

Research

Open Access

A computational framework for qualitative simulation of nonlinear dynamical models of gene-regulatory networks

Liliana Ironi* and Luigi Panzeri

Address: IMATI-CNR, via Ferrata 1, 27100 Pavia, Italy

E-mail: Liliana Ironi* - ironi@imati.cnr.it; Luigi Panzeri - panzeri@imati.cnr.it

*Corresponding author

from Bioinformatics Methods for Biomedical Complex Systems Applications (NETTAB2008)
Varenna, Italy 19–21 May 2008

Published: 15 October 2009

BMC Bioinformatics 2009, 10(Suppl 12):S14 doi: 10.1186/1471-2105-10-S12-S14

This article is available from: <http://www.biomedcentral.com/1471-2105/10/S12/S14>

© 2009 Ironi and Panzeri; licensee BioMed Central Ltd.

This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/2.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

Background: Due to the huge amount of information at genomic level made recently available by high-throughput experimental technologies, networks of regulatory interactions between genes and gene products, the so-called *gene-regulatory networks*, can be uncovered. Most networks of interest are quite intricate because of both the high dimension of interacting elements and the complexity of the kinds of interactions between them. Then, mathematical and computational modeling frameworks are a must to predict the network behavior in response to environmental stimuli. A specific class of Ordinary Differential Equations (ODE) has shown to be adequate to describe the essential features of the dynamics of gene-regulatory networks. But, deriving quantitative predictions of the network dynamics through the numerical simulation of such models is mostly impracticable as they are currently characterized by incomplete knowledge of biochemical reactions underlying regulatory interactions, and of numeric values of kinetic parameters.

Results: This paper presents a computational framework for qualitative simulation of a class of ODE models, based on the assumption that gene regulation is threshold-dependent, i.e. only effective above or below a certain threshold. The simulation algorithm we propose assumes that threshold-dependent regulation mechanisms are modeled by continuous steep sigmoid functions, unlike other simulation tools that considerably simplifies the problem by approximating threshold-regulated response functions by step functions discontinuous in the thresholds. The algorithm results from the interplay between methods to deal with incomplete knowledge and to study phenomena that occur at different time-scales.

Conclusion: The work herein presented establishes the computational groundwork for a *sound* and a *complete* algorithm capable to capture the dynamical properties that depend only on the network structure and are invariant for ranges of values of kinetic parameters. At the current state of knowledge, the exploitation of such a tool is rather appropriate and useful to understand how specific activity patterns derive from given network structures, and what different types of dynamical behaviors are possible.

Background

Although, up to now, there is no model that efficiently and accurately represents the gene interactions underlying regulatory mechanisms in their whole complexity, recent experimental evidence seems to confirm the adequacy of a specific class of ODES to describe their essential dynamical features. These models assume that genes are controlled by transcription factors, and that the effect of a transcription factor on the transcription rate of a gene, *the response function*, is threshold-dependent. Such switch-like behaviors across variable thresholds are mathematically well represented by steep sigmoidal functions.

Although these models provide detailed description of gene regulatory molecular mechanisms [1-3], their predictive usefulness, at quantitative level, is rather limited even when the network at hand is very well studied. In fact, the exploitation of classical numerical approaches is mostly impracticable as precise and quantitative information on (i) the biochemical reaction mechanisms underlying regulatory interactions, and (ii) kinetic parameters and threshold concentrations are currently unknown and not identifiable from available data. However, at the current state of knowledge, qualitative predictions of the dynamical properties are not make-shift solutions but rather appropriate to get insight into the functioning of systems not completely understood as molecular interaction networks are.

The application of generic qualitative simulation approaches, as originally proposed in the Artificial Intelligence research framework [4] under the label QR, is not the proper solution. The mathematical tools they are grounded on are too simple to compensate for knowledge incompleteness. This results in a number of drawbacks, e.g. their inability to upscalability, the exponential growth of the generated behaviors, and the generation of spurious behaviors, that seriously limit the range of applicability of such methods to predict the nonlinear dynamics of regulatory networks. A qualitative study of GRNs dynamics could, in theory, be performed by analytical methods [5-8] based on the classical theory of qualitative analysis of dynamical systems, and properly adapted to the specific class of models. But, in practice, the network complexity makes quite hard or even unfeasible traditional analysis.

Pioneering work towards automated qualitative analysis and simulation of GRNs has resulted in a computational tool, called GNA[9]. Although GNA has been successfully applied to study real world networks, namely the initiation of sporulation in *Bacillus subtilis* [10], and the response to nutritional stress and carbon starvation in

Escherichia coli [11,12], it is not flawless. GNA circumvents the hard problem of dealing with sigmoidal nonlinear response functions by approximating them with step functions, discontinuous in the threshold hyperplanes. Such an assumption considerably simplifies the analysis as the model results in piecewise-linear equations, but it raises the problem to find a proper continuous solution across the threshold hyperplanes, or, in other words, to seek for generalized solutions of ODES with discontinuous right-side terms. The solution to this problem is not straightforward as (i) there exist in the literature several definitions of generalized solutions, (ii) it is not yet completely understood what are the relationships between different definitions, and then, (iii) it is not clear how to choose the "right" definition for a particular task [13]. GNA adopts the Filippov approach [14] that results particularly convenient to deal with control problems but it may present drawbacks when applied to approximate the limit solutions of a continuous ODE model: it might find "too many" solutions, and fail to reach all stable ones. Thus, GNA suffers from the same problems, that in addition to those raised by a further approximation introduced to deal with computational issues, might compromise its soundness and completeness (Dordan O, Ironi L, Panzeri L, Some critical remarks on GNA, *in preparation*).

The qualitative simulation algorithm we propose works under the assumptions that (i) threshold-dependent regulation mechanisms are modeled by continuous steep sigmoid functions, and (ii) any two genes are never regulated at the same threshold by a certain variable. The sigmoidal-nonlinearities make the problem quite hard to be tackled. But, the assumption that all sigmoids have very high steepness allows us to apply a systematic way of analysis. Let us observe that, due to the switch-like behavior of the response functions around the thresholds, the GRN dynamics occurs at different time-scales. To be able to deal with both slow and fast nonlinear dynamics we theoretically base our algorithm on a classical singular perturbation analysis method properly adapted to the assumed class of ODES [8,15]. Such a method suitably combined with QR key concepts computationally drives, starting from initial conditions, the construction of all possible state transitions, and calculates the sets of symbolic inequalities on parameter values that hold when specific transitions occur.

A class of models of GRN dynamics

As phenomenological model of the complex dynamics of GRNs made up of n components, we consider the following generic equations:

$$\dot{x}_i = f_i(\mathbf{Z}) - \gamma_i x_i \quad i = 1, \dots, n, \quad (1)$$

where the dot denotes time derivative, x_i is the concentration of the i -th gene product, $\gamma_i > 0$ is the decay rate of x_i , \mathbf{Z} is a vector with Z_{jk} as components, and $Z_{jk} = S(x_j, \theta_{jk}, q)$ is a sigmoid function with threshold θ_{jk} . The response, or regulatory, function $S : \mathbb{R}^+ \rightarrow [0, 1]$ is a continuous monotonic S-shaped map depending on the parameter q ($0 < q \ll 1$), that determines the steepness of S around the threshold value θ_{jk} , such that for $q \rightarrow 0$ we have $S(x_j, \theta_{jk}, q) = 0$ (respectively 1) when the value of x_j is smaller (greater) than θ_{jk} (Fig. 1).

Each x_{i_l} with domain $\Omega_i \subset \mathbb{R}^+$, is associated with n_i thresholds ordered according to $\theta_{ij} < \theta_{ik}$ if $j < k$. The state equations describe the balance between the production process $f_i(\mathbf{Z})$ and the degradation one, herein supposed to be linear. The functions f_i are multilinear polynomials in the variables Z_{jk} , and are frequently composed by algebraic equivalents of Boolean functions. More precisely,

$$f_i(\mathbf{Z}) = \sum_{l=1}^{L_i} \kappa_{il} \prod_{\substack{j=1, n \\ k=1, n_j}} Z_{jk}^{\alpha_{jkl}}, \quad (2)$$

where κ_{il} are real values that denote kinetic rate parameters, L_i is the possibly empty number of interactions that synthesize x_{i_l} and, in accordance with the network structure, α_{jkl} assumes value either equal to 1 when Z_{jk} takes part in the l -th interaction or equal to 0 otherwise.

The validity of the biological assumptions underlying the model described by Eq. (1) has been confirmed by

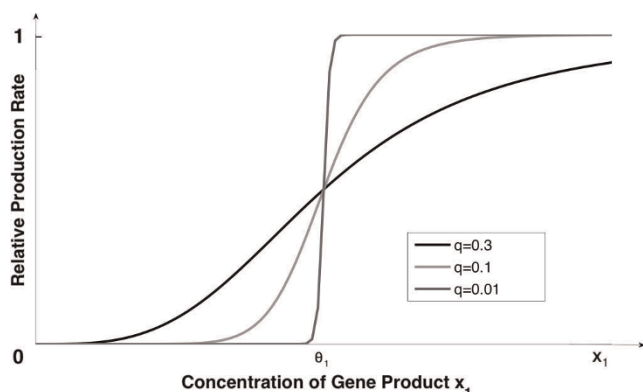


Figure 1
Response function shape. The sigmoid response function $S(x_1, \theta_1, q)$ describes the relationship between the concentration of a gene product x_1 and the relative production rate of the regulated gene. The parameter q determines its steepness around the threshold θ_1 .

recent experimental evidence [16-28] and theoretical studies [29-32]. Thus, we can reasonably assume that such a model is suitable to give a phenomenological description of a wide range of regulatory systems in which the combined effects of a series of genetic processes, e.g. transcriptional and translational regulation, protein-protein interactions, metabolic processes, etc., can be properly described by threshold-dependent response functions.

Let us represent the dynamics of the n state variables x_i , modeled by the Eq. (1) and associated with appropriate initial conditions, in the phase space. The ordered sets Θ_i of the n_i threshold values θ_{ij} associated with each x_i naturally induce a partition of the phase space into qualitatively distinct domains. In the set Δ of all the domains identified by the partition, we can distinguish the set $\Delta_s \subset \Delta$ of *switching domains* from the set $\Delta_r \subset \Delta$ of *regular domains*, such that $\Delta = \Delta_s \cup \Delta_r$ (Fig. 2).

A regular domain D_r (also called a *box*) is an open rectangular region between adjacent threshold hyperplanes in which the values of all response functions Z_{jk}

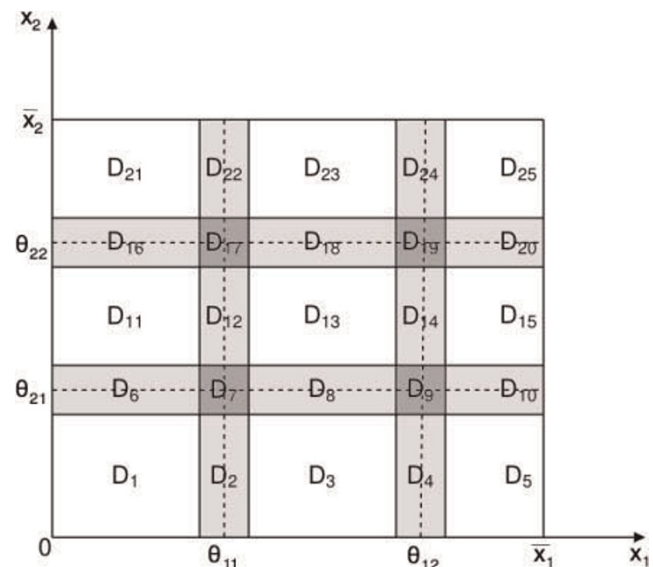


Figure 2
An example of phase space partition. Partition of the phase space associated to a dynamical system with two state variables, where each variable $x_i \in [0, \bar{x}_i]$, $i = 1, 2$ is associated with two ordered thresholds θ_{ij} , $j = 1, 2$. The white rectangles denote regular domains. The gray rectangles denote switching domains, whose width, $\delta > 0$ is a monotonic function of the parameter q with $\delta(q) \rightarrow 0$ for $q \rightarrow 0$. In the light gray regions one variable is switching and the other one is regular, whereas in the dark gray ones both variables are switching.

equal either 0 or 1. A switching domain D_s is a narrow region, whose width depends on the parameter q , surrounding either a section of a threshold hyperplane or an intersection of threshold hyperplanes. In a switching domain D_s we distinguish $\sigma(D_s)$ switching variables, x_s , from $n - \sigma(D_s)$ regular ones x_r . The values of a variables x_s lies in the neighborhood of one of its thresholds and Z_{sk} takes values in $(0,1)$, while the values of a variable x_r lies between two adjacent thresholds. The network dynamics in each domain $D \in \Delta$ is described by different models. The motion in $D_r \in \Delta_r$ is described by linear ODES, and its analysis is straightforward. The motion in $D_s \in \Delta_s$, or equivalently around the thresholds, is described by nonlinear equations, and occurs at different time-scales. Therefore, to capture the salient features of the nonlinear dynamics in a switching domain, and to determine how the trajectories cross it to move towards other domains, a specific mathematical method is required.

Methods

Singular perturbation analysis in the switching domains

Singular Perturbation Analysis (SPA) is a classical approach to study phenomena that occur at different time-scales [33]. The dynamics of such phenomena are described by ODES, associated with appropriate initial conditions, in which a small parameter ($0 < q \ll 1$) multiplies either one of the derivatives or higher order derivative. Let us indicate this initial value problem by \mathcal{M}_q , and by \mathcal{M}_0 the same problem where $q = 0$. As $q \rightarrow 0$, the solution of \mathcal{M}_q identifies a "small" region, called *boundary-layer region*, of non-uniform convergence to the solution of the *reduced system* \mathcal{M}_0 . The region of uniform-convergence of \mathcal{M}_q to \mathcal{M}_0 is called *outer region*. Taken together, the outer and boundary-layer solution approximate the solution of \mathcal{M}_q for small nonzero values of q .

Such a general approach is not directly applicable to our problem, but it must be tailored to it [8]. In outline, (i) the Eqs. (1) related to the x_s variables are rewritten into the standard form of SPA through a change of coordinate system, (ii) the boundary-layer and outer solutions are calculated in the new coordinates, and (iii) they are converted back into the usual frame of reference.

Let $\Sigma: \Omega \mapsto [0, 1]^n$ be the coordinate transformation that converts the x_s coordinates into the Z_s ones. As under our assumptions, $\frac{\partial Z_s}{\partial x_s} = \frac{1}{q} d_s(Z_s, x_s)$, where d_s is a continuous and limited function, we can rewrite Eq. (1) as in the following:

$$\begin{aligned} q\dot{Z}_s &= d_s(Z_s, x_s)(f_s(\mathbf{Z}_R, \mathbf{Z}_S) - \gamma_s x_s), \\ \dot{x}_r &= f_r(\mathbf{Z}_R, \mathbf{Z}_S) - \gamma_r x_r, \end{aligned} \tag{3}$$

where $\mathbf{Z}_S, \mathbf{Z}_R$ are the vectors of switching and regular variables Z_s and Z_r , respectively.

The fast dynamics in the boundary-layer is studied by suitably scaling the time variable, namely $\tau = t/q$. In the limit, the fast dynamics is obtained by solving the system:

$$\begin{aligned} Z'_s &= d_s(Z_s, \theta_s)(f_s(\mathbf{Z}_R, \mathbf{Z}_S) - \gamma_s \theta_s), \\ x'_r &= 0, \end{aligned} \tag{4}$$

associated with appropriate initial conditions, where the prime denotes the derivative with respect to τ . As \mathbf{Z}_R is constant in any $D_s \in \Delta_s$, we focus on the switching variables Z_s only. The system (4) has a manifold of stationary points, called *slow-manifold*, given by the solutions, for all s , of the stationary equations $Z'_s = 0$. We call *exit point set (EP)* the set of points in the slow-manifold that are stable and satisfy the Tikhonov-Levinson theorem, and we call *Z-cube* $\mathbf{Z}(D_s) = [0, 1]^{\sigma(D_s)}$ the frame of reference where we search for exit points.

The motion in the original time t along the exit points is described by the reduced system \mathcal{M}_0 . Thus, under the hypothesis that at least one exit point $\tilde{\mathbf{Z}}_S$ exists, the motion equations of regular variables, or equivalently the outer solution, is represented by:

$$\dot{x}_r = f_r(\tilde{\mathbf{Z}}_S, \mathbf{Z}_R) - \gamma_r x_r. \tag{5}$$

The problem (5) is linear, and then, given the initial conditions, the outer solution, that determines how the trajectories move along the x_r directions, is easily calculated.

The calculation of the slow-manifold generally leads to an heavy, nonlinear computational problem as it consists in the solution of a set of polynomial equations. In the current implementation we adopt a further assumption that sounds realistic:

Assumption A. Every gene product only regulates one gene at each of its thresholds

Such an assumption considerably simplifies the calculation of the slow-manifold. Mathematically, it implies that each Z_{jk} only occurs in one equation, and thus the terms $f_s(\mathbf{Z}_R, \mathbf{Z}_S)$ are linear.

Remark 1

The location of each exit point is crucial in our analysis as it indicates the next adjacent domains the trajectories are moving towards along the x_s directions.

Let us observe that stationary points always exist on the vertices of $\mathbf{Z}(D_s)$ as they are the roots of $d_s(Z_s, \theta_s) = 0$. The computational cost of the search for all the other exit points could be quite high, but it can be considerably reduced by checking first a necessary condition for the existence of a stationary point on the other elements of $\mathbf{Z}(D_s)$. Let F be a face or the interior of $\mathbf{Z}(D_s)$. In [15], it has been proved that necessary condition for the existence of a stationary point in F is that the Jacobian matrix $\mathbf{J}_F = \left(\frac{\partial f_i}{\partial Z_j} \right)$ restricted to the switching variables in F has a complete loop. This holds if and only if there is a non-zero loop involving all variables in \mathbf{J}_F , and can be checked by using concepts from graph theory.

Let $\tilde{\mathbf{Z}}$ be a stationary point located on $\tilde{F} \in \mathbf{Z}(D_s)$, and $\mathcal{L}_{\tilde{F}} = \{l : l \in \{1, \dots, \sigma(D_s)\}, \tilde{Z}_l \in \{0, 1\}\}$. $\tilde{\mathbf{Z}}$ is an exit point if (i) it is stable. If \tilde{F} is on the boundary of $\mathbf{Z}(D_s)$, it is required, in addition to (i), that (ii) $Z_l, \forall l \in \mathcal{L}_{\tilde{F}}$, heads towards \tilde{F} . The stability of $\tilde{\mathbf{Z}}$ is checked by analyzing the spectrum of the Jacobian matrix, and the condition (ii) is verified when, $\forall l \in \mathcal{L}_{\tilde{F}}$, the sign of Z'_l , in a neighborhood of $\tilde{\mathbf{Z}}$ is in agreement with the motion of Z_l towards \tilde{F} . More precisely, in a neighborhood of $\tilde{\mathbf{Z}}$, $Z'_l > 0$ and $\tilde{Z}_l = 1$ or $Z'_l < 0$ and $\tilde{Z}_l = 0$.

Remark 2

SPA works out in the limit $q \rightarrow 0$, but the calculated solution approximates the solution of Eq. (3) for sufficiently small q ($0 < q \ll 1$). Moreover, it can be proved that the Jacobian matrix is stable for $0 < q < \bar{q} \ll 1$. Thus, the exit points calculated in the limit are also valid for values of $q < \bar{q}$.

Remark 3

Under the Assumption \mathcal{A} , the reduced equations are always independent of the variables Z_s in the Eq. (4), and the two sets of equations are mutually independent. Thus, the behavior of the variables x_s in a Z -cube is completely independent of the values of variables x_r , and the motion in a switching domain may be studied by first analyzing the former variables, and then the latter ones.

Qualitative reasoning concepts

Among the generic qualitative approaches proposed in the literature, QSIM provides both the most suitable formalism and algorithm to represent and simulate models qualitatively abstracted from ODES [4]. Thus, we revise and *ad hoc* tailor its key concepts to our specific class of models.

Qualitative value

The real values of each state variable x_i with domain $\Omega_i = [0, \bar{x}_i]$ are discretized into a finite ordered set of values

qualitatively important from the biological point of view. In our context, the *qualitative value space* of each x_i is defined by the ordered set Θ_i made up of both its n_i threshold symbolic values and the endpoints of Ω_i . The partition of the whole system domain, induced by the sets $\theta_i, i = 1, \dots, n$ identifies qualitatively distinct hyper-rectangles D that define all possible *system qualitative values*.

Qualitative state

Let $A(D)$ be the set of domains adjacent to $D \in \Delta$. The *qualitative state* of D , $QS(D)$, is defined by all of its adjacent domains D_k towards which a transition from it is possible:

$$QS(D) = \{D_k \mid D_k \in A(D), D \rightarrow D_k\}.$$

Each transition from D identifies a domain next traversed by a system trajectory. More precisely, if we number by i the domain D traversed at time t_i , each $D_k \in QS(D)$ will be traversed by different trajectories at time t_{i+1} .

State transition

Qualitative simulation of network dynamics is achieved by iteratively applying local transition strategies from one domain to its adjacent domains. The possible transitions from any D are determined by different strategies according to whether $D \in \Delta_r$ or $D \in \Delta_s$.

In the case $D \in \Delta_r$, like in traditional QR methods and in GNA[9], possible transitions are determined by the signs of \dot{x}_i . As \dot{x}_i are defined by linear expressions, such signs are easily determined by exploiting the inequalities that define the parameter space domain.

In the case $D \in \Delta_s$, a sign-based strategy is not practicable as the expressions for \dot{x}_i are nonlinear. A convenient way to proceed is given by SPA: transitions from D towards adjacent D_k are determined by the locations of the exit points on the boundary elements of the associated $\mathbf{Z}(D)$. Due to Assumption \mathcal{A} , each boundary element of $\mathbf{Z}(D)$ may contain at most one exit point. But, as, in a qualitative context, knowledge incompleteness on the parameter values is expressed by coarse constraints, the boundary element of $\mathbf{Z}(D)$ where an exit point is located is not, in general, uniquely determined. Thus, unless the exit point is located in the interior of $\mathbf{Z}(D)$ and a stable solution is reached, the successors of D are not uniquely determined. However, through symbolic computation procedures, it is possible to calculate the set of inequalities, I_k^i , on parameter values that hold when the transition from D_i to D_k occurs. Then, the 3-tuple $\langle D_i, D_k, I_k^i \rangle$ clearly and uniquely identifies each path from D_i to D_k .

Qualitative behavior

A finite sequence of paths, where each path is clearly both linked and consistent with its predecessor and successor, defines a *qualitative behavior*: $QB = \langle D_0, I_0 \rangle, \langle D_0, D_1, I_1^0 \rangle, \dots, \langle D_k, D_i, I_i^k \rangle, \dots, \langle D_F, I_F \rangle$, where D_0 is the initial domain, and D_F either contains a stable fixed point or identifies a cycle, i.e it is an already visited domain. I_0 is the initial set of inequalities that defines the parameter space domain, and I_F the set of inequalities on parameter values associated with D_F .

Qualitative simulation

Starting from an initial domain D_0 and a set, I_0 , of symbolic inequalities on parameter values, qualitative simulation generates all possible state transitions, and represents them by a directed tree rooted in D_0 , $BT(D_0)$, where the vertices correspond to D_i , and the arcs, labeled by the inequalities I_j^i , to the transitions from D_i to D_j . Each branch in $BT(D_0)$ defines a qualitative trajectory QB from D_0 , that traverses specific domains and occurs when the values of parameters satisfy its related inequalities. Such a trajectory abstracts all those numerical solution of the ODE model with initial conditions x^0 varying in D_0 , and with kinetic parameter values fulfilling the inequalities.

Results

Given as input, (i) n symbolic state equations of the form (1); (ii) n qualitative value spaces $\Theta_i = \{\theta_{ij}\}$ of the state variables; (iii) $D_0 \in \Delta$; (iv) a set of symbolic inequalities I_0 on parameters defining a parameter space domain PSD_0 , the algorithm, through an iterative procedure initialized by $\langle D_0, I_0 \rangle$, provides as output the entire range of possible network dynamics represented by $BT(D_0)$. Its main steps are outlined in the following:

1. Define the qualitative values by partitioning the phase space into regular and switching domains.
2. Calculate the qualitative state $QS(D_i)$ of the current domain D_i .
 - (a) Calculate the symbolic state equations in D_i , and the set $A(D_i)$;
 - (b) Determine constraints I_k^i on parameters that need to be fulfilled to have a transition from D_i to $D_k \in A(D_i)$;
 - (c) Check consistency of I_k^i with I_0 , and build the path $e_{ik} = D_i \rightarrow D_k$;
3. Append $\langle D_i, D_k, I_k^i \rangle$ to $BT(D_0)$, and mark D_i as visited domain.
4. Repeat from step 2 for each not visited D_k .

Step 2, and in particular the calculation of the conditions on parameters I_k^i , is the core of the overall algorithm. A transition from D_i to D_k actually occurs, and then $D_k \in QS(D_i)$, only if the set I_k^i is consistent with the set I_0 , i.e. when it defines a not empty parameter space domain PSD_k^i such that $PSD_k^i \subseteq PSD_0$. The calculation of an inequality set I_k^i is performed by two distinct algorithms that implement a different transition strategy according to whether $D_i \in \Delta_r$ or $D_i \in \Delta_s$.

Moving from a regular domain

The algorithm in charge of the construction of the possible paths from regular domains is, in principle, similar to that one proposed by GNA, but it is more informative as it calculates the I_k^i s. In the limit, we indicate $D_i \in \Delta_r$ by the product $D_i = \prod_{j=1}^n (\theta_{ji}, \theta_{j(i+1)})$ where $(\theta_{ji}, \theta_{j(i+1)})$ denotes the interval of x_j in D_i .

(a) *Symbolic state equations in $D_i \in \Delta_r$.* In each box D_i , Z_{jk} equals either 0 or 1 in the step function limit. This simplifies Eq. (1) as they reduces to linear equations:

$$\dot{x}_j = \mu_j - \gamma_j x_j, \quad j = 1, \dots, n, \tag{6}$$

where μ_j depends on D_i , and is given by the sum of some κ_{js} s. From Eq. (6) we can easily find the focal point $x^* = \left\{ x_j^* = \frac{\mu_j}{\gamma_j} \right\}$ the trajectories are heading towards.

Herein, we assume that focal points do not belong to switching domains. If x^* belongs to D_i , there is a stable point in it. Such a stable point exists in D_i , i.e. $D_i \in QS(D_i)$, if the set of inequalities $(\theta_{ji} < \frac{\mu_j}{\gamma_j} < \theta_{j(i+1)}) \quad \forall j \in \{1, \dots, n\}$ exists and is consistent with I_0 . Otherwise, the trajectories move towards a switching domain D_k adjacent to D_i .

(b) *Constraints I_k^i on parameters.* $\forall D_k \in A(D_i)$, the algorithm calculates the set of inequalities on parameters I_k^i that need to be fulfilled to have a transition from D_i to D_k . As in D_i all the equations (6) are linear, and all the trajectories head towards a focal point x^* in a regular domain, such inequalities are calculated by imposing that the signs of state variable rates match the relative position of D_k with respect to D_i . We define the relative position of D_k with respect to D_i , indicated by $V(D_k, D_i) = \{v_j\}_{j=1}^n$ where $v_j \in \{-1, 0, 1\}$, through the comparison of the intervals defining D_i and D_k . I_k^i , initialized to I_0 , is updated, $\forall j \in \{1, \dots, n\}$, with either the inequality

$$\left(\frac{\mu_j}{\gamma_j} > \theta_{j(i+1)} \right) \text{ if } v_j = 1 \text{ or } \left(\frac{\mu_j}{\gamma_j} < \theta_{ji} \right) \text{ if } v_j = -1.$$

(c) *Possible transitions from D_i to D_k .* If the calculated inequality set defines a not empty parameter space domain

$PSD_k^i \subseteq PSD_0$ then a transition towards D_k is possible and the qualitative state $QS(D_i)$ is updated accordingly.

To exemplify how the algorithm works we consider the ODE model:

$$\begin{aligned} \dot{x}_1 &= \kappa_{11}(1 - Z_{11})(1 - Z_{22}) + \kappa_{12}(1 - Z_{21}) - \gamma_1 x_1, \\ \dot{x}_2 &= \kappa_{21}(1 - Z_{12}) - \gamma_2 x_2, \end{aligned} \quad (7)$$

where the parameters are positive, and all the response functions are expressed by the Hill function commonly used in the literature.

Let us consider the domain D_{11} in Fig. 2 as current D_i , and let us define I_0 as follows:

$$I_0 : \left(\frac{\kappa_{11} + \kappa_{12}}{\gamma_1} > \theta_{11} \right) \wedge \left(\theta_{21} < \frac{\kappa_{21}}{\gamma_2} < \theta_{22} \right). \quad (8)$$

The conditions on parameters in Table 1 that are consistent with I_0 identify the possible transitions from D_{11} towards adjacent domains in $A(D_{11}) = \{D_6, D_7, D_{12}, D_{16}, D_{17}\}$. Among the inequalities in Table 1, I_{12}^{11} is the only one in agreement with I_0 . Thus, $QS(D_{11}) = \{D_{12}\}$ and the only possible transition from D_{11} occurs when $I_{12}^{11} \wedge I_0$ holds.

Moving from a switching domain

In a domain $D_i \in \Delta_s$, the nonlinear dynamics is characterized by fast and slow motions, respectively associated with x_s and x_r , that are independently calculated. Let us reindex the variables x_r, Z_j such that the switching variables come first, and proceed first with the construction of the fast motion by exploiting the SPA approach.

A – Fast motion

The study of the fast dynamics is performed in $Z(D_i)$ in the scaled time $\tau = t/q$, and aims at localizing the set of exit points in $Z(D_i)$ rather than at detailing the dynamics within it. Such points clearly identify the next adjacent domains the trajectories are moving towards from D_i along the x_s directions.

Table 1: Example parameter inequalities

Inequality Constraints		
I_k^i	Sign Constraints	Parameter Inequalities
I_6^{11}	$\dot{x}_2 < 0$	$\left(\frac{\kappa_{21}}{\gamma_2} < \theta_{21} \right)$
I_{12}^{11}	$\dot{x}_1 > 0$	$\left(\frac{\kappa_{11}}{\gamma_1} > \theta_{11} \right)$
I_7^{11}	$\dot{x}_1 > 0, \dot{x}_2 < 0$	$I_6^{11} \wedge I_{12}^{11}$
I_{16}^{11}	$\dot{x}_2 > 0$	$\left(\frac{\kappa_{21}}{\gamma_2} > \theta_{22} \right)$
I_{17}^{11}	$\dot{x}_1 > 0, \dot{x}_2 > 0$	$I_{12}^{11} \wedge I_{16}^{11}$

To this end, the algorithm defines, in the limit, the mapping $\Sigma_{D_i} : D_i \rightarrow Z(D_i)$, where $D_i = D_i \cup A(D_i)$, such that the interior of $Z(D_i)$, and its boundary are the images of D_i , and $A(D_i)$, respectively. More precisely, the domains $D_k \in A(D_i)$ are mapped into the faces of $Z(D_i)$ when $D_k \in \Delta_s$ or into its vertices, otherwise. Let \mathcal{F} be the set of both the faces and the interior of $Z(D_i)$, and F its generic element.

To exemplify, let us consider the switching domain D_7 in Fig. 2. The set \mathcal{F} has five elements, the four faces corresponding to D_2, D_6, D_8, D_{12} , and the interior of $Z(D_7)$ that correspond to D_7 . Moreover, the vertices of $Z(D_7)$ are the images, through the mapping Σ , of the adjacent regular domains D_1, D_{11}, D_3 , and D_{13} (Fig. 3).

(a) 1. Symbolic state equations in $D_i \in \Delta_s$

The algorithm symbolically calculates the boundary-layer equations (4) in the Z variables associated with the current switching domain.

As an example, let us consider the model (7), and the domain D_7 , where both variables x_1, x_2 are switching in it, respectively, around the thresholds θ_{11}, θ_{21} . In D_7 , characterized by fast motion only, the boundary-layer system is given by:

$$\begin{aligned} Z'_{11} &= \frac{Z_{11}(1-Z_{11})}{\theta_{11}} (\kappa_{11}(1 - Z_{11}) + \kappa_{12}(1 - Z_{21}) - \gamma_1 \theta_{11}), \\ Z'_{21} &= \frac{Z_{21}(1-Z_{21})}{\theta_{21}} (\kappa_{21} - \gamma_2 \theta_{21}). \end{aligned} \quad (9)$$

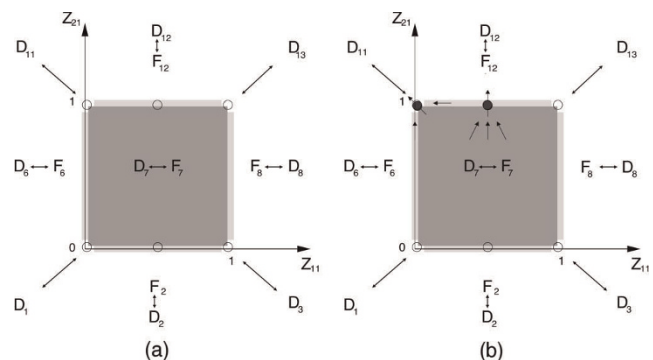


Figure 3 Mapping between D_i and $Z(D_i)$. (a) Mapping of D_7 and its adjacent domains in Fig. 2 into the elements of $Z(D_7)$. The empty circles on the boundary of $Z(D_7)$ denote the candidate exit points. (b) The filled circles denote the exit points from $Z(D_7)$, whereas the empty circles denote unstable stationary points that correspond to possible entrance points to $Z(D_7)$. The single-headed arrows denote the direction of change of $Z_r, \forall l \in \mathcal{L}_F$ in a neighborhood of the exit points $\bar{Z}_{11}, \bar{Z}_{12}$.

2. Identification of the candidate next traversed domains D_k

Among all the domains $D_k \in A(D_i)$ only those that correspond to an element on the boundary of $Z(D_i)$ where a stationary point is located are possible domains next traversed by the trajectories. Then, the algorithm first builds the subset of elements of $A(D_i)$ that are candidate successors of D_i .

Let us denote by EP the set of stationary points, initially made up of the vertices of $Z(D_i)$. Under the Assumption A, each element $F \in \mathcal{F}$, may contain at most one stationary point. A necessary condition for the existence of a stationary point on F is the presence of a non-zero loop in J_F , the Jacobian matrix calculated on F . Then, $\forall F \in \mathcal{F}$, the algorithm calculates J_F and searches for a non-zero loop involving all variables in J_F . In case, it symbolically calculates the stationary point on F , and updates accordingly the set of candidate exit points EP . Let us observe that one internal stationary point exists if all variables in $Z(D_i)$ are involved in a loop.

In the example, only J_{F_2} and $J_{F_{12}}$ have a non-zero loop. Then, the algorithm looks for the stationary state on F_2 and F_{12} : $\tilde{Z}^2 = (1 + \frac{\kappa_{12}}{\kappa_{11}} - \frac{\gamma_1 \theta_{11}}{\kappa_{11}}, 0)$ and $\tilde{Z}^{12} = (1 - \frac{\gamma_1 \theta_{11}}{\kappa_{11}}, 1)$. Finally, the exit point candidate set is updated with the points \tilde{Z}^2 and \tilde{Z}^{12} (Fig. 3(a)).

(b) Constraints I_k^i on parameters

The inequality set I_k^i , initialized to I_0 , is calculated for each candidate exit point $\tilde{Z}^k = \{\tilde{Z}_s^k\} \in EP$ by requiring that each point is stable and fulfills other motion conditions on parameters. More precisely, the algorithm:

1. analyzes the spectrum of the Jacobian matrix J_{F_i} and imposes possible conditions $I_{k,s}^i$ on parameters to ensure stability of \tilde{Z}^k ;
2. imposes conditions $I_{k,l}$ on the sign of $Z_l^i, \forall l \in \mathcal{L}_{F_i}$, in a neighborhood of \tilde{Z}^k so that the motion of Z_l heads towards \tilde{Z}^k ;
3. for those \tilde{Z}^k located on elements of \mathcal{F} , imposes the further condition, $I_{k,(0,1)}, 0 < \tilde{Z}_s^k < 1$ for each \tilde{Z}_s^k that does not take value 0 or 1.

Finally, \tilde{Z}^k is an exit point if $I_k^i = I_0 \wedge I_{k,s}^i \wedge I_{k,l} \wedge I_{k,(0,1)}$ holds.

Conditions 2. and 3. are easily checked, while the stability condition is checked by using concepts from graph theory, and the usual definition of stability based on the sign of the eigenvalues of J_F . It follows from

Assumption A that the matrix J_F is block-structured, where each block is a permutation matrix associated with a sub-loop. It can be proved that \tilde{Z}^k is stable if: (i) J_{F_k} has no blocks with dimension strictly greater than 2; (ii) in 1-dimensional blocks the elements are negative; (iii) in 2-dimensional blocks the loop products are positive.

Remark 4

Let us consider the general case when $D_i \in \Delta_s$ is characterized by both switching and regular variables. The motion from an exit point located in the interior of D_i occurs in a sliding mode along a stable point in the slow-manifold of the boundary-layer system, and is described, in the normal time, by the reduced system. Then, a stable stationary point exists in D_i , i.e. $D_i \in QS(D_i)$, if a stable point exists in the interior of $Z(D_i)$, and the regular variable rates are zero in a point inside D_i . The set of inequalities that checks the latter condition are defined by $\theta_{ji} < \frac{\mu_j}{\gamma_j} < \theta_{j(i+1)} \forall j \in \{\sigma(D_i) + 1, \dots, n\}$.

(c) Possible transition from D_i to D_k

The exit points located on $Z(D_i)$ clearly identify the set of all possible exit domains, i.e. those domains towards which a transition from D_i is possible. Such domains are easily calculated by applying the map Σ^{-1} to each element of $Z(D_i)$ that contains an exit point. Let us observe that the remaining unstable stationary points in EP are possible entrance points to D_i .

In the example, both J_{F_2} and $J_{F_{12}}$ are 1-dimensional block matrices with negative elements. Then, both exit points \tilde{Z}^2 and \tilde{Z}^{12} fulfill the stability condition 1. without further constraints imposed on the parameters. The condition 2. on variable $Z_l, l = 2$ imposes:

$$I_{2,2} : f_2(\tilde{Z}^2) < 0 \Rightarrow \kappa_{21} - \gamma_2 \theta_{21} < 0, \quad (10)$$

$$I_{12,2} : f_2(\tilde{Z}^{12}) > 0 \Rightarrow \kappa_{21} - \gamma_2 \theta_{21} > 0. \quad (11)$$

The inequality $I_{12,2}$ defined by (11) is compatible with (8), but the inequality $I_{2,2}$ is not. Then, \tilde{Z}^2 is removed from the exit point set. To be an exit point \tilde{Z}^{12} must satisfy the condition 3.:

$$I_{12,(0,1)} : 0 < \tilde{Z}_1^{12} < 1 \Rightarrow (\kappa_{11} - \gamma_1 \theta_{11}).$$

Finally, \tilde{Z}^{12} is an exit point if I_{12}^7 , defined by $I_0 \wedge I_{12,2} \wedge I_{12,(0,1)}$, holds.

As for vertices in the example, the conditions 1. and 2. are fulfilled in the point $\tilde{Z}^{11} = (0, 1)$ that corresponds to the vertex defined as image of D_{11} by the map Σ .

Finally, the only exit domains are D_{12} , D_{11} , and then $QS(D_7) = \{D_{12}, D_{11}\}$ (Fig. 3(b)).

B – Slow motion

The slow motion of regular variables x_r along the exit points is studied in the normal time t in the usual frame of reference, and it is reconstructed from the reduced system (5) through the same symbolic procedure given for regular domains.

Qualitative simulation outcomes

Through the described iterative procedure, the algorithm builds (i) a behavior tree, rooted in the initial domain, and (ii) sets of inequalities on parameter values that characterize specific state transitions. The behavior tree captures the entire spectrum of network dynamical behaviors that are invariant for ranges of parameter values defined by the calculated inequalities.

To illustrate the type of output the algorithm produces, let us consider the results of the simulation of the ODE model (7) starting from D_1 with the parameter space defined by the inequalities I_0 (8). Through the described iterative procedure, the algorithm builds the behavior tree, $BT(D_1)$, showed in Fig. 4, and calculates the inequalities on the parameters, listed in Table 2, that are associated with each path in $BT(D_1)$. As in the example $n = 2$, the trajectories described by the tree are also easily summarized in the phase plane (Fig. 5). Three reachable stable states, located in D_{11} , D_{12} and D_5 , are identified by the final leaf of each branch in BT. As $D_{12} \in \Delta_s$, one of them is a singular stable point whereas the others are regular stable points. These stable states are reached by different predicted qualitative behaviors, each of them occurring under specific constraints on

Table 2: Inequalities associated with $BT(D_1)$

I_{12}^7, I_{12}	$I_0 \wedge \left(\frac{\kappa_{11}}{\gamma_1} > \theta_{11} \right)$
I_3^2	$I_0 \wedge \left(\frac{\kappa_{12}}{\gamma_1} > \theta_{11} \right)$
$I_5^9, I_9^3, I_4^3, I_5^4$	$I_0 \wedge \left(\frac{\kappa_{12}}{\gamma_1} > \theta_{12} \right)$
$I_{11}, I_{11}^7, I_{11}^{12}$	$I_0 \wedge \left(\frac{\kappa_{11}}{\gamma_1} < \theta_{11} \right)$
I_7^2	$I_0 \wedge \left(\frac{\kappa_{12}}{\gamma_1} < \theta_{11} \right)$
I_5	$I_0 \wedge \left(\theta_{12} < \frac{\kappa_{12}}{\gamma_1} < \bar{x}_1 \right)$
$I_2^1, I_{16}^1, I_7^1, I_8^3$	I_0
$I_{11}^6, I_{18}^8, I_{12}^{13}, I_{13}^9$	I_0

parameters. For example, the trajectory QB_{13} starting from D_1 , crossing D_6 , and reaching a stable point in D_{11} is allowed when the inequalities I_6^1, I_{11}^6 and I_{11} , consistent with I_0 , hold.

In general, for $n > 2$, due to graphical difficulties, each branch in the tree is given a time representation. More precisely, a specific qualitative behavior, e.g. QB_4 , is described by the temporal evolution of each variable (Fig. 6). The behavior in Fig. 6 abstracts all numerical trajectories of the ODE model (7) obtained with different initial values in D_1 , and different sets of parameter values that fulfill the same inequalities associated with QB_4 (Fig. 7).

Discussion

The algorithm is currently under implementation. As for symbolic calculus, the implementation requires to tackle complex tasks, such as: (i) update an inequality set with

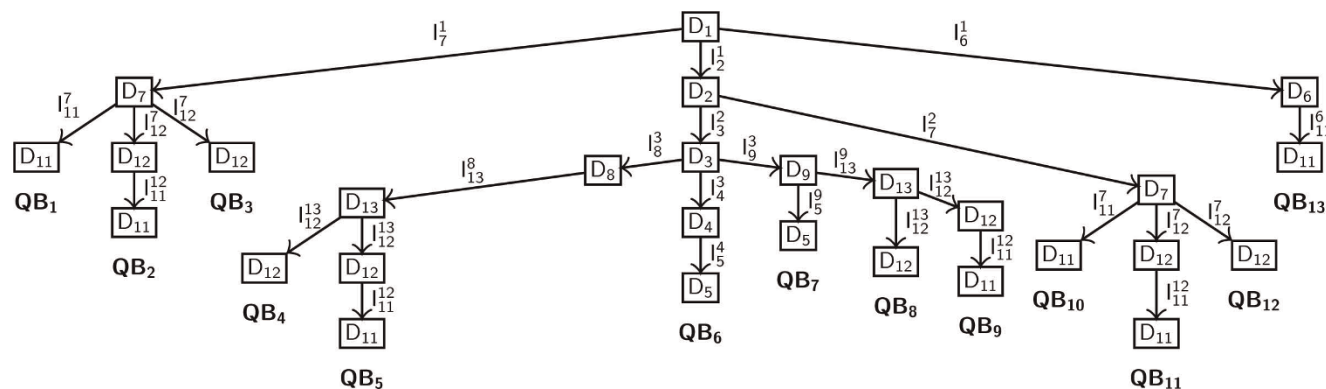


Figure 4 Behavior tree. Behavior tree rooted in D_1 , $BT(D_1)$, obtained by the simulation of the model Eqs.(7) with initial condition $\langle D_1, I_0 \rangle$, where I_0 is defined by Eq. (8). Each branch in the tree defines a qualitative trajectory that occurs when the inequalities that characterize each path in the branch hold.

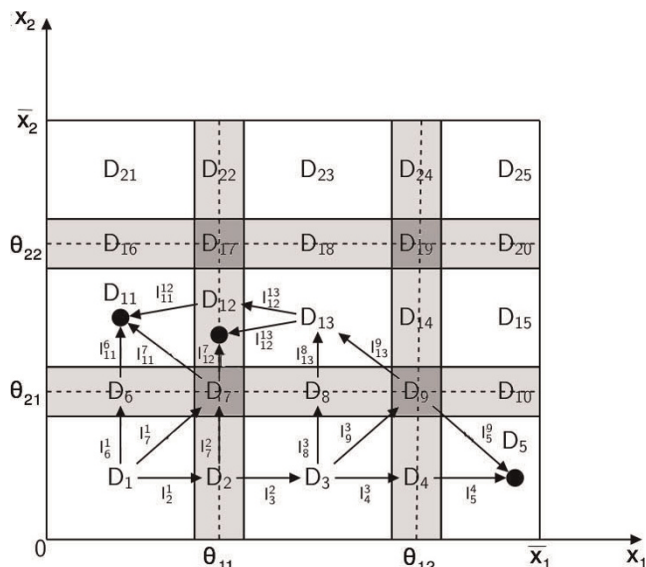


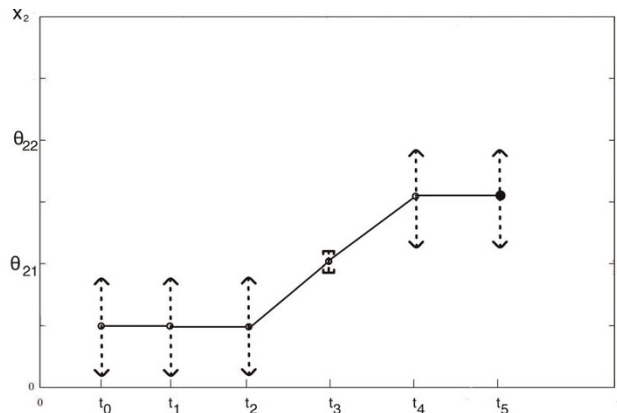
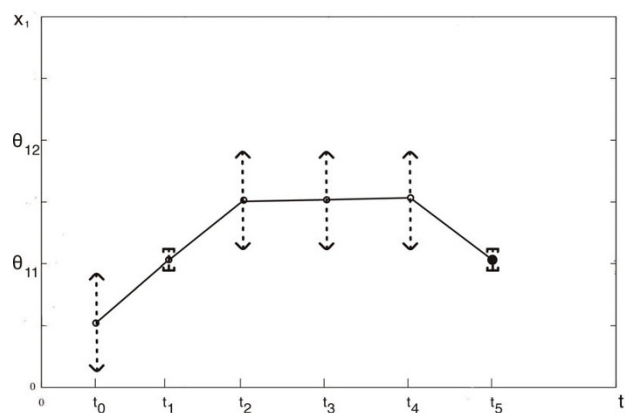
Figure 5
Representation of the qualitative trajectories in the phase space. Phase space representation of trajectories defined by $BT(D_1)$ in Fig. 4, filtered of the spurious behaviors QB_2 and QB_{11} . Stable states are denoted by \cdot . The labels I_k^i denote the sets of inequalities on parameter values that hold when transitions from D_i to D_k occur.

an another one; (ii) check the consistency of two sets of inequalities I_1 and I_2 ; (iii) solve systems of equations; (iv) find cycles in the Jacobian matrix. As for (iii), the original equations are multilinear in Z_s , but due to *Assumption A* they can be straightforward solved and analyzed for stability. Also the solution of problems (i) and (ii) benefits from *Assumption A* as the inequalities are always linear. Then, thanks to the *Assumption A*, and to algorithms proposed both by the literature and common symbolic computation package, such as Mathematica [34], the tasks (i)-(iii) are simplified and feasible. As for the task (iv), it is performed by using cycle-detection algorithms and matrix graph theory tools [35].

Soundness and completeness

Soundness

The algorithm guarantees that the behavior tree captures all of the sound behaviors for values of q sufficiently small. A closed-form expression of a symbolic upper bound of q , \bar{q} , in terms of model parameters guarantees the Jacobian matrix stability. Then, the generated behaviors are sound for values of $q < \bar{q}$. However, for a complete formal proof of soundness we need to prove that assuming, in the limit, stability instead of asymptotic stability does not affect the results for values of $q < \bar{q}$.



(a)

(b)

Figure 6
Temporal representation of a qualitative trajectory. Qualitative temporal representation of the behavior labeled QB_4 in Fig. 4, namely of the variables x_1 (a) and x_2 (b). On the horizontal axis, the discrete sequence of symbolic instants at which the domains ($D_1, D_2^1, D_3^2, D_3^3, D_8^8, D_{12}^{13}$) are sequentially traversed. On the vertical axis, the qualitative value space of each variable is reported. The qualitative values (\circ) abstract all real values in the related open and closed intervals, whose width is highlighted by a dashed line. The stable states are denoted by \cdot . The continuous line connecting the qualitative values has been introduced to give a hint of the qualitative profile.

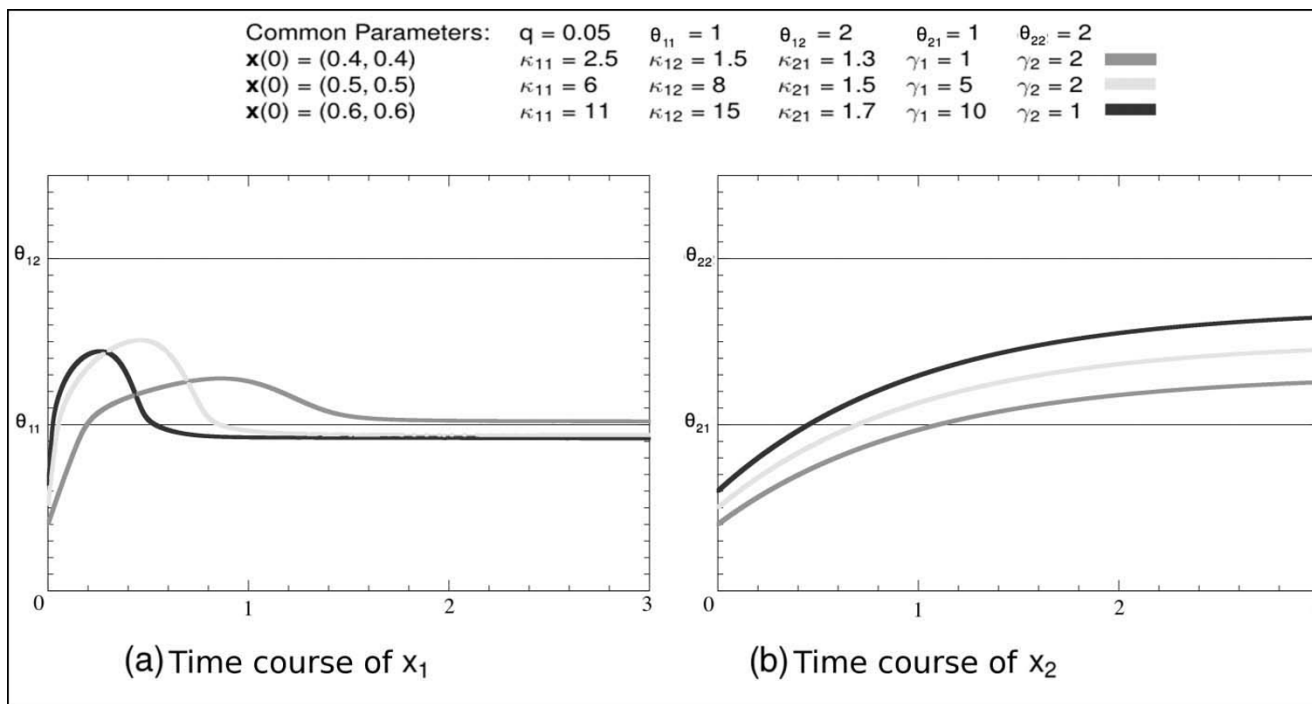


Figure 7
Numerical simulation results. Time courses of the variables x_1 (a) and x_2 (b) obtained by numerical simulations of the model (7). The different plots result from simulations with different initial numeric values in D_1 , and different sets of parameter numeric values but fulfilling the same inequalities that characterize the qualitative trajectory QB_4 . The horizontal continuous lines in correspondence to threshold values should facilitate the comparison with Fig. 6. All plots related to each variable are qualitatively abstracted by the related plot in Fig. 6.

Completeness

At the current stage, the algorithm may generate spurious behaviors. There is a twofold explanation for that. First, we have not yet performed a thorough analysis with respect to entrance-exit transition, or in other words, we have not yet solved the problem of identifying the only admissible connections between entrance and exit points. Moreover, singular perturbation analysis is a *local* procedure that works quite well in a quantitative context but that needs, in a qualitative context, to be supported by a *global* criterion when local paths are connected to produce a specific trajectory. For example, the behavior QB_2 is spurious as I_{12}^7 is not consistent with I_{11}^{12} . Similarly, QB_{11} is spurious.

The characterization of the paths from one domain to the next ones by sets of inequalities constraining the model parameters is quite new in the field of qualitative simulation, as for both general-purpose and specifically tailored algorithms. Such a strategy may reveal quite useful in the definition of a “global criterion” that allows us to distinguish sound behaviors from spurious ones by requiring that the sets of inequalities that label the local paths in a specific trajectory are consistent with each other. Both the definition of such a criterion and its

implementation are not a trivial task, especially from a computational point of view. As for algorithm completeness, another essential methodological and computational issue to be deepened deals with the definition of the transition map that *properly* connects the entrance points to the exit points associated with a switching domain.

Comparison between the algorithm and GNA

Until now, to the best of our knowledge, GNA is the only computational counterpart of the algorithm herein proposed. Unlike GNA , that simplifies the simulation problem by approximating the sigmoid response functions by step functions *discontinuous* in the thresholds, our qualitative simulation algorithm tackles the problem in all of its complexity, and works for models of GNA s with *continuous* sigmoid response functions. As a matter of fact, it has been designed to both provide a tool for a more realistic modeling framework and overcome the limits of GNA . Furthermore, in addition to more solid mathematical grounds for soundness and completeness, our algorithm differs from GNA as far as the required input information and the output outcomes are concerned. In GNA , the trajectories are constructed under the assumption that the locations of

the focal points associated with the regular domains, symbolically expressed by a set of inequalities on parameters, are specified as input. But, as this precise knowledge is unlikely available, a thorough study of the dynamics of the system at hand might require several runs of the algorithm to simulate the GNA dynamics under possible different assumptions on focal point locations. Then, the study could result quite heavy due to the high number of possible scenarios, and the need to check, to avoid further causes of generation of spurious behaviors, the consistency of all the parameter inequalities that precisely localize a specific focal point.

Our algorithm does not need precise information on the location of focal points but, to be fired, it just requires the parameter space domain, that is inequalities on parameters that univocally determine the sign of direction of change of each state variable in the initial domain. Afterwards, it is up to the algorithm to calculate and refine the parameter inequalities that distinguish each simulated behavior. Due to more relaxed constraints, a single run of our algorithm leads to a richer and more informative set of behaviors than that one obtained by GNA as it is highlighted by the comparison of Fig. 5 with Fig. 8. Let us observe that any other possible

definition of parameter inequalities would not have led to the set of behaviors in Fig. 5 but to one of its subsets.

A complete and fair performance evaluation of our approach when compared with GNA would not disregard its application to those real world networks successfully simulated within the GNA framework, namely the initiation of sporulation in *Bacillus subtilis* [10], and the response to nutritional stress and carbon starvation in *Escherichia coli* [11,12].

Application to Systems/Synthetic Biology

The qualitative simulation algorithm presented provides powerful computational grounds for *dry* experiments to be performed in both Systems and Synthetic Biology contexts. It is an economic tool for hypothesis testing and knowledge discovery as it allows us to understand how specific activation patterns are derived from models of fixed network structures, and which different dynamical behaviors are possible.

In a Systems Biology context, its exploitation is useful in the formulation phase of new hypotheses and theories that explain, in both physiological and pathological

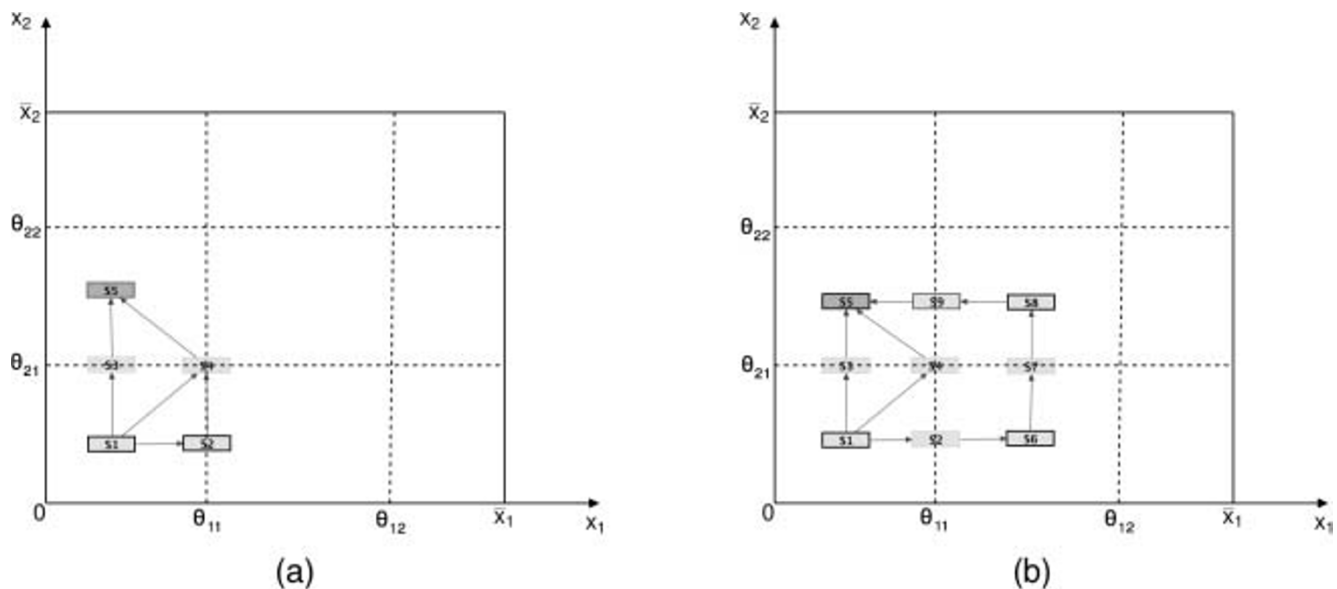


Figure 8
GNA simulation outcomes. Simulation outcomes of the example model obtained by the algorithm GNA given as input D_1 and the parameter inequalities, compatible with l_0 , $0 < \frac{\kappa_{11}}{\gamma_1} < \theta_{11}$; $0 < \frac{\kappa_{12}}{\gamma_1} < \theta_{11}$; $\theta_{11} < \frac{\kappa_{12} + \kappa_{11}}{\gamma_1} < \theta_{12}$; $\theta_{21} < \frac{\kappa_{21}}{\gamma_2} < \theta_{22}$ (panel (a)), $0 < \frac{\kappa_{11}}{\gamma_1} < \theta_{11}$; $\theta_{11} < \frac{\kappa_{12}}{\gamma_1} < \theta_{12}$; $\theta_{12} < \frac{\kappa_{12} + \kappa_{11}}{\gamma_1} < \bar{x}_1$; $\theta_{21} < \frac{\kappa_{21}}{\gamma_2} < \theta_{22}$ (panel (b)). The states generated by GNA , and labeled S_n , have been located at their position in the phase space to make the comparison with the results in Fig. 5 possible. Both sets of trajectories in (a) and (b) are subsets of the set of trajectories generated by our algorithm.

situations, experimental data either previously unobserved or not interpretable by the well-established biological knowledge. The result of the comparison of the simulated behaviors with the observed ones allows us either to confirm or to refuse the model that represents the new formulated biological hypotheses. Moreover, as qualitative simulation derives the entire range of network dynamics consistent with the initial conditions, the simulation outcomes might capture behaviors never observed. Such behaviors need to be experimentally tested by *ad hoc* designed experiments to possibly further confirm or suggest how to revise the model, i.e. the underlying biological hypotheses. The effective applicative potential of the proposed algorithm within a model-based reasoning cycle for the deep comprehension of complex real world biological systems is currently under study. More precisely, it will be applied (1) to explain and interpret the dynamics of the complex regulatory network that controls motility in *Bacillus subtilis* [36], and (2) to explore the molecular mechanisms that regulate the transition of a normal cell phenotype to an invasive-metastatic phenotype [37].

The model-based reasoning cycle results particularly convenient and useful also when applied to Synthetic Biology problems in the design phase. The construction of a synthetic biological network in the cell, able to perform specific tasks or to produce desired behaviors of a biological process in response to external stimuli, could benefit from a preliminary *dry* design of the network. To this end, models of different network structures can be simulated in correspondence to different inputs, i.e signals either exogenous or cellular, till the desired stable behaviors are obtained. Let us remind that, in our simulation framework, each specific simulated behavior is characterized by a chain of parameter inequalities. Thus, the actual construction of a synthetic network could benefit from such a piece of information as it provides the admissible parameter space domain for the target behaviors. The synthetic network IRMA[38] built in the yeast *Saccharomyces cerevisiae* is made up of a small number of components but rather complex in their interconnections as it includes multiple feedback loops generated by the combination of transcriptional activators and repressors. It offers a suitable benchmark for *in vivo* testing our computational framework.

Conclusion

The assumption of continuous response functions makes the simulation problem hard to be tackled but it is crucial in view of the realization of tools that can be gradually extended to tackle more and more realistic models. Our algorithm is grounded on a set of symbolic

computation algorithms that carry out the integration of qualitative reasoning techniques with singular analysis perturbation methods: the former techniques allow us to cope with uncertain and incomplete knowledge whereas the latter ones lay the mathematical groundwork for a sound and complete algorithm capable to deal with regulation processes that occur at different time-scales.

The modeling framework and the simulation algorithm proposed can be applied to predict the possible qualitative behaviors of GRNs of any size and complexity, both natural and synthetic. The range of applicability of such a computational approach is quite large. First, it makes possible the validation of hypothesized models of GRNs by matching the simulated predictions against observed gene expression profiles, the derivation of the most plausible model, and the identification of parameter inequalities that account for the observations. Then, it greatly facilitates the investigation of the effects on the dynamics of a specific GRN in response to external stimuli. Finally, it allows us to build a total envisionment of the model through the generation of all possible state transitions, and to search, within them, for the parameter constraints and initial conditions that allow us to achieve a desired behavior, e.g. a stable solution. In term of tasks, besides its undoubted contribution to the comprehension of complex networks of genes and interactions, the proposed computational approach could be fruitfully used in the design of either new drugs or synthetic regulatory networks.

The code will be freely available for non-profit academic research by making a user licence request to ironi@imati.cnr.it.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

LI devised the work, and drafted the manuscript. Both authors participated in the design and development of the algorithms. LP implemented the software. All authors read and approved the final manuscript.

Acknowledgements

This research is carried out within the Interdepartmental CNR-BIOINFORMATICS Project. We gratefully acknowledge O. Dordan, E. Plahte, and V. Simoncini for the useful discussions on different methodological aspects and mathematical problems related to the presented work.

This article has been published as part of *BMC Bioinformatics* Volume 10 Supplement 12, 2009: Bioinformatics Methods for Biomedical Complex System Applications. The full contents of the supplement are available online at <http://www.biomedcentral.com/1471-2105/10?issue=S12>.

References

1. Glass L and Kauffman SA: **The logical analysis of continuous, nonlinear biochemical control networks.** *Journal of Theoretical Biology* 1973, **39**:103–129.
2. Plahte E, Mestl T and Omholt SW: **A methodological basis for description and analysis of systems with complex switch-like interactions.** *Journal of Mathematical Biology* 1998, **36(4)**:321–348.
3. de Jong H: **Modeling and Simulation of Genetic Regulatory Systems: A Literature Review.** *Journal of Computational Biology* 2002, **9**:67–103.
4. Kuipers B: *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge* Cambridge, MA: MIT Press; 1994.
5. Glass L: **Global analysis of nonlinear chemical kinetics.** *Statistical Mechanics, Part B: Time Dependent Processes* Plenum Press, New York; Berne B 1977, 311–349.
6. Hastay J, McMillen D, Isaacs F and Collins JJ: **Computational studies of gene regulatory networks: In numero molecular biology.** *Nat Rev Genet* 2001, **2**:268–279.
7. Gouzé JL and Sari T: **A class of piecewise linear differential equations arising in biological models.** *Dynamical systems* 2003, **17**:299–316.
8. Plahte E and Kjøglum S: **Analysis and generic properties of gene regulatory networks with graded response functions.** *Physica D: Nonlinear Phenomena* 2005, **201(1–2)**:150–176.
9. de Jong H, Gouzé JL, Hernandez C, Page M, Sari T and Geiselmann J: **Qualitative Simulations of Genetic Regulatory Networks Using Piecewise Linear Models.** *Bulletin of Mathematical Biology* 2004, **66(2)**:301–340.
10. de Jong H, Geiselmann J, Batt G, Hernandez C and Page M: **Qualitative simulation of the initiation of sporulation in *Bacillus subtilis*.** *Bulletin of Mathematical Biology* 2004, **66(2)**:261–300.
11. Ropers D, de Jong H and Geiselmann J: **Mathematical modeling of genetic regulatory networks: Stress response in *Escherichia coli*.** *Systems and Synthetic Biology* Wiley & Sons: Fu P, Latterich M, Panke S in press.
12. Ropers D, de Jong H, Page M, Schneider D and Geiselmann J: **Qualitative simulation of the carbon starvation response in *Escherichia coli*.** *BioSystems* 2006, **84(2)**:124–152.
13. Bacciotti A: **Some remarks on generalized solutions of discontinuous differential equations.** *Int Journal of Pure and Applied Mathematics* 2003, **10(3)**:257–266.
14. Filippov AF: *Differential equations with discontinuous right hand sides* Dordrecht: Kluwer Academic Publishers Group; 1988.
15. Veflingstad SR and Plahte E: **Analysis of gene regulatory network models with graded and binary transcriptional responses.** *Biosystems* 2007, **90**:323–339.
16. Goldbeter A and Koshland DE: **An Amplified Sensitivity Arising from Covalent Modification in Biological Systems.** *PNAS* 1981, **78(11)**:6840–6844.
17. Yuh CH, Bolouri H and Davidson EH: **Genomic cis-Regulatory Logic: Experimental and Computational Analysis of a Sea Urchin Gene.** *Science* 1998, **279(5358)**:1896–1902.
18. Gardner T, Cantor C and Collins J: **Construction of a genetic toggle switch in *Escherichia coli*.** *Nature* 2000, **403(6767)**:339–342.
19. Rossi FMV, Kringstein AM, Spicher A, Guicherit OM and Blau HM: **Transcriptional control: Rheostat converted to on/off switch.** *Molecular Cell* 2000, **6(3)**:723–728.
20. Biggar SR and Crabtree GR: **Cell signaling can direct either binary or graded transcriptional responses.** *The EMBO Journal* 2001, **20**:3167–3176.
21. Yuh CH, Bolouri H and Davidson EH: **cis-regulatory logic in the endo16 gene: switching from a specification to a differentiation mode of control.** *Development* 2001, **128(5)**:617–629.
22. Guet CC, Elowitz MB, Hsing WH and Leibler S: **Combinatorial synthesis of genetic networks.** *Science* 2002, **296(5572)**:1466–1470.
23. Setty Y, Mayo AE, Surette MG and Alon U: **Detailed map of a cis-regulatory input function.** *PNAS* 2003, **100(13)**:7702–7707.
24. Kramer BP, Fischer C and Fussenegger M: **BioLogic gates enable logical transcription control in mammalian cells.** *Biotechnol Bioeng* 2004, **87(4)**:478–484.
25. Istrail S and Davidson EH: **Logic functions of the genomic cis-regulatory code.** *PNAS* 2005, **102(14)**:4954–4959.
26. Rosenfeld N, Young JW, Alon U, Swain PS and Elowitz MB: **Gene Regulation at the Single-Cell Level.** *Science* 2005, **307(5717)**:1962–1965.
27. Kim J, Bates DG, Postlethwaite I, Ma L and Iglesias PA: **Robustness analysis of biochemical network models.** *Syst Biol (Stevange)* 2006, **153(3)**:96–104.
28. Mayo AE, Setty Y, Shavit S, Zaslaver A and Alon U: **Plasticity of the cis-Regulatory Input Function of a Gene.** *PLoS Biology* 2006, **4(4)**:e45.
29. Wolf DM and Eeckman FH: **On the Relationship Between Genomic Regulatory Element Organization and Gene Regulatory Dynamics.** *Journal of Theoretical Biology* 1998, **195(2)**:167–186.
30. Buchler NE, Gerland U and Hwa T: **Nonlinear protein degradation and the function of genetic circuits.** *PNAS* 2005, **102(27)**:9559–9564.
31. Bintu L, Buchler NE, Garcia HG, Gerland U, Hwa T, Kondev J and Phillips R: **Transcriptional regulation by the numbers: models.** *Current Opinion in Genetics & Development* 2005, **15(2)**:116–124.
32. Bintu L, Buchler NE, Garcia HG, Gerland U, Hwa T, Kondev J, Kuhlman T and Phillips R: **Transcriptional regulation by the numbers: applications.** *Current Opinion in Genetics & Development* 2005, **15(2)**:125–135.
33. Holmes M: *Introduction to Perturbation Methods* Berlin: Springer; 1995.
34. Wolfram S: *The Mathematica Book* Wolfram Media; 2003.
35. Gross J and Yellen J: *Graph Theory and its Applications* New York: Chapman & Hall/CRC Press; 2006.
36. Calvio C, Osera C, Amati G and Galizzi A: **Autoregulation of swrAA and motility in *Bacillus subtilis*.** *J Bacteriol* 2008, **190**:5720–5728.
37. Gentile A, D'Alessandro L, Lazzari L, Martinoglio B, Bertotti A, Mira A, Lanzetti L, Comoglio PM and Medico E: **Met-driven invasive growth involves transcriptional regulation of Arhgap12.** *Oncogene* 2008, **27**:5590–5598.
38. Cantone I, Marucci L, Iorio F, Ricci M, Belcastro V, Bansal M, Santini S, di Bernardo M, di Bernardo D and Cosma M: **A Yeast Synthetic Network for In Vivo Assessment of Reverse-Engineering and Modeling Approaches.** *Cell* 2009, **137**:172–181.

Publish with **BioMed Central** and every scientist can read your work free of charge

"BioMed Central will be the most significant development for disseminating the results of biomedical research in our lifetime."

Sir Paul Nurse, Cancer Research UK

Your research papers will be:

- available free of charge to the entire biomedical community
- peer reviewed and published immediately upon acceptance
- cited in PubMed and archived on PubMed Central
- yours — you keep the copyright

Submit your manuscript here:
http://www.biomedcentral.com/info/publishing_adv.asp

