Qualitative simulation of nonlinear dynamical models of gene-regulatory networks

Liliana Ironi and **Luigi Panzeri**
IMATI-CNR
via Ferrata 1, Pavia (Italy)

Abstract

This paper discusses the work-in-progress of a research effort aiming at the design and implementation of a qualitative simulation algorithm of the dynamics of a specific class of ODE models of Gene-Regulatory Networks (GRN). In such models, characterized by incomplete knowledge of regulation mechanisms and kinetic parameters, regulation is assumed to be threshold-dependent, i.e. only effective above or below a certain threshold. Switch-like behaviors across variable thresholds are properly modeled by steep sigmoid functions the values of which continuously vary from zero to one around the threshold. The ODE models that result from the algebraic combination of such switch-like interaction terms describe both linear and nonlinear GRN dynamics that occur at different time-scales. Qualitative simulation of such kinds of models is a quite hard problem that requires the development of *ad hoc* tailored algorithms. Unlike GNA, that considerably simplifies the problem by approximating threshold-regulated response functions by step functions, we propose a qualitative simulation algorithm that works for continuous models, being the continuity assumption crucial in view of more and more realistic models. The algorithm is grounded on the integration of QR techniques with singular perturbation analysis methods that lay the mathematical basis for dealing with both slow and fast nonlinear dynamics.

Introduction

Due to unprecedented amount of information at genomic level made available, in recent years, by high-throughput experimental technologies, it has become increasingly clear that computational modeling and simulation frameworks are needed to represent, understand and predict the complex dynamics of Gene-Regulatory Networks (GRN). Although, up to now, there is no model that efficiently and accurately represents the gene interactions underlying regulatory mechanisms in their whole complexity, a specific class of ODEs has shown to be adequate to describe the essential features of their dynamics. These models assume that the interactions between variables are threshold-dependent, i.e. the effect of a variable on another one is regulated by a threshold value. Such an assumption is quite reasonable as switch-like behaviors across variable thresholds are well-suited to mathematically represent the effects of the transcription factors on the transcription rates of genes. Although such models allow us to provide detailed description of gene regulatory mechanisms at the molecular level (Glass & Kauffman 1973; Plahte, Mestl, & Omholt 1998), their applicability to predict their quantitative dynamics is rather limited even when the network at hand is very well studied. As a matter of fact, making predictions of the dynamics of specific networks, either from an initial state or in response to environmental stimuli, by exploiting 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 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.

To this end, the application of generic qualitative simulation approaches (Kuipers 1994), at least in their original form, is not the right solution. The mathematical tools they are grounded on are too much simple to compensate for the lack of complete knowledge. 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 reveal to be particularly serious in predicting nonlinear dynamics of regulatory networks even in the case of networks with a small number of interacting genes. A qualitative study of GRNs dynamics could, in theory, be performed by more sophisticated analytical methods based on the classical theory of qualitative analysis of dynamical systems, and properly adapted to the specific class of models (Glass & Kauffman 1973; de Jong 2002; Plahte, Mestl, & Omholt 1998; Plahte & Kjøglum 2005). But, in practice, given the complexity of the network structures due to the large number of both components and interactions, such kind of analysis is quite hard or even unfeasible to be performed by hand. Thus, the need for the development of qualitative simulation algorithms, based on more sophisticated and adequate mathematical tools, and specifically tailored to capture the network dynamical properties that depend only on the model structure and are invariant for ranges of values of kinetic parameters.

The work, herein presented, is an effort in this direction,

and aims at providing a qualitative simulation algorithm of ODE models of GRN dynamics which 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 simulation 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 character 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 (Plahte & Kjøglum 2005; Veflingstad & Plahte 2007). Such a method suitably combined with QR key concepts computationally drives, starting from an initial state and constraints that define the parameter space domain, the construction of all possible state transitions along with the sets of symbolic inequalities on parameter values that hold when specific transitions occur.

Related work

Since frameworks for phenomenological modeling of GRNs by ODE equations have been proposed (de Jong 2002; Glass & Kauffman 1973; Plahte, Mestl, & Omholt 1998), a rather significant number of efforts in developing analytical methods for their qualitative study has been made (Glass 1977; Hasty *et al.* 2001; Gouzè & Sari 2003; Plahte & Kjøglum 2005). But, due to the difficulty to perform by hand such kind of analysis, the actual application of these methods has been restricted to toy-examples of scarce biological interest. Pioneering work towards automated qualitative analysis and simulation of GRNs results in a computational tool, called GNA (de Jong *et al.* 2004). 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. But, the solution to this problem is not straightforward as (i) there exists 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 (Bacciotti 2003). GNA adopts the Filippov approach that results particularly popular and 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 solutions. As a consequence, GNA suffers from the same disadvantages that together with a further approximation introduced in the algorithm for computational problems might compromise its soundness and completeness (Dordan, Ironi, & Panzeri).

Therefore, the algorithm we propose aims at both overcoming the limits of GNA and providing a framework that, thanks to the continuity assumption, can be gradually extended to tackle wider and more and more realistic classes of models.

Theoretical background

Singular perturbation analysis: basic ideas

Singular perturbation analysis is a classical approach to study phenomena that occur at different time-scales (Holmes 1995). The dynamics of such phenomena are described by ODEs in which a small parameter multiplies either one of the derivatives or higher order derivative, that is by system equations of the form:

$$\begin{aligned} \epsilon\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x},\mathbf{y},\epsilon) \\ \dot{\mathbf{y}} &= \mathbf{g}(\mathbf{x},\mathbf{y},\epsilon) \end{aligned} \tag{1}$$

where the dot denotes differentiation with respect to the ordinary time t, $\mathbf{x}(t) \in R^m, \mathbf{y}(t) \in R^l$, $0 < \epsilon \ll 1$, and $\mathbf{f}, \mathbf{g}$ smooth functions of $\mathbf{x}, \mathbf{y}, t$.

Let us indicate Eq. (1) associated with appropriate initial conditions by $\mathcal{M}_\epsilon$, and the same initial value problem where $\epsilon = 0$ by $\mathcal{M}_0$. The system modeled by $\mathcal{M}_\epsilon$, called *full system*, is singularly perturbed if, as $\epsilon \to 0$, the solution of $\mathcal{M}_\epsilon$ 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-converge of $\mathcal{M}_\epsilon$ to $\mathcal{M}_0$ is called *outer region*.

Singular perturbation methods aim at calculating an approximate solution of $\mathcal{M}_\epsilon$ for $0 < \epsilon \ll 1$, and differ from each other for the way they calculate and combine the boundary-layer solution and outer solution. In outline, the fundamental idea underlying these methods is to calculate local solutions in both boundary-layer and outer region, and combine them to find the global approximate solution. The fast dynamics in the boundary-layer is studied by suitably scaling the time variable, namely $\tau = t/\epsilon$. Then, the full initial value problem turns into the boundary-layer system:

$$\begin{aligned} \mathbf{x}' &= \mathbf{f}(\mathbf{x},\mathbf{y},\epsilon) \\ \mathbf{y}' &= \epsilon\, \mathbf{g}(\mathbf{x},\mathbf{y},\epsilon) \end{aligned} \tag{2}$$

where the prime denotes the derivative with respect to τ. In the limit, the fast dynamics is obtained by solving:

$$\begin{aligned} \mathbf{x}' &= \mathbf{f}(\mathbf{x},\mathbf{y},0) \\ \mathbf{y}' &= 0 \end{aligned} \tag{3}$$

This system has a manifold of stationary points given by $\mathbf{f}(\mathbf{x},\mathbf{y},0) = 0$, called slow-manifold. The reduced system $\mathcal{M}_0$:

$$\begin{aligned} 0 &= \mathbf{f}(\mathbf{x},\mathbf{y},0) \\ \dot{\mathbf{y}} &= \mathbf{g}(\mathbf{x},\mathbf{y},0) \end{aligned} \tag{4}$$

describes the motion in the original time t along those points in the slow-manifold, $\mathbf{x} = \mathbf{x}(\mathbf{y})$, that satisfy suitable hypotheses among those stability (Tikhonov-Levinson theorem, 1952). Then, the outer solution is described by the equation:

$$\dot{\mathbf{y}} = \mathbf{g}(\mathbf{x}(\mathbf{y}),\mathbf{y},0) \tag{5}$$

Taken together, the reduced equation and the boundary-layer solution approximate the solution of $\mathcal{M}_\epsilon$ for small nonzero values of ϵ.

A computational framework for the analysis of GRN dynamics

Experimental and theoretical studies seem to confirm the adequacy of the following specific class of ODEs to describe the essential features of a wide range of regulatory systems, and, in particular, of the complex dynamics of GRNs:

$$\dot{x_i} = f_i(\mathbf{Z}) - \gamma_i x_i \quad i = 1, \ldots, n \tag{6}$$

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 $S : \mathrm{R}^+ \to [0, 1]$ 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 $q \to 0$ we have $S(x_j, \theta_{jk}, q) = 0$ (respectively 1) when the value of x_j is smaller (greater) than θ_{jk}.

Each x_i, defined in $\Omega_i \subset \mathrm{R}^+$, 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,

$$f_i(\mathbf{Z}) = \sum_{l=1}^{L_i} \kappa_{il} \prod_{\substack{j=1,n \\ k=1,n_j}} Z_{jk}^{\alpha_{jkl}} \tag{7}$$

where κ_{il} are real values that denote kinetic rate parameters, L_i is the possibly empty number of interactions that synthesize x_i, 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.

In the present paper we adopt a further assumption that sounds quite realistic:

Assumption $\mathcal{A}$. Every gene product only regulates one gene at each of its thresholds.

Mathematically, this assumption implies that each Z_{jk} only occurs in one equation. This simplifies the calculation of the slow–motion manifold that, otherwise, generally consists in an heavy, nonlinear computational problem.

To exemplify the concepts and the definitions as they are introduced, all through the paper we will consider the ODE model:

$$\begin{aligned} \dot{x_1} &= \kappa_{11}(1 - Z_{11})(1 - Z_{22}) + \kappa_{12}(1 - Z_{21}) - \gamma_1 x_1 \\ \dot{x_2} &= \kappa_{21}(1 - Z_{12}) - \gamma_2 x_2 \end{aligned} \tag{8}$$

where all parameters are strictly positive, and, the response function Z_{jk} is expressed by the standard Hill function $S(x_j, \theta_{jk}, q) = \frac{x_j^{1/q}}{x_j^{1/q} + \theta_{jk}^{1/q}}$, commonly used in the literature.

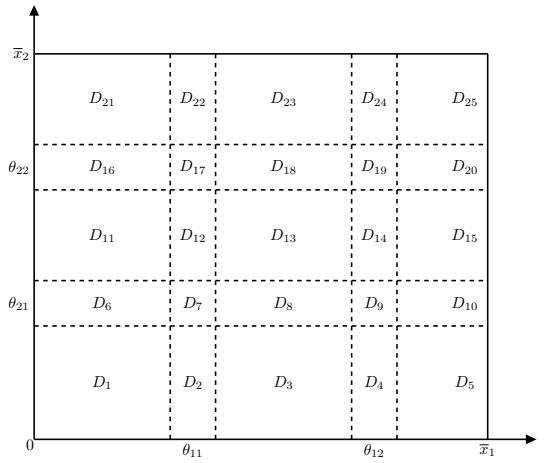


Figure 1: Partition of the phase-plane into regular and switching domains.

A - Regular and Switching Domains. Let us consider the n-dimensional vector of the state variables $\mathbf{x}$ whose domain $\Omega = \Omega_1 \times \ldots \times \Omega_n$ is given by the product of the domains $\Omega_i \subset \mathrm{R}^+$ of each of its component. The ordered set Θ_i of the n_i threshold values θ_{ij} associated with each x_i naturally induces a partition of Ω_i into qualitatively distinct domains. As a matter of fact, instead of the sharp value θ_{ij} we must consider a range of values around it, whose width, $\delta > 0$, is a monotonic function of the steepness parameter q with $\delta(q) \to 0$ for $q \to 0$, and characterizes the domain where the related response function takes values other than zero or one. Let us denote by $\underline{\theta}_{ij}$ and $\overline{\theta}_{ij}$ the values $\theta_{ij} - \delta/2$ and $\theta_{ij} + \delta/2$, respectively. Then, each Ω_i results from the product of open, $(\overline{\theta}_{ik}, \underline{\theta}_{i(k+1)})$, and closed, $[\underline{\theta}_{i(k+1)}, \overline{\theta}_{i(k+1)}]$, intervals $\mathcal{I}_{ik}$[1]. The whole system domain Ω is then partitioned, as showed in Fig. 1, into hyper-rectangles $D = \mathcal{I}_{1l_1} \times \ldots \mathcal{I}_{il_i} \ldots \times \mathcal{I}_{nl_n}, \quad l_i \in \{0, \ldots, n_{i+1}\}$.

In the set Δ of all the domains identified by the partition, we can distinguish the set $\Delta_s \subset \Delta$ of *switching domains* (SD) from the set $\Delta_r \subset \Delta$ of *regular domains* (RD), such that $\Delta = \Delta_s \cup \Delta_r$.

A domain D belongs to Δ_s if one, several, or all variables are at (one of) their thresholds or, equivalently, if it results from the product of at least one closed interval, e.g. D_6, D_{14}, D_{17}. Let $\sigma(D)$, $D \in \Delta_s$, be the switching order of D, i.e.the number of switching state variables, then those ones that assume values in a closed interval in D. A SD with $\sigma = n$ is called a *center*. In the example, the centers are D_7, D_9, D_{17}, and D_{19}.

A domain D belongs to Δ_r if it is an open set, e.g. D_1, D_5, D_{13}, and it is also called *box*.

The network dynamics in each domain $D \in \Delta$ is described by different models: the slow motion in $D_r \in \Delta_r$ is described by linear ODEs whereas the fast motion in $D_s \in \Delta_s$, or equivalently around the thresholds, is described

[1] $k = 0, \ldots, n_i$, where θ_{i0} and θ_{n_i+1} denote 0 and $\overline{x}_i = \max(x_i)$, respectively.

by nonlinear equations. Thus, the need to adopt different analysis strategies of the motion in regular and switching domains.

B - Motion equations in regular domains. In each box D_r, Z_{jk} equals either 0 or 1 in the step function limit. This simplifies Eq. (6) as they reduces to linear equations:

$$\dot{x_i} = \mu_i - \gamma_i x_i, \quad i = 1, \ldots, n \tag{9}$$

where μ_i depends on D_r, and is given by the sum of some κ_{il}. From Eq. (9) we can easily find the *focal point* $\mathbf{x}^* = \{x_j^* = \frac{\mu_j}{\gamma_j}\}$ the trajectories are heading towards. Herein, we assume that focal points do not belong to switching domains. If $\mathbf{x}^*$ belongs to the initial domain D_r, there is a stable point in it, called Regular Stable Point (RSP). Otherwise, the trajectories move to a switching domain adjacent to D_r.

For example, the motion equations (8) in the domain D_{11}, being $Z_{21} = 1$ and $Z_{11} = Z_{12} = Z_{22} = 0$, reduce to:

$$\begin{aligned} \dot{x_1} &= \kappa_{11} - \gamma_1 x_1 \\ \dot{x_2} &= \kappa_{21} - \gamma_2 x_2 \end{aligned} \tag{10}$$

whose trajectories move towards the focal point $\mathbf{x}^* = (\frac{\kappa_{11}}{\gamma_1}, \frac{\kappa_{21}}{\gamma_2})$. If $\mathbf{x}^* \in D_{13}$ the trajectories starting in D_{11} point to $\mathbf{x}^*$. Thus, they escape from D_{11}, and heading towards D_{13}, they first move to D_{12}.

C - Motion equations in switching domains. In a switching domain D_s we distinguish $\sigma(D_s)$ switching variables, $x_s \in [\underline{\theta_s}, \overline{\theta_s}]$, from $n - \sigma(D_s)$ regular ones x_r. For example, in the domain D_{12} ($\sigma(D_{12}) = 1$), x_1 is the switching variable while x_2 is the regular one.

Using singular perturbation analysis as properly adapted to study the system (6) (Plahte & Kjøglum 2005), we can capture the salient features of the nonlinear dynamics in a switching domain D_s, and determine how the trajectories cross it to move towards other domains. In outline, (i) Eq. (6) related to the x_s variables are rewritten into the form (1) 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 write the full system:

$$\begin{aligned} q\dot{Z}_s &= d_s(Z_s, x_s)(f_s(\mathbf{Z_R}, \mathbf{Z_S}) - \gamma_s x_s) \\ \dot{x}_r &= f_r(\mathbf{Z_R}, \mathbf{Z_S}) - \gamma_r x_r \end{aligned} \tag{11}$$

where $\mathbf{Z_S}, \mathbf{Z_R}$ are the vectors of switching and regular variables Z_s and Z_r, respectively. For $q \to 0$, Eq. (11) are of the form (1), and then we study the fast dynamics in the boundary-layer in the scaled time variable $\tau = \frac{t}{q}$:

$$\begin{aligned} Z_s' &= d_s(Z_s, \theta_s)(f_s(\mathbf{Z_R}, \mathbf{Z_S}) - \gamma_s \theta_s) \\ x_r' &= 0. \end{aligned} \tag{12}$$

The solution of the system (12) associated with appropriate initial conditions gives us the boundary-layer solution. As $\mathbf{Z_R}$ is constant in any $D_s \in \Delta_s$, we focus on the switching variables Z_s only, and calculate the slow-manifold of the system (12) that is the set of solutions, for all s, of the stationary equations $Z_s' = 0$. We call *exit point set* (*EP*) the set of stable solutions satisfying the conditions of the Tikhonov-Levinson theorem, and we call *Z-cube* $\mathcal{Z}(D_s) = [0,1]^{\sigma(D_s)}$ the frame of reference where we search for an exit point. Then, under the hypothesis that at least one exit point $\tilde{\mathbf{Z}}_\mathbf{S}$ exists, the reduced equations are obtained by substituting it in the motion equations of regular variables:

$$\dot{x}_r = f_r(\tilde{\mathbf{Z}}_\mathbf{S}, \mathbf{Z_R}) - \gamma_r x_r. \tag{13}$$

The problem (13) 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.

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 $A(D_s)$ be the set of domains adjacent to D_s, and $\mathcal{D}_s = D_s \cup A(D_s)$. In the limit $q \to 0$, we define a map $\Sigma_{D_s} : \mathcal{D}_s \mapsto \mathcal{Z}(D_s)$ such that the interior of $\mathcal{Z}(D_s)$, $int(\mathcal{Z}(D_s))$, and its boundary are the images of D_s, and $A(D_s)$, respectively. More precisely, the domains $D_k \in A(D_s)$ are mapped into the faces of $\mathcal{Z}(D_s)$ when $D_k \in \Delta_s$ or into its vertices, otherwise. If an exit point exists in the interior of $\mathcal{Z}(D_s)$, and the associated reduced system has a critical point inside D_s then it exists a stable point in D_s, also called Singular Stable Point (SSP).

As an example let us consider the boundary-layer system in D_{12}:

$$\begin{aligned} Z_{11}' &= \frac{Z_{11}(1 - Z_{11})}{\theta_{11}}(\kappa_{11}(1 - Z_{11}) - \gamma_1 \theta_{11}) \\ x_2' &= 0 \end{aligned} \tag{14}$$

Its candidate exit point set *EP* = $\{0, 1, 1 - \frac{\gamma_1 \theta_{11}}{\kappa_{11}}\}$ includes the vertices, and a point in the interior of $\mathcal{Z}(D_{12})$, being $\mathcal{Z}(D_{12})$ the segment $[0,1]$, whose endpoints 0, 1, and its interior are the images of D_{11}, D_{13}, and D_{12}, respectively. Then, D_{11} and D_{13} are possible next traversed domains and D_{12} may contain a stable point.

D - Search for exit points. Let us observe that stationary points always exist on the vertices of $\mathcal{Z}(D_s)$. Then, for a vertex to be an exit point it should fulfill the stability condition. The computational cost of the search for all the other exit points could be quite heavy, but it can be considerably reduced by checking first a necessary condition for the existence of a stationary point on the other elements of $\mathcal{Z}(D_s)$. Let F be a face or the interior of $\mathcal{Z}(D_s)$. In (Veflingstad & Plahte 2007), it has been proved that necessary condition for the existence of a stationary point in F is that the Jacobian matrix $\mathbf{J}_F = (\frac{\partial f_i}{\partial Z_j})$ 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 it can be checked by using concepts from graph theory.

Let be $\tilde{F}$ any elements of $\mathcal{Z}(D_s)$, face, vertex, or interior, where a stationary point $\tilde{\mathbf{Z}}$ is located, and $\mathcal{L}_{\tilde{F}} = \{l : l \in \{1, \ldots, \sigma(D_s)\}, \quad \tilde{Z}_l \in \{0, 1\}\}$. $\tilde{\mathbf{Z}}$ is an exit point if: (i) it is stable, and (ii), if $\tilde{F}$ is on the boundary of $\mathcal{Z}(D_s)$, $Z_l, \forall l \in \mathcal{L}_{\tilde{F}}$ has to head towards $\tilde{F}$. The stability of a candidate exit point $\tilde{\mathbf{Z}}$ is checked by analyzing the spectrum of the Jacobian matrix, and the condition (ii) is verified when the sign of $Z_l'(\tilde{\mathbf{Z}})$, given by $f_l(\tilde{\mathbf{Z}})$, $\forall l \in \mathcal{L}_{\tilde{F}}$, is coherent with the value of $\tilde{Z}_l$, namely $f_l(\tilde{\mathbf{Z}}) > 0$ and $\tilde{Z}_l = 1$ or $f_l(\tilde{\mathbf{Z}}) < 0$ and $\tilde{Z}_l = 0$.

Remark 2. Let us remind that singular perturbation analysis works out in the limit $q \to 0$, but the calculated solution approximates the solution of Eq. (11) for sufficiently small q ($0 < q \ll 1$). Moreover, it can be proved that the Jacobian matrix is stable for $0 < q \ll 1$. This means that the exit points calculated in the limit also hold for sufficiently small q (Ironi, Panzeri, & Simoncini).

Remark 3. Let us observe that, under *Assumption* $\mathcal{A}$, the reduced equations are always independent of the Z_s occurring in the boundary-layer equations, and that the two sets of equations are mutually independent. For this reason, the behavior of the switching variables in a Z-cube is completely independent of the values of regular variables. Then, the study of the motion in a switching domain may be performed by first analyzing the switching variables, and then the regular ones.

A qualitative simulation algorithm

Among the generic qualitative approaches proposed in the literature, QSIM results to be both the most suitable formalism and algorithm to model and simulate models qualitatively abstracted from ODEs (Kuipers 1994). For this reason, the description of the specialized qualitative algorithm we are developing will be mostly given in accordance with the QSIM jargon.

Qualitative value. The qualitative value of each state variable x_i with domain $\Omega_i = [0, \overline{x}_i]$ is described in terms of its quantity space. In our context, the *quantity space* of x_i is defined by the ordered set Θ_i of its n_i threshold symbolic values. The set Θ_i also contains the endpoints of the domain of x_i, namely 0, and $\overline{x}_i$. The partition, induced by the state variable quantity-spaces, of the whole system domain Ω 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:

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

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 (de Jong *et al.* 2004), transitions are determined by the signs of $\dot{x}_i$. As $\dot{x}_i$ are defined by linear expressions, such signs are easily and uniquely determined by exploiting the inequalities that define the parameter space domain, and constrain the RSPs to belong to specific domains.

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 singular perturbation analysis: transitions from D towards adjacent D_k are determined by the locations of the exit points in the associated $\mathcal{Z}(D)$ that can be either on (i) the boundary of $\mathcal{Z}(D)$ or in (ii) its interior. Except in the case (ii), the number of exit points may be greater than one, and especially in a qualitative context where knowledge incompleteness on the parameter values is expressed by coarse constraints. Then, in general, the successors of D are not uniquely determined. But, through symbolic computation procedures, it is possible to calculate the set of inequalities, I_j^i, on parameters that hold when a transition from D_i to D_j occurs. Then, each path from D_i to D_j is clearly identified by the 3-tuple $\langle D_i, D_j, I_j^i \rangle$.

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*:

$QB = \langle D_0, I_0 \rangle, \langle D_0, D_1, I_1^0 \rangle, \ldots \langle D_k, D_i, I_i^k \rangle, \ldots \langle D_F, I_F \rangle$, 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, $\mathrm{BT}(D_0)$, 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 $\mathrm{BT}(D_0)$ defines a qualitative trajectory from D_0, that occurs when the values of parameters satisfy its related inequalities. Then, although, from a strictly computational point of view, the tree representation is a little bit less convenient than the graph one, it is much preferable as it enables an easier, more direct, and less ambiguous interpretability of the simulation outcomes.

Given as input, (i) n symbolic state equations of the form (6); (ii) n quantity spaces $\Theta_i = \{\theta_{ij}\}$ of the state variable x_i; (iii) $D_0 \in \Delta$; (iv) a set of symbolic inequalities I_0 on parameters defining a parameter space domain PSD_0, the algorithm provides as output $\mathrm{BT}(D_0)$. Its main steps are outlined in the following:

1. *Calculate* the qualitative state $QS(D_i)$ of the current domain D_i, or equivalently the possible transitions from D_i.
2. *Determine constraints* I_k^i on parameters for each path $e_k = D_i \to D_k$, where $D_k \in QS(D_i)$.
3. *Append* $\langle D_i, D_k, I_k^i \rangle$ to $\mathrm{BT}(D_0)$ if I_k^i are consistent with

the initial constraints I_0, and *mark* D_i as visited domain.

4. *Repeat* from step 1 for each not visited D_k.

Step 1 is the core of the overall algorithm, and it requires two separate algorithms to implement the different strategies adopted according to whether $D_i \in \Delta_r$ or $D_i \in \Delta_s$. Both algorithms calculate the conditions on parameters I_k^i that should hold for a possible transition from D_i to D_k. 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. Let us define I_k^i *consistent* with I_0 when it defines a not empty parameter space domain PSD_k^i such that $PSD_k^i \subseteq PSD_0$. Furthermore, as both the algorithms involve the calculation of the relative positions of two regions we define the relative position of D_1 with respect to D_2, indicated by $V(D_1, D_2) = \{v_j\}_{j=1}^n$ where $v_j \in \{-1, 0, 1\}$, by the comparison of the intervals defining D_1 and D_2, where D_1 and $D_2 \in \Delta$.

Transition 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^is. As $\delta(q) \to 0$ for $q \to 0$, for the sake of simplicity, 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. In outline, the algorithm performs the following steps:

1- *Calculate $A(D_i)$ and state equations in D_i.* The algorithm calculates the set $A(D_i)$, the symbolic state equations in D_i (9), and its focal point $\mathbf{x}^*$. As an example, let us consider the domain D_{11}: $A(D_{11}) = \{D_6, D_7, D_{12}, D_{16}, D_{17}\}$, the state equations (10) have a stationary solution $\mathbf{x}^* = (\frac{\kappa_{11}}{\gamma_1}, \frac{\kappa_{21}}{\gamma_2})$.

2- *Calculate I_k^i and possible transitions.* $\forall D_k \in A(D_i)$, 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 (9) are linear, and all the trajectories head towards a focal point $\mathbf{x}^*$ 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. Let $V(D_k, D_i) = \{v_j\}_{j=1}^n$ be the relative position of D_k with respect to D_i. I_k^i, initialized to I_0, is updated, $\forall j \in \{1, \ldots, n\}$, with either the inequality $(\frac{\mu_j}{\gamma_j} > \theta_{j(i+1)})$ if $v_j = 1$ or $(\frac{\mu_j}{\gamma_j} < \theta_{ji})$ if $v_j = -1$. Thus, 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. As an example, let us define I_0 as follows:

$$I_0 : (\frac{\kappa_{11} + \kappa_{12}}{\gamma_1} > \theta_{11}) \quad \wedge \quad (\theta_{21} < \frac{\kappa_{21}}{\gamma_2} < \theta_{22}) \quad (15)$$

Then, transitions from D_{11} in Fig. 1 are possible under the following conditions on parameters:

$$\begin{array}{lll} I_6^{11}: & \dot{x_2} < 0 \Rightarrow (\frac{\kappa_{21}}{\gamma_2} < \theta_{21}) & \text{to go to } D_6 \\ I_{12}^{11}: & \dot{x_1} > 0 \Rightarrow (\frac{\kappa_{11}}{\gamma_1} > \theta_{11}) & \text{to go to } D_{12} \\ I_7^{11}: & \dot{x_1} > 0, \dot{x_2} < 0 \Rightarrow I_6^{11} \wedge I_{12}^{11} & \text{to go to } D_7 \\ I_{16}^{11}: & \dot{x_2} > 0 \Rightarrow (\frac{\kappa_{21}}{\gamma_2} > \theta_{22}) & \text{to go to } D_{16} \\ I_{17}^{11}: & \dot{x_1} > 0, \dot{x_2} > 0 \Rightarrow I_{12}^{11} \wedge I_{16}^{11} & \text{to go to } D_{17} \end{array}$$

Among the inequalities given above, only I_{12}^{11} is not in disagreement with I_0. Thus, a possible transition from D_{11} towards D_{12} occurs when $I_{12}^{11} \wedge I_0$ holds. In other words, $QS(D_{11}) = \{D_{12}\}$.

3- *Check the existence of a* RSP *in D_i.* A stable point RSP exists in D_i, i.e. $D_i \in QS(D_i)$, if $\widetilde{PSD} \subseteq PSD_0$ and $\widetilde{PSD} \neq \emptyset$, where $\widetilde{PSD}$ is a parameter space domain defined by the set of inequalities $(\theta_{ji} < \frac{\mu_j}{\gamma_j} < \theta_{j(i+1)})$ $\forall j \in \{1, \ldots, n\}$.

Transition from a switching domain

Let $D_i \in \Delta_s$ be defined by the $\sigma(D_i)$ switching variables x_s with their values around θ_s and by the $n - \sigma(D_i)$ regular ones, x_r. In a switching domain, 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_j, Z_j such that the switching variables come first, and proceed first with the construction of the fast motion.

A - Fast motion. The study of the fast dynamics is performed in $\mathcal{Z}(D_i)$ in the scaled time, and aims at localizing the set of exit points in $\mathcal{Z}(D_i)$ 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. To this end, the algorithm proceeds as follows:

1- *Calculate the boundary-layer equations in D_i.* The algorithm symbolically calculates the boundary-layer equations (12) in the Z variables, and defines the mapping $\Sigma_{D_i} : \mathcal{D}_i \to \mathcal{Z}(D_i)$ that states a correspondence between D_i and its adjacent domains D_k with the interior and the elements on the boundary of $\mathcal{Z}(D_i)$. Let $\mathcal{F}$ be the set of both the faces and the interior of $\mathcal{Z}(D_i)$: its generic element $F = \Sigma_{D_i}(D)$, $D \in \Delta_s$ is either a face of $\mathcal{Z}(D_i)$ when $D \in A(D_i)$ or its interior when $D = D_i$.

To exemplify, let us consider the switching domain D_7, which is characterized by fast motion only as both variables are switching in it. In D_7 the boundary-layer system is given by:

$$\begin{aligned} Z'_{11} &= \frac{Z_{11}(1 - Z_{11})}{\theta_{11}}(\kappa_{11}(1 - Z_{11}) + \kappa_{12}(1 - Z_{21}) - \gamma_1\theta_{11}) \\ Z'_{21} &= \frac{Z_{21}(1 - Z_{21})}{\theta_{21}}(\kappa_{21} - \gamma_2\theta_{21}) \end{aligned} \quad (16)$$

The set $\mathcal{F}$ has five elements, the four faces F_k corresponding to D_2, D_6, D_8, D_{12}, and the interior of $\mathcal{Z}(D_7)$ that correspond to D_7. Moreover, the vertices of $\mathcal{Z}(D_7)$ are the

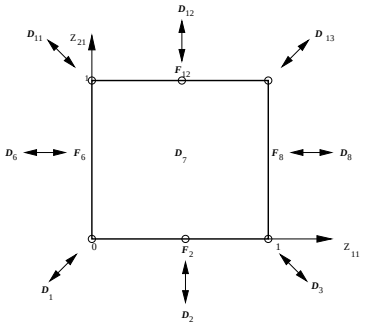


Figure 2: Correspondence between the domains in the phase-plane and the elements of $\mathcal{Z}(D_7)$. The candidate exit-points are denoted by an empty circle.

images, through the mapping Σ, of the adjacent regular domains D_1, D_{11}, D_3, and D_{13}.

2- *Search for stationary points.* Let us denote by EP the set of stationary points, initially made up of the vertices of $\mathcal{Z}(D_i)$. The set of the candidate exit points EP is updated by the possible stationary points on each element of $\mathcal{F}$, that under the *Assumption* $\mathcal{A}$ contains at most one stationary point. To this end, the algorithm symbolically calculates, $\forall F \in \mathcal{F}$, the Jacobian matrix $\mathbf{J}_F$, obtained by removing, $\forall i \in \mathcal{L}_F$, the i-th rows and columns from the Jacobian matrix of the system, and by computing its elements on F.

The Jacobian matrices associated with the elements of $\mathcal{F}$ in the example above are:

$$\mathbf{J}_{F_7} = \begin{pmatrix} -\kappa_{11} & -\kappa_{12} \\ 0 & 0 \end{pmatrix}; \qquad \mathbf{J}_{F_2} = (-\kappa_{11});$$
$$\mathbf{J}_{F_6} = (0); \qquad \mathbf{J}_{F_8} = (0); \qquad \mathbf{J}_{F_{12}} = (-\kappa_{12})$$

As the presence of a non-zero loop is a necessary condition for the existence of a stationary point, the algorithm first searches for a non-zero loop involving all variables in $\mathbf{J}_F$: in case, it symbolically calculates the stationary point on F, and updates accordingly the set of candidate exit points EP.

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

3- *Calculate I_k^i and possible transitions by checking stability of stationary points.* The inequality set I_k^i, initialized to I_0, is calculated for each candidate exit point $\tilde{\mathbf{Z}}^{\mathbf{k}} = \{\tilde{Z}_s^k\} \in EP$ by requiring that each point fulfills stability conditions. In addition, for those $\tilde{\mathbf{Z}}^{\mathbf{k}}$ located on elements of $\mathcal{F}$, I_k^i is further constrained by the inequalities on parameters that impose $0 < \tilde{Z}_s^k < 1$ for each $\tilde{Z}_s^k \notin \{0, 1\}$. The algorithm checks the stability conditions (i) by analyzing the spectrum of the Jacobian matrix $\mathbf{J}_F$, and (ii) by imposing conditions on the sign of $Z_l'(\tilde{\mathbf{Z}}^{\mathbf{k}})$, given by $f_l(\tilde{\mathbf{Z}}^{\mathbf{k}})$, $\forall l \in \mathcal{L}_F$. The latter condition is easily checked as it is of the form $(f_l(\tilde{\mathbf{Z}}^{\mathbf{k}}) > 0)$ if $\tilde{Z}_l = 1$ and $(f_l(\tilde{\mathbf{Z}}) < 0)$ if $\tilde{Z}_l = 0$, while the former one is

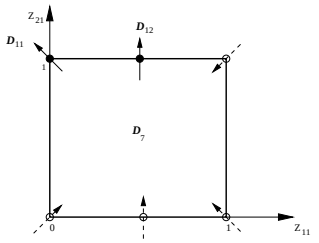


Figure 3: $\mathcal{Z}(D_7)$ and the exit-points denoted by a filled circle. The empty circles denote unstable stationary points that correspond to possible entrance points to the domain.

checked by using concepts from graph theory, and the usual definition of stability based on the sign of the eigenvalues of $\mathbf{J}_F$. Due to *Assumption* $\mathcal{A}$, $\mathbf{J}_F$ has just one element per row and per column. Then, reordering the variables leads to a matrix $\mathbf{J}_F$ block-structured, where each block is a permutation matrix associated with a sub-loop. It follows that the characteristic equation $|J_F - \lambda I| = 0$ is:

$$\prod_{i=1}^{m} (\lambda^{l(i)} + L_i) = 0 \tag{17}$$

where the roots of the equation above, λ_i, are the eigenvalues, m is the number of sub–loops of $\mathbf{J}_F$, l_i is the i–th sub–loop, $l(i)$ is the length of l_i and L_i is the loop product of l_i. As stability is guaranteed when the eigenvalues of $\mathbf{J}_F$ have not positive real part, we exclude the case $l(i) > 2$. Then, $\tilde{\mathbf{Z}}$ is stable if: (i) $\mathbf{J}_F$ has no blocks with dimension strictly greater than 2; (ii) in blocks with $l(i) = 1$, $L_i = b_i < 0$; (iii) in blocks with $l(i) = 2$, the product of the non-zero elements is negative.

The stable points located on $\mathcal{Z}(D_i)$ 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 the algorithm by applying the map Σ^{-1} to each element of $\mathcal{Z}(D_i)$ that contains an exit point. Let us observe that the remaining unstable stationary points in EP are possible entrance points to D_i.

Going back to the example, both exit points $\tilde{\mathbf{Z}}^{\mathbf{2}}$ and $\tilde{\mathbf{Z}}^{\mathbf{12}}$ fulfill the stability condition (i) as $-\kappa_{11} < 0$ and $-\kappa_{12} < 0$. The condition (ii) on variable $Z_l, l = 2$ requires that :

$$I_{s,2}: \quad f_2(\tilde{\mathbf{Z}}^{\mathbf{2}}) < 0 \quad \Rightarrow \quad \kappa_{21} - \gamma_2\theta_{21} < 0 \tag{18}$$
$$I_{s,12}: \quad f_2(\tilde{\mathbf{Z}}^{\mathbf{12}}) > 0 \quad \Rightarrow \quad \kappa_{21} - \gamma_2\theta_{21} > 0 \tag{19}$$

The inequality $I_{s,12}$ defined by (19) is compatible with (15), but the inequality $I_{s,2}$ is not. Then, $\tilde{\mathbf{Z}}^{\mathbf{2}}$ is removed from the exit point set. To be an exit point $\tilde{\mathbf{Z}}^{\mathbf{12}}$ must satisfy the condition:

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

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

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

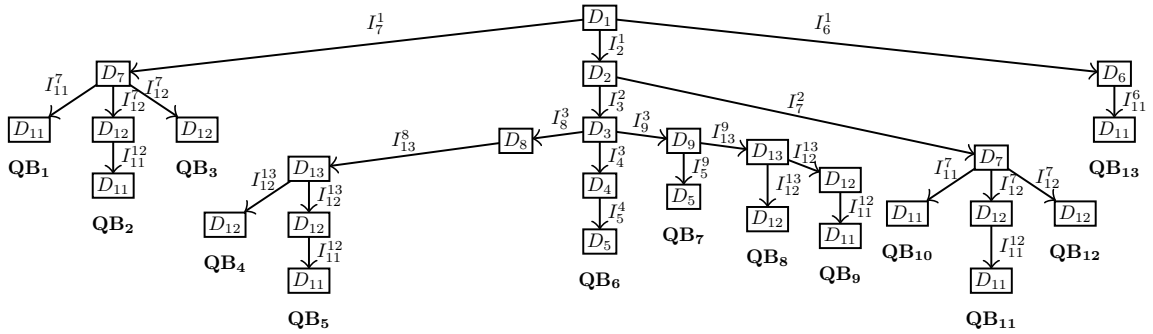


Figure 4: Behavior tree rooted in D_1.

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

B - Slow motion. The slow dynamics of regular variables x_r is studied in the normal time in the usual frame of reference, and it is reconstructed from the reduced system through the same symbolic procedure given for regular domains.

Locating SSPs in D_i. Let us consider the general case when the domain model is characterized by both switching and regular variables. The motion towards a SSP, that occurs in a sliding mode along a stable point in the slow-manifold of the boundary-layer system, is described, in the normal time, by the reduced system. 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 $\mathcal{Z}(D_i)$, and the regular variable rates are *zero* in a point inside D_i. The set $I_{r,i}$ of inequalities that checks the latter condition are defined by $\theta_{ji} < \frac{\mu_j}{\gamma_j} < \theta_{j(i+1)}$ $\forall j \in \{\sigma(D_i)+1, \ldots, n\}$. Let us observe that, due to *Assumption* $\mathcal{A}$, any $\mathcal{Z}(D_i)$ can at most contain one internal stable point. This occurs if all variables in $\mathcal{Z}(D_i)$ are involved in a loop.

As an example, let us consider again the domain D_{12}, and check if it contains one SSP. In D_{12} the boundary-layer system is given by Eq. (14), and the stationary state $\tilde{\mathbf{Z}}^{\mathbf{12}} = 1 - \frac{\gamma_1 \theta_{11}}{\kappa_{11}}$ in the interior of $\mathcal{Z}(D_{12})$ fulfills the stability conditions when it holds $I_0 \wedge I_{(0,1),12}$, being $I_{(0,1),12}$ the inequality set that constrains it to belong to the interval (0,1).

Finally, the stability condition $I_{r,12} : (\theta_{21} < \frac{\kappa_{21}}{\gamma_2} < \theta_{22})$ on the regular variable x_2 is provided by the reduced equation: $\dot{x_2} = \kappa_{21} - \gamma_2 x_2$. Thus, a SSP exists in D_{12} when parameter values satisfy the set of inequalities $I_0 \wedge I_{(0,1),12} \wedge I_{r,12}$.

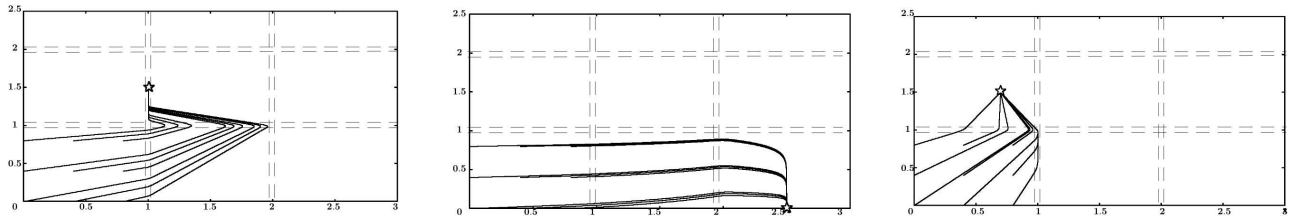
Results

The algorithm is currently under implementation. To illustrate the type of output, let us consider the results of a simulation of the ODE model (8) starting from D_1 with the parameter space defined by the inequalities I_0. The algorithm builds the behavior tree showed in Fig. 4, and calculates the inequalities on parameters, listed in Fig. 5(b), that are associated with each path in $\mathrm{BT}(D_1)$. As $n = 2$, a representation in the phase plane of the trajectories described by the tree, possibly filtered as explained below, is also given (Fig. 5(a)). 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 SSP whereas the others are RSPs. These stable states are reached by different predicted qualitative behaviors, each of them occurring under specific constraints on parameters. For example, the trajectory QB_{13} starting from D_1, crossing D_6, and reaching a RSP in D_{11} is allowed when the inequalities I_6^1, I_{11}^6 and I_{11}, consistent with I_0, hold.

The qualitative simulation outcomes are numerically confirmed. In Fig. 6 we report some of the numerical simulations performed under different conditions. Fig. 6(a) shows trajectories, abstracted by QB_4 and QB_8, and characterized by a sliding motion along θ_{11} towards the SSP in D_{12}. The trajectories moving towards the RSP in D_5 (Fig. 6(b)) are abstracted by QB_6, and those moving towards the RSP in D_{11} (Fig. 6(c)) are abstracted by QB_1 and QB_{13}.

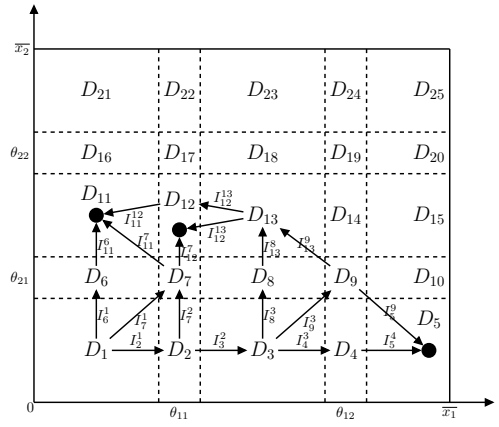
Soundness. The algorithm guarantees that the behavior tree captures all of the sound behaviors for values of q sufficiently small. The determination of a symbolic upper bound, $\overline{q}$, depending on parameters would better characterize the terms of validity of the generated behaviors. However, for a formal proof of soundness we need to prove that assuming stability instead of asymptotic stability does not affect the main results in singular perturbation analysis. Both problems are currently under study, and close to be solved.

Completeness. At present, 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 (i) 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 combined 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. We



(a) (b) (c)

Figure 6: Phase space plots of the numerical simulations performed with different parameter sets and initial conditions taken on an uniform grid of points in D_1. Common parameter values are: $\theta_{11} = \theta_{21} = 1$, $\theta_{12} = \theta_{22} = 2$, $q = 0.01$, $\kappa_{21} = 1.5$, $\gamma_2 = 1$. Other parameters are: (a) $\kappa_{11} = 2.5, \kappa_{12} = 2.5, \gamma_1 = 1$; (b) $\kappa_{11} = 25, \kappa_{12} = 2.5, \gamma_1 = 10$; (c) $\kappa_{11} = 0.7, \kappa_{12} = 0.7, \gamma_1 = 1$.



(a)

I_{12}^7, I_{12}	$I_0 \wedge (\frac{\kappa_{11}}{\gamma_1} > \theta_{11})$
I_3^2	$I_0 \wedge (\frac{\kappa_{12}}{\gamma_1} > \theta_{11})$
$I_5^9, I_9^3, I_4^3, I_5^4$	$I_0 \wedge (\frac{\kappa_{12}}{\gamma_1} > \theta_{12})$
$I_{11}, I_{11}^7, I_{11}^{12}$	$I_0 \wedge (\frac{\kappa_{11}}{\gamma_1} < \theta_{11})$
I_7^2	$I_0 \wedge (\frac{\kappa_{12}}{\gamma_1} < \theta_{11})$
I_5	$I_0 \wedge (\theta_{12} < \frac{\kappa_{12}}{\gamma_1} < \overline{x}_1)$
$I_2^1, I_6^1, I_7^1, I_8^3$ $I_{11}^6, I_{13}^8, I_{12}^{13}, I_{13}^9$	I_0

(b)

Figure 5: (a) Phase space representation of trajectories described by BT after filtering; • denotes a stable state. (b) Inequalities calculated by the algorithm.

are quite confident that the automatic analysis of the consistency of the whole sequence of inequalities that characterizes a behavior together with the solution of problem (i) will allow us to filter out all spurious solutions, and to prove the completeness of the algorithm.

Conclusion and future work

The qualitative simulation algorithm we propose works for models of GRNs with continuous sigmoid response functions. The continuity assumption 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. The 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.

As for symbolic calculus, the algorithm requires to tackle complex tasks, such as: (i) update an inequality set with 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* $\mathcal{A}$ they assume a linear form in the boundary layer, and then they can be straightforward solved and analyzed for stability. Also the solution of problems (i) and (ii) benefits from *Assumption* $\mathcal{A}$ as the inequalities are always linear. Then, thanks to the *Assumption* $\mathcal{A}$, and to algorithms proposed both by the literature and common symbolic computation package, such as Mathematica (Wolfram 2003), the tasks (i)-(iii) are simplified and feasible. As for the task (iv), it is performed by using cycle–detection algorithms and tools of matrix graph theory (Gross & Yellen 2006).

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 states the *proper* connection of the entrance points to the exit points associated with a switching domain. Moreover, the complex nonlinearities of the models we are interested in require to design methodological and computational methods to deal with possible aspects of the model dynamics that we have ignored herein, such as *limit cycles*.

Acknowledgement

The authors gratefully acknowledge Olivier Dordan, Erik Plahte and Valeria Simoncini for the useful discussions on the different methodological aspects and mathematical problems related to the work described in this paper.

References

Bacciotti, A. 2003. Some remarks on generalized solutions of discontinuous differential equations. *Int. Journal of Pure and Applied Mathematics* 10(3):257–266.

de Jong, H.; Gouzé, J. L.; Hernandez, C.; Page, M.; Sari, T.; and Geiselmann, J. 2004. Qualitative simulations of genetic regulatory networks using piecewise linear models. *Bulletin of Mathematical Biology* 66(2):301–340.

de Jong, H. 2002. Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology* 9(1):67–103.

Dordan, O.; Ironi, L.; and Panzeri, L. Some critical remarks on GNA. *in preparation*.

Glass, L., and Kauffman, S. A. 1973. The logical analysis of continuous, nonlinear biochemical control networks. *Journal of Theoretical Biology* 39(1):103–129.

Glass, L. 1977. Global analysis of nonlinear chemical kinetics. In Berne, B., ed., *Statistical Mechanics, Part B: Time Dependent Processes*, 311–349. Plenum Press, New York.

Gouzè, J. L., and Sari, T. 2003. A class of piecewise linear differential equations arising in biological models. *Dynamical systems* 17:299–316.

Gross, J., and Yellen, J. 2006. *Graph Theory and its Applications*. New York: Chapman & Hall/CRC Press.

Hasty, J.; McMillen, D.; Isaacs, F.; and Collins, J. 2001. Computational studies of gene regulatory networks: *In numero* molecular biology. *Journal of Computational Biology* 2:268–279.

Holmes, M. 1995. *Introduction to Perturbation Methods*. Berlin: Springer.

Ironi, L.; Panzeri, L.; and Simoncini, V. Matrix perturbation analysis for qualitative simulation of the dynamics of gene-regulatory networks. *in preparation*.

Kuipers, B. 1994. *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge*. Cambridge, MA: MIT Press.

Plahte, E., and Kjøglum, S. 2005. Analysis and generic properties of gene regulatory networks with graded response functions. *Physica D: Nonlinear Phenomena* 201(1-2):150–176.

Plahte, E.; Mestl, T.; and Omholt, S. W. 1998. A methodological basis for description and analysis of systems with complex switch-like interactions. *Journal of Mathematical Biology* 36(4):321–348.

Veflingstad, S. R., and Plahte, E. 2007. Analysis of gene regulatory network models with graded and binary transcriptional responses. *Biosystems* 90:323–339.

Wolfram, S. 2003. *The Mathematica Book*. Wolfram Media.