *et al.*, 1995; Snoussi, 1989). While abstracting from the precise molecular mechanisms involved, the PL models capture essential aspects of gene regulation. Their simple mathematical form permits a qualitative analysis of the dynamics of the genetic regulatory systems to be carried out. Instead of numerical values for parameters and initial conditions, GNA asks the user to specify qualitative constraints on these values in the form of algebraic inequalities. Unlike precise numerical values, these constraints can usually be inferred from the experimental literature.

GNA supports a qualitative simulation method that recasts the mathematical analysis of PL models of genetic regulatory networks in a computational form (de Jong *et al.*, 2002b). In this paper we will give only a brief outline of the simulation method, abstracting away advanced mathematical concepts and algorithmic details. We will focus on the computer tool supporting the method and its practical use. That is, we will explain how GNA can be used by biologists and bioinformaticians to formulate models of genetic regulatory networks, simulate the behavior emerging from the interactions in the network, and interpret the simulation results in biological terms.

\*To whom correspondence should be addressed.

Readers interested in details of the qualitative simulation method are referred to (de Jong *et al.*, 2002b).

The use of GNA will be illustrated in the context of a regulatory network of biological interest, consisting of the genes and interactions regulating the initiation of sporulation in the Gram-positive soil bacterium *Bacillus subtilis* (Burkholder and Grossman, 2000; Grossman, 1995; Hoch, 1993). Under conditions of nutrient deprivation, *B. subtilis* can decide not to divide and form a dormant, environmentally resistant spore instead. The decision to either divide or sporulate is controlled by a regulatory network integrating various environmental, cell-cycle, and metabolic signals. The aim of the example is to show that GNA is able to reproduce experimental findings in the case of a large and complex network that is well-understood by molecular biologists.

In the next section, we describe the PL models of genetic regulatory networks and the basics of the simulation method. We then discuss the computer tool GNA, followed by an illustration of its application to the sporulation network. The paper concludes with a discussion of the simulation tool and an outline of ideas for further work.

## QUALITATIVE SIMULATION OF GENETIC REGULATORY NETWORKS

### PL models

The dynamics of genetic regulatory networks can be modeled by a class of PL differential equations originally proposed by Glass and Kauffman (1973), and generalized by Mestl *et al.* (1995). The equations have the form

$$\dot{x}_i = f_i(\mathbf{x}) - g_i(\mathbf{x})x_i, \quad x_i \geq 0, \quad 1 \leq i \leq n, \quad (1)$$

where  $\mathbf{x} = (x_1, \dots, x_n)'$  is a vector of cellular protein concentrations. The *state equations* (1) define the rate of change of the concentration  $x_i$  as the difference of the rate of synthesis  $f_i(\mathbf{x})$  and the rate of degradation  $g_i(\mathbf{x})x_i$  of the protein.

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

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

where  $\kappa_{il}$  is a rate parameter ( $\kappa_{il} > 0$ ),  $b_{il} : \mathbb{R}_{\geq 0}^n \rightarrow \{0, 1\}$  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 (Mestl *et al.*, 1995; Thomas *et al.*, 1995).

In the simplest case,  $f_i(x_j) = \kappa_i s^+(x_j, \theta_j)$ , where  $s^+(x_j, \theta_j) = 0$ , if  $x_j < \theta_j$ , and  $s^+(x_j, \theta_j) = 1$ , if  $x_j > \theta_j$ . The function  $f_i(x_j)$  states that below a threshold concentration  $\theta_j$  ( $\theta_j > 0$ ), gene  $i$  is not expressed, whereas above this threshold it is expressed at a rate  $\kappa_i$ .

If protein  $J$ , encoded by gene  $j$ , is a negative regulator of gene  $i$ , we have  $f_i(x_j) = \kappa_i s^-(x_j, \theta_j)$ , with  $s^-(x_j, \theta_j) = 1 - s^+(x_j, \theta_j)$ . More complex regulation functions can express the combined effects of several regulatory proteins (de Jong *et al.*, 2002a). The use of step functions in gene regulation models has been motivated by the observation that the rate of activation of a gene, as a function of the concentration of a regulatory protein, often follows a steep sigmoidal curve.

The function  $g_i$  expresses the regulation of protein degradation. It is defined analogously to  $f_i$ , except that we demand that  $g_i(\mathbf{x})$  is strictly positive. In addition, in order to formally distinguish degradation rates from synthesis rates, we will denote the former by  $\gamma$  instead of  $\kappa$ . As can be easily verified, with the above definitions of  $f_i$  and  $g_i$ , the differential equations (1) are *piecewise linear (PL)* (Glass and Kauffman, 1973; Mestl *et al.*, 1995; Snoussi, 1989).

### Properties of PL models

The dynamical properties of the PL models can be analyzed in the  $n$ -dimensional phase space box  $\Omega = \Omega_1 \times \dots \times \Omega_n$ , where every  $\Omega_i$ ,  $1 \leq i \leq n$ , is defined as  $\Omega_i = \{x_i \in \mathbb{R}_{\geq 0} \mid 0 \leq x_i \leq \max_i\}$ .  $\max_i$  is a parameter denoting a maximum concentration for the protein. Given that the protein encoded by gene  $i$  has  $p_i$  threshold concentrations, the  $n - 1$ -dimensional threshold hyperplanes  $x_i = \theta_i^{k_i}$ ,  $1 \leq k_i \leq p_i$ , partition  $\Omega$  into hyperrectangular regions that are called *domains*. More precisely, a domain  $D \subseteq \Omega$  is defined by  $D = D_1 \times \dots \times D_n$ , where every  $D_i$ ,  $1 \leq i \leq n$ , is given by one of the following equations:

$$\begin{aligned} D_i &= \{x_i \mid 0 \leq x_i < \theta_i^1\}, \\ D_i &= \{x_i \mid x_i = \theta_i^1\}, \\ D_i &= \{x_i \mid \theta_i^1 < x_i < \theta_i^2\}, \\ &\dots, \\ D_i &= \{x_i \mid \theta_i^{p_i} < x_i \leq \max_i\}. \end{aligned} \quad (3)$$

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*. Otherwise,  $D$  is called a *regulatory domain*.

When evaluating the step function expressions in (2) in a regulatory domain,  $f_i$  and  $g_i$  reduce to sums of rate constants. More precisely, in every regulatory domain  $D \subseteq \Omega$ ,  $f_i$  reduces to some  $\mu_i^D \in M_i \equiv \{f_i(\mathbf{x}) \mid \mathbf{0} \leq \mathbf{x} \leq \mathbf{max}\}$ , and  $g_i$  to some  $\nu_i^D \in N_i \equiv \{g_i(\mathbf{x}) \mid \mathbf{0} \leq \mathbf{x} \leq \mathbf{max}\}$ .  $M_i$  and  $N_i$  collect the synthesis and degradation rates of the protein in different domains of  $\Omega$ . It can be easily shown that all trajectories in  $D$  monotonically tend towards a stable steady state  $\mathbf{x} = \boldsymbol{\mu}^D / \boldsymbol{\nu}^D$ , the *target equilibrium*, lying at the intersection of the  $n$  hyperplanes

$x_i = \mu_i^D/v_i^D$  (Glass and Kauffman, 1973; Mestl *et al.*, 1995; Snoussi, 1989). The target equilibrium level  $\mu_i^D/v_i^D$  of the protein concentration  $x_i$  gives an indication of the strength of gene expression in  $D$ . Call  $\Phi(D) = \{\mu^D/v^D\}$  the *target equilibrium set* of  $D$ . If  $\Phi(D) \cap D \neq \{\}$ , then all trajectories will remain in  $D$ . If not, they will leave  $D$  at some point.

In switching domains,  $f_i$  and  $g_i$  may not be defined, because some concentrations assume their threshold value. Moreover,  $f_i$  and  $g_i$  may be discontinuous in switching domains. In order to cope with this problem, the system of differential equations (1) is extended into a system of differential inclusions, following an approach widely used in control theory (Gouzé and Sari, 2002). 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  $\Phi(D)$ . Here,  $\Phi(D)$  is the smallest closed convex set including the target equilibria of regulatory domains having  $D$  in their boundary, intersected with the hyperplane containing  $D$ . If  $\Phi(D) \cap D \neq \{\}$ , then the trajectories may remain in  $D$ . If not, they will leave  $D$  at some point.

### Qualitative constraints on parameters

Most of the time, precise numerical values for the threshold and rate parameters in (1) are not available. Rather than numerical values, we specify qualitative constraints on the parameter values. These constraints, having the form of algebraic inequalities, can usually be inferred from biological data. They are exploited by the simulation method to predict the qualitative dynamics of the regulatory system.

The first type of constraints, the *threshold inequalities*, are obtained by ordering the  $p_i$  threshold concentrations of the protein encoded by gene  $i$ , i.e.,

$$0 < \theta_i^1 < \dots < \theta_i^{p_i} < \max_i, \quad (4)$$

The threshold inequalities determine the partitioning of  $\Omega$  into regulatory and switching domains.

The second type of constraints, the *equilibrium inequalities*, order the possible target equilibrium levels of  $x_i$  in different regulatory domains  $D \subseteq \Omega$  with respect to the threshold concentrations. Biologically speaking, the 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  $\mu_i \in M_i$ ,  $v_i \in N_i$ , and  $\mu_i, v_i \neq 0$ , we specify one of the following pairs of inequalities:

$$\begin{aligned} 0 &< \mu_i/v_i < \theta_i^1, \\ \theta_i^1 &< \mu_i/v_i < \theta_i^2, \\ &\dots \\ \theta_i^{p_i} &< \mu_i/v_i < \max_i. \end{aligned} \quad (5)$$

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

The models of genetic regulatory networks treated by the simulation method consist of state equations (1), supplemented by parameter inequalities (4) and (5). Every such *qualitative* PL model corresponds to a set of *quantitative* PL models consisting of state equations (1) and a particular combination of numerical parameter values consistent with the parameter inequalities.

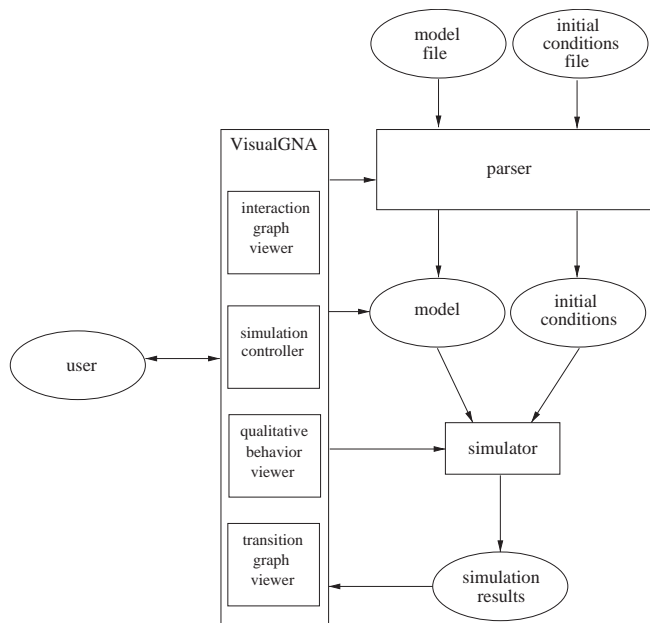
### Qualitative simulation

Let  $x$ , 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  $t$ ,  $0 \leq t < \tau$ ,  $x(t) \in D$ . A qualitative description of  $x$  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  $D^0, \dots, D^m$  in  $\Omega$ . Whenever  $x$  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.

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. More precisely, denoting by  $X$  the set of solutions  $x(t)$  of all quantitative PL models corresponding to the qualitative model, such that  $x(0) = x_0 \in D^0$ , the aim of qualitative simulation is to find the set of qualitative behaviors abstracting from some  $x \in X$ .

In de Jong *et al.* (2002b) 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  $D^0$ . This 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.

A sequence of qualitative states in the transition graph represents a predicted qualitative behavior of the system. It has been demonstrated that the transition graph generated by the simulation algorithm covers all qualitative behaviors abstracting from some  $x \in X$  (de Jong *et al.*, 2002b). That is, whatever the exact numerical values for the parameters may 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.



**Fig. 1.** The global architecture of GNA.

## GENETIC NETWORK ANALYZER (GNA)

The qualitative simulation method has been implemented in Java 1.3 in the program GNA. The global architecture of GNA is shown in Figure 1. 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 (Figure 2). The current version of GNA consists of about 110 classes, one third of which concerns *VisualGNA*.

We have chosen Java to implement the qualitative simulation method, because it has become a standard for object-oriented programming and is widely used for biological applications. In addition, the powerful graphical capabilities provided by Swing and other graphical libraries facilitate the development of the GUI, which is a critical component of GNA. The choice for Java is further motivated by its portability, allowing GNA to be used without special adaptations on different platforms (the program has been tested under Windows NT 4.0, 98, 2000, Solaris 6, and Linux 2.4.2). GNA is available for non-profit academic research purposes at <http://www-helix.inrialpes.fr/gna>.

### Parser

The simulation of a genetic regulatory network requires the user to prepare text files containing the PL model

and the initial conditions. For each protein, the *model file* contains declarations of the state variable and the production, degradation, and threshold parameters, as well as definitions of the state equations and threshold and equilibrium inequalities. A fragment of the model file for the sporulation network is shown in Figure 2(a). The *initial conditions file* contains the inequalities defining the initial domain (not shown).

The input files are read and compiled by a *parser* generated by Java CUP. The parser transforms the information in the input file into internal representations of the model and initial conditions, which are accessed by the simulator and the GUI.

### Simulator

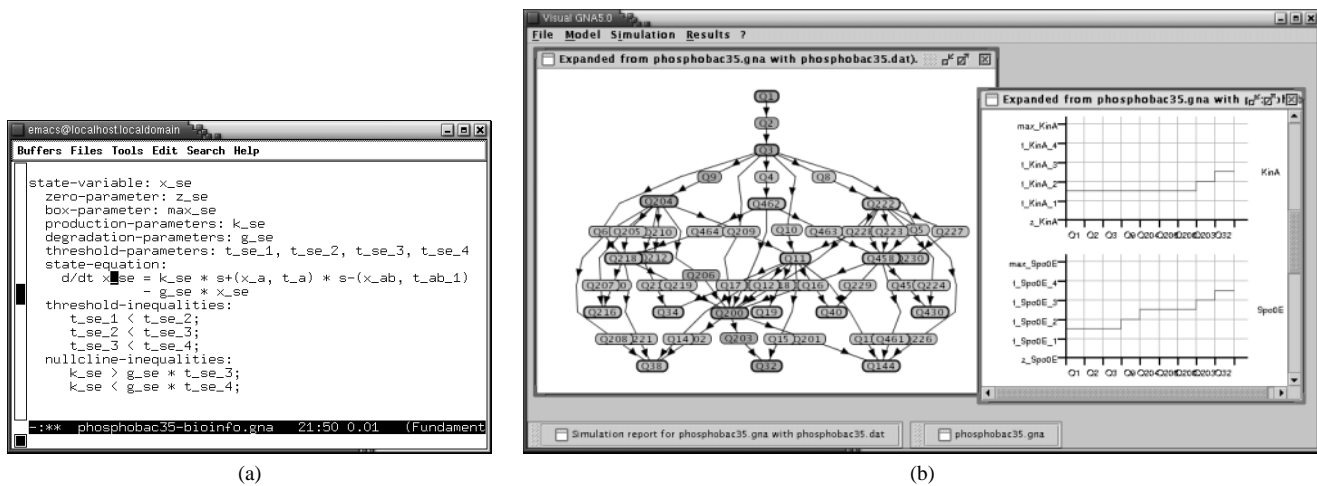
The essential tasks of the simulator consist in computing the qualitative states associated with a domain and in determining the possible transitions from these states. Instead of performing extensive numerical calculations, the simulator reaches its goal through symbolic computation, by exploiting the parameter inequalities (4)–(5) (de Jong *et al.*, 2002b). Whereas previous versions of GNA used a simple inequality reasoner to achieve this, the current version employs an integer coding scheme that permits more efficient computation.

The output of the simulator is a transition graph consisting of qualitative states and transitions between qualitative states. Various graph algorithms operate on the internal representation of this graph to detect, among other things, qualitative equilibrium states and qualitative cycles.

### Graphical user interface: VisualGNA

*VisualGNA*, the GUI component of GNA, actually consists of a simulation controller, an interaction graph viewer, a transition graph viewer, and a qualitative behavior viewer (Figure 1).

The *simulation controller* allows the user to load a model, launch a simulation, and analyze the results in a menu-driven way. The *interaction graph viewer* extracts the network structure from the model and visualizes it in the form of an interaction graph. The transition graph and qualitative behavior viewers help the user in interpreting the output of a qualitative simulation. The *transition graph viewer* displays the transition graph with the qualitative states and transitions generated by GNA. In addition, it offers the facility to focus on interesting parts of the graph by selecting, filtering, and expanding subgraphs. The *qualitative behavior viewer* explores the contents of the transition graph by visualizing a sequence of qualitative states as a set of concentration profiles, one for each protein. Alternatively, this information can be displayed as a sequence of projections on the interaction graph, providing an intuitive representation of the global change



**Fig. 2.** Modeling and simulation of a genetic regulatory network by means of GNA. (a) Excerpt of the model file of the sporulation network (see de Jong *et al.* (2002a) for the complete model). The figure shows the parameter declarations, state equations, and parameter inequalities concerning the state variables  $x_{se}$  (concentration Spo0E). (b) Screen capture of GNA. The window on the left shows a part of the state transition graph resulting from a simulation of the network under initial conditions inducing sporulation, while the window on the right visualizes the temporal sequence of qualitative states in one selected path in the state transition graph.

of gene expression over time. Figure 2(b) illustrates some of the functionalities of VisualGNA (for more details, see the HTML documentation of GNA).

## MODELING AND SIMULATION: INITIATION OF SPORULATION IN *B.SUBTILIS*

The use of GNA will be illustrated by modeling and simulating the regulatory network underlying the initiation of sporulation in *B.subtilis*. The modeling and simulation process consists of four steps: (i) collecting information on relevant genes, proteins, and regulatory interactions; (ii) describing the network by a qualitative PL model; (iii) simulating the system from appropriate initial conditions; and (iv) interpreting the simulation results. Although presented in a linear fashion below, in practice the four steps are usually carried out iteratively.

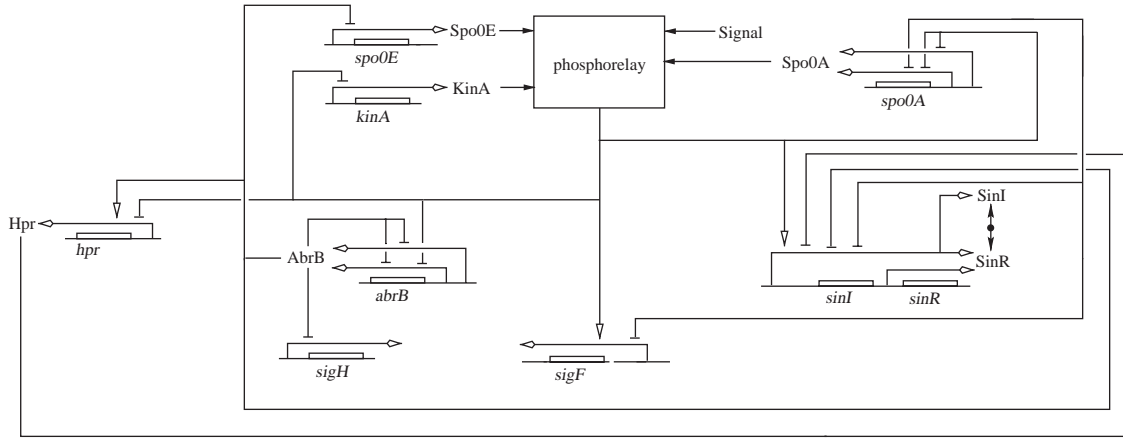
Sporulation in *B.subtilis* is one of the best-understood model systems for prokaryotic development. However, notwithstanding the enormous amount of work devoted to the elucidation of the network of interactions underlying the sporulation process, very little quantitative data on kinetic parameters and 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 (de Jong *et al.*, 2002a). This provides a validation of the tool in the case of a large and complex genetic regulatory network.

## Sporulation network

While nutrients are plentiful, *B.subtilis* cells divide as fast as possible in order to efficiently compete with their neighbors. However, when conditions become unfavorable, the bacterium protects itself by forming environmentally resistant spores (Burkholder and Grossman, 2000; Grossman, 1995; Hoch, 1993; Stragier and Losick, 1996). A graphical representation of the regulatory network controlling the initiation of sporulation is shown in Figure 3, 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 de Jong *et al.* (2002a).

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 Spo0A~P may reach a threshold value above which it activates various genes that commit the bacterium to sporulation. The choice between vegetative growth and



**Fig. 3.** Network of key genes, proteins, and regulatory interactions involved in *B. subtilis* sporulation. The graphical representation follows the conventions proposed in Kohn (2001).

sporulation in response to adverse environmental conditions is the outcome of competing positive and negative feedback loops, controlling the accumulation of Spo0A~P (Grossman, 1995; Hoch, 1993).

### Modeling of the sporulation network

For modeling, the graphical representation of Figure 3 has to be translated into state equations (1), supplemented by parameter inequalities (4) and (5). The Spo0E protein serves as an example to illustrate the procedure.

The phosphatase Spo0E involved in the phosphorelay is encoded by the gene *spo0E* (Perego and Hoch, 1987). Transcription of *spo0E* is directed by  $E\sigma^A$  (Perego and Hoch, 1987) and repressed by AbrB (Perego and Hoch, 1991; Strauch *et al.*, 1989). Spo0E degradation is not known to be regulated. Denoting by  $x_{se}$  the Spo0E concentration, and by  $\kappa_{se}$  and  $\gamma_{se}$  the synthesis and degradation rate constants, we obtain

$$\dot{x}_{se} = \kappa_{se} s^+(x_a, \theta_a) s^-(x_{ab}, \theta_{ab}^1) - \gamma_{se} x_{se}. \quad (6)$$

The state equation expresses that *spo0E* is transcribed at a rate  $\kappa_{se}$ , if the concentration  $x_{ab}$  of AbrB is below the threshold  $\theta_{ab}^1$  ( $s^-(x_{ab}, \theta_{ab}^1) = 1$ ), while the concentration of  $\sigma^A$  is above the threshold  $\theta_a$  ( $s^+(x_a, \theta_a) = 1$ ).

Four threshold concentrations are defined for Spo0E,  $\theta_{se}^1, \dots, \theta_{se}^4$ , indicating four levels of activation of the phosphorelay. The thresholds are ordered by the threshold inequalities

$$0 < \theta_{se}^1 < \theta_{se}^2 < \theta_{se}^3 < \theta_{se}^4 < \max_{se}. \quad (7)$$

The possible synthesis and degradation rates of Spo0E are given by  $M_{se} = \{0, \kappa_{se}\}$  and  $N_{se} = \{\gamma_{se}\}$ , respectively. We will set  $\kappa_{se}/\gamma_{se} > \theta_{se}^3$ . If this were not the case,

Spo0E would not be able to reach a concentration at which it can exert any influence on the sporulation decision (Perego and Hoch, 1991). In fact, above its threshold concentration  $\theta_{se}^3$ , Spo0E is able to reduce the activation of the phosphorelay to the point that the expression of  $\sigma^F$ , and hence the initiation of sporulation, is inhibited (de Jong *et al.*, 2002a). This motivates the following equilibrium inequalities:

$$\theta_{se}^3 < \kappa_{se}/\gamma_{se} < \theta_{se}^4. \quad (8)$$

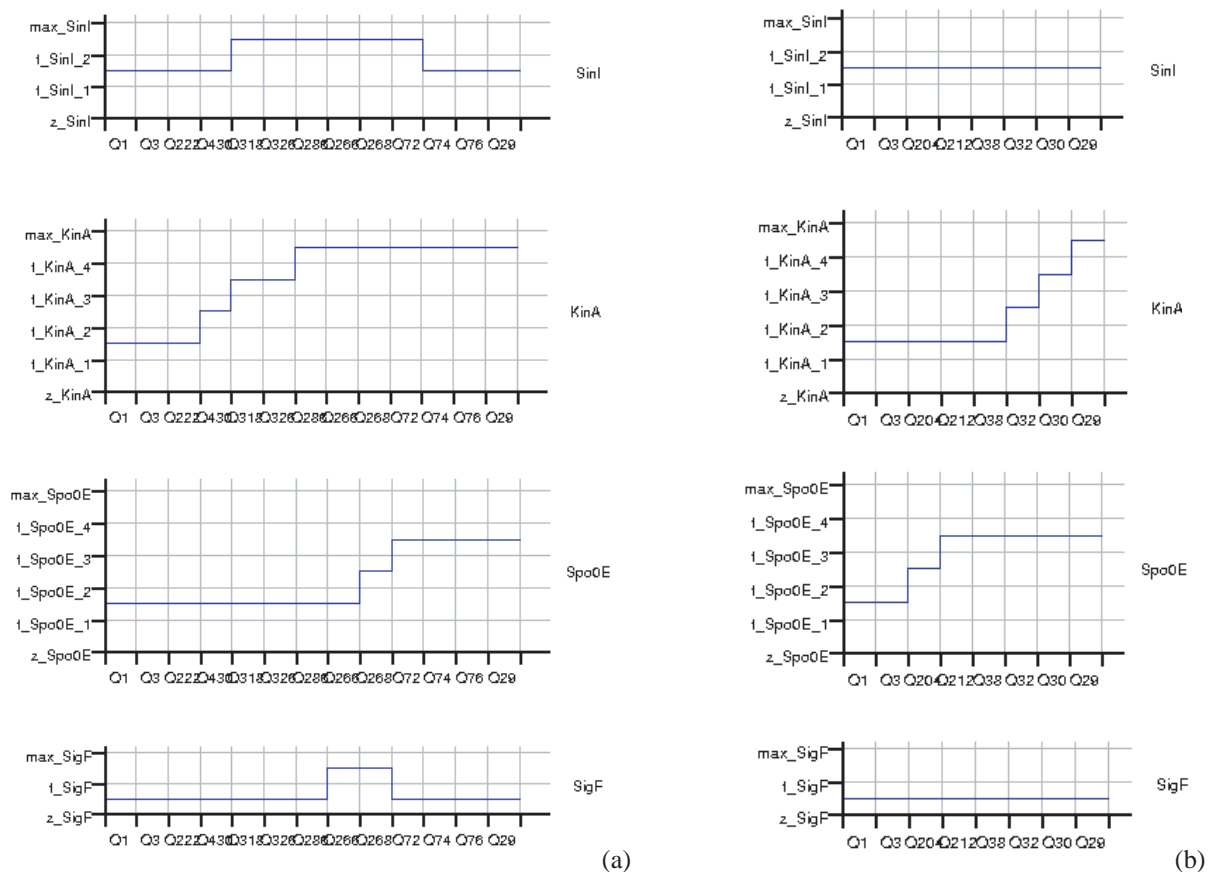
Figure 2(a) shows the model file entry for Spo0E.

The state equations and parameter inequalities for the other proteins in the network of Figure 3 can be formulated in an analogous way (de Jong *et al.*, 2002a). The resulting model consists of nine state variables and two input variables. The state equations are often complex expressions of step functions, reflecting the dense pattern of interactions in the regulatory network. The 49 parameters are constrained by 58 threshold and equilibrium inequalities, the choice of which is determined by biological data. If the parameter inequalities cannot be unambiguously determined, all possible combinations have to be systematically tested.

### Simulation of wild-type and mutant bacteria

GNA has been used to simulate the dynamics of the initiation of sporulation. 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), 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



**Fig. 4.** Results of simulating wild-type strain under sporulation conditions. The pictures have been generated by GNA. (a) Temporal evolution of selected protein concentrations in a typical qualitative behavior leading to the  $spo^+$  phenotype. (b) Idem, but for a typical qualitative behavior leading to the  $spo^-$  phenotype.

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 the left window in Figure 2(b). The transition graph contains a single qualitative equilibrium state that the system eventually reaches in all qualitative behaviors.

A crucial test for the pertinence and validity of our model consists in analyzing the behavior of sporulation mutants. These can be modeled by adapting the state equations. In the case of a *sinI* deletion, for example, we have  $\dot{x}_{si} = -\gamma_{si} x_{si}$ , with  $x_{si}$  denoting the cellular concentration of SinI. The equation expresses that in the absence of the *sinI* gene, SinI cannot be synthesized. Simulating the behavior of the *sinI* mutant under sporulation conditions produces a reduced transition graph of 31 qualitative states (not shown). In (de Jong et al., 2002a) a dozen examples of the simulation of *B.subtilis* sporulation mutants are discussed.

### Analysis of simulation results

The simulated behavior of the network should reflect the essential biological characteristics of the sporulation initiation process. Analysis of the state transition graphs by means of VisualGNA allows one to investigate in detail the predicted qualitative equilibrium state, as well as qualitative behaviors leading to the equilibrium state.

Consider first the state transition graph for the wild-type strain. The qualitative equilibrium state corresponds to the  $spo^-$  phenotype, because the concentration of  $\sigma^F$ , a sigma factor essential for the development of the forespore (Stragier and Losick, 1996), has not reached the threshold above which it directs the transcription of its target genes. Whereas in some qualitative behaviors the system directly reaches the qualitative equilibrium state, in others it first passes through a period of transitory *sigF* expression, which corresponds to the  $spo^+$  phenotype.

A typical qualitative behavior for sporulation as well as for vegetative growth are shown in Figure 4. For ease

of presentation, only four of the nine state variables are represented. The initial phosphorylation of Spo0A leads to downregulation of *abrB*, which favors expression of *sigH* and *sinI*. Increasing levels of  $\sigma^H$  and SinI lead to an increase in *spo0A* and *kinA* expression. As a consequence, Spo0A~P levels continue to rise and activate *sigF* after some time, as can be seen in (a). On the other hand, as shown in (b), *abrB* downregulation also causes the Spo0E concentration to rise, which prevents Spo0A~P from accumulating to the point where *sinI* is expressed at a level sufficient to inactivate SinR. This leaves SinR repression of the  $\sigma^H$ -dependent promoter of Spo0A intact, and thus prevents the initiation of sporulation.

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. 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 outpaces the rate of accumulation of the phosphatase, we observe transient expression of *sigF*, i.e. a *spo*<sup>+</sup> phenotype (Figure 4(a)). If the kinetics of these processes are inverted, *sigF* is never activated and we observe a *spo*<sup>-</sup> phenotype (Figure 4(b)).

The expression of *sigF* in the sporulation behavior is transitory, because the concentration of Spo0E continues to rise and therefore the concentration of Spo0A~P eventually falls below the level required for *sigF* activation (Figure 4(b)). The reversion of the sporulation decision through the continuing accumulation of Spo0E is due to the fact that the genes and interactions required for later stages of sporulation have not been included in our model.

In the case of the *sinI* deletion strain, no expression, not even transitory, of *sigF* occurs. That is, a *sinI* mutant is not predicted to sporulate, an observation supported by biological data (Bai and Mandić-Mulec, 1993). Deletion of *sinI* disrupts the positive feedback loops, because SinR repression of the  $\sigma^H$ -dependent promoter of Spo0A cannot be relieved, thus preventing further Spo0A~P accumulation.

## DISCUSSION

We have presented the computer tool GNA for the qualitative simulation of genetic regulatory networks and illustrated its use in the analysis of the network of interactions controlling the initiation of sporulation in *B.subtilis*. GNA implements a simulation method that is based on a class of PL differential equation models described in mathematical biology (de Jong *et al.*, 2002b). While abstracting from precise molecular mechanisms, the PL models capture essential aspects of gene regulation. Instead of giving numerical values to the parameters

and initial conditions, which are usually not available, we use qualitative constraints in the form of algebraic inequalities. These are obtained by directly translating biological data into a mathematical formalism.

The predictions obtained through qualitative simulation have the form of a graph of qualitative states and transitions between qualitative states. A path of qualitative states in the transition graph is a prediction of the qualitative shape of the solution obtained for some combination of parameter values consistent with the parameter inequalities. Although the predictions obtained through qualitative simulation are less precise than their numerical counterparts, it has been demonstrated that no solution of a quantitative model subsumed by the qualitative model is omitted. That is, the qualitative behavior abstracted from each such solution is covered by the transition graph (de Jong *et al.*, 2002b). GNA thus provides guaranteed coverage of the solutions of the class of quantitative models consistent with the qualitative model, without performing exhaustive numerical simulation.

The lack of quantitative information on kinetic parameters and molecular concentrations has stimulated an interest in other methods and tools for qualitative simulation (de Jong, 2002). For example, the method QSIM (Kuipers, 1994) has been used to model and simulate  $\lambda$  phage growth in *E.coli* (Heidtke and Schulze-Kremer, 1998). A major problem in the application of qualitative simulation approaches is their lack of upscalability. GNA differs from traditional qualitative simulators in that it has been tailored to a class of PL models with favorable mathematical properties. This allows it to deal with large and complex genetic regulatory networks, as illustrated by the sporulation example. In comparison with the logical method of Thomas and colleagues (Thomas *et al.*, 1995), which is based on comparable abstractions of regulatory interactions, GNA has been developed for differential equation models. We believe that the latter formalism is intuitively clear and of large generality. Moreover, it facilitates the integration of any quantitative data becoming available through advances in measurement technologies.

The graphical user interface of GNA allows the user to analyze the simulation results, by manipulating the transition graph and by exploring the predicted qualitative behaviors of the system on the level of individual protein concentrations. GNA also visualizes the interactions between the genes and proteins in the regulatory network, as described by the model. In order to facilitate the specification of the model and initial conditions, a graphical model editor is currently under development.

Simulation of the sporulation network by means of GNA reveals that essential features of the initiation of sporulation can be reproduced by means of a model constructed from the experimental literature (de Jong *et al.*, 2002a). 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 regulatory 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, Eric Gauthier, Jean-Luc Gouzé, Sébastien Maza, Anne Morgat, Sébastien Provencher, François Rechenmann, Marie-France Sagot, Tewfik Sari, Denis Thieffry, Ivayla Vatcheva, Alain Viari, and three anonymous referees for comments on previous versions of this paper and other contributions.

## REFERENCES

- Bai,U. and Mandić-Mulec,I. (1993) SinI modulates the activity of SinR, a developmental switch protein of *Bacillus subtilis*, by protein-protein interaction. *Genes Dev.*, **7**, 139–148.
- Burkholder,W.F. and Grossman,A.D. (2000) Regulation of the initiation of endospore formation in *Bacillus subtilis*. In Brun,Y.V. and Shimkets,L.J. (eds), *Prokaryotic Development*. ASM, pp. 151–166.
- de Jong,H., Geiselmann,J., Batt,G., Hernandez,C. and Page,M. (2002a) Qualitative simulation of the initiation of sporulation in *B.subtilis*. *Technical Report RR-4527*. INRIA.
- de Jong,H., Gouzé,J.-L., Hernandez,C., Page,M., Sari,T. and Geiselmann,H. (2002b) Qualitative simulation of genetic regulatory networks using piecewise-linear models. *Technical Report RR-4407*. INRIA.
- de Jong,H. (2002) Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.*, **9**, 69–105.
- Glass,L. and Kauffman,S.A. (1973) The logical analysis of continuous non-linear biochemical control networks. *J. Theor. Biol.*, **39**, 103–129.
- Goryanin,I., Hodgman,T.C. and Selkov,E. (1999) Mathematical simulation and analysis of cellular metabolism and regulation. *Bioinformatics*, **15**, 749–758.
- Gouzé,J.-L. and Sari,T. (2002) A class of piecewise linear differential equations arising in biological models. *Dyn. Syst.*, in press.
- Grossman,A.D. (1995) Genetic networks controlling the initiation of sporulation and the development of genetic competence in *Bacillus subtilis*. *Annu. Rev. Genet.*, **29**, 477–508.
- Heidtke,K.R. and Schulze-Kremer,S. (1998) Design and implementation of a qualitative simulation model of  $\lambda$  phage infection. *Bioinformatics*, **14**, 81–91.
- Hoch,J.A. (1993) Regulation of the phosphorelay and the initiation of sporulation in *Bacillus subtilis*. *Ann. Rev. Microbiol.*, **47**, 441–465.
- Kohn,K.W. (2001) Molecular interaction maps as information organizers and simulation guides. *Chaos*, **11**, 1–14.
- Kuipers,B. (1994) *Qualitative Reasoning*. MIT Press.
- Mendes,P. (1993) GEPASI: a software package for modelling the dynamics, steady states and control of biochemical and other systems. *CABIOS*, **9**, 563–571.
- Mestl,T., Plahte,E. and Omholt,S.W. (1995) A mathematical framework for describing and analysing gene regulatory networks. *J. Theor. Biol.*, **176**, 291–300.
- Perego,M. and Hoch,J.A. (1987) Isolation and sequence of the *spo0E* gene: Its role in initiation of sporulation in *Bacillus subtilis*. *Mol. Microbiol.*, **1**, 125–132.
- Perego,M. and Hoch,J.A. (1991) Negative regulation of *Bacillus subtilis* sporulation by the *spo0E* gene product. *J. Bacteriol.*, **173**, 2514–2520.
- Snoussi,E.H. (1989) Qualitative dynamics of piecewise-linear differential equations. *Dyn. Stab. Syst.*, **4**, 189–207.
- Stragier,P. and Losick,R. (1996) Molecular genetics of *Bacillus subtilis*. *Annu. Rev. Genet.*, **30**, 297–341.
- Strauch,M.A., Spiegelman,G.B., Perego,M., Johnson,W.C., Burbulys,D. and Hoch,J.A. (1989) The transition state transcription regulator *abrB* of *Bacillus subtilis* is a DNA binding protein. *EMBO J.*, **8**, 1615–1621.
- Thomas,R., Thieffry,D. and Kaufman,M. (1995) Dynamical behaviour of biological regulatory networks: I. Biological rule of feedback loops. *Bull. Math. Biol.*, **57**, 247–276.
- Voit,E.O. (2000) *Computational Analysis of Biochemical Systems*. Cambridge University Press, Cambridge.