Chapter 2
Modeling and Analysis of Gene Regulatory Networks

Gilles Bernot, Jean-Paul Comet, Adrien Richard, Madalena Chaves, Jean-Luc Gouzé, and Frédéric Dayan

2.1 Introduction

In many ways, a living cell can be compared to a complex factory animated by molecular nanomachines, mainly proteins complexes. Hence it is easy to conceive that the expression of proteins, which are cellular effectors, cannot be constant. On the contrary, it is highly dependent on the general context; environmental conditions (pH, temperature, oxygenation, nutrient availability), developmental stage of an organism (fetal spectrum of proteins differ from adult proteins in mammals), response to a stress (UV irradiation, presence of a chemical toxic, osmotic pressure alteration) and even diseases (cancer, attack of a pathogen) are examples of contextual changes in the level of protein expression.

In order to understand this cellular state plasticity, a simplified view of this machinery, following general transfers of information according to the central dogma of molecular biology, is the sequence of events: (1) stimulation via a signaling pathway (e.g. presence of an environmental stimulation, followed by internal transduction of the signal), (2) effective stimulation of a transcription factor, (3) activation of the transcription of a particular gene, (4) production of messenger RNA (mRNA) (see Fig. 2.1), (5) translation of mRNA, i.e. production of a functional

G. Bernot (✉) · J.-P. Comet · A. Richard
I3S – UMR 6070 CNRS/UNSA, Algorithmes-Euclide-B, 2000 Route des Lucioles, B.P. 121, 06903 Sophia Antipolis, France
e-mail: bernot@unice.fr; comet@unice.fr; richard@unice.fr

M. Chaves · J.-L. Gouzé
Inria Sophia Antipolis Méditerranée, Biocore project-team, 2004 Route des Lucioles, 06902 Sophia Antipolis, France
e-mail: madalena.chaves@inria.fr; jean-luc.gouze@inria.fr

F. Dayan
SOBIOS SA, 2229 Route des Crêtes, 06560 Valbonne Sophia Antipolis, France
e-mail: frederic.dayan@sobios.com

F. Cazals and P. Kornprobst (eds.), *Modeling in Computational Biology and Biomedicine*,
DOI 10.1007/978-3-642-31208-3_2, © Springer-Verlag Berlin Heidelberg 2013

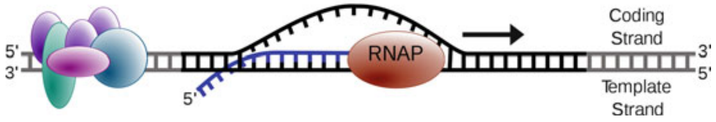


Fig. 2.1 Gene transcription (steps (3) and (4) in the Introduction): the enzyme RNA polymerase (RNAP) binds to DNA (*black* double strand) and produces a strand of RNA messenger (*blue* strand). This blue strand is a complementary copy of a sequence of DNA code (Image taken from [2])

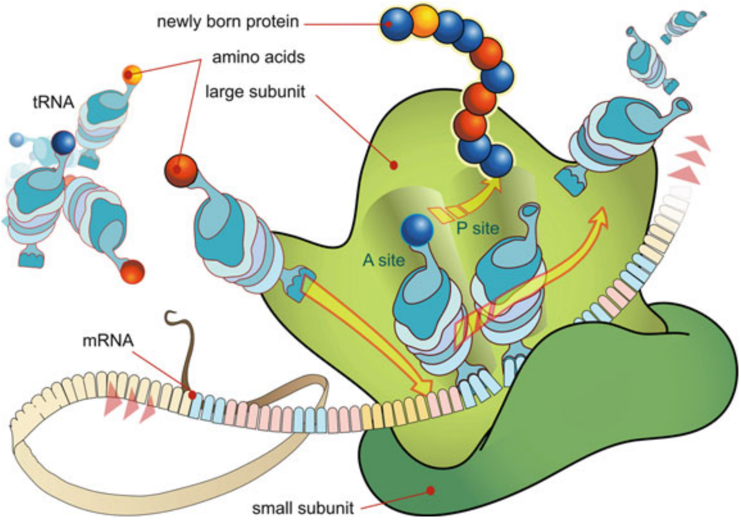


Fig. 2.2 Translation and protein synthesis (step (5) in the Introduction): a ribosome (two *green* units) is a large complex involving specific RNA (ribosomic RNA) complexed with proteins, synthesizing a polypeptide chain from a messenger RNA. Such a chain may form a protein on its own, or may contribute to a multimeric protein, see also Chap. 1 (Image taken from [1])

protein (see Fig. 2.2). Transcription factors are particular proteins that can recognize DNA motifs on the genome and consequently stimulate the transcription of a precise gene. A recognition motif is a short DNA sequence that is specific to a certain transcription factor. The transcription factor itself can be considered as a sensor of the cellular context.

2.1.1 Biological Systems and Experimental Techniques

A first example is provided by the Hypoxia Inducible Factor-1 (HIF-1), which is stimulated when oxygen pressure decreases: chemically, low intracellular oxygen concentration impairs the hydroxylation of the HIF-1α subunit, which leads to stabilization of this transcription factor (hypoxia signaling pathway). As a consequence, it recognizes Hypoxia Response Elements which are DNA motifs associated to a series of genes involved in adaptation to low pO_2. Among these genes, *erythropoietin* (or *epo*) is a well known inducer of red blood cells production; thus the low oxygen signal leads to secretion of the EPO protein and ultimately to an increase in erythropoiesis. This whole process aims at compensating poor oxygenation. In contrast, under normal pO_2 conditions, this stimulation of the *epo* gene is absent.

Another classical example, for bacterial cells, is the *lac* operon. An operon is a typical structure of bacterial genomes. It can be viewed as a "pack" of genes that are regulated by a unique promoter sequence. For instance, the *lac* operon is composed of the *lacZ*, *lacY* and *lacA* genes. In an environment with no glucose but with lactose available, the *lac* operon genes are transcribed, leading to lactose consumption as a source of energy. In that case, the level of transcription factors does not change directly, but the efficiency of transcription is regulated by a lactose repressor protein (impairing transcription in the absence of lactose) and by a Catabolite Activator Protein (which favors transcription in the absence of glucose).

These examples illustrate the complexity of gene regulation networks (GRN) for eukaryotic as much as prokaryotic cells. From an experimental point of view, biologists can access different intermediaries of these networks: genomic data (presence/absence of a DNA motif, complete sequence determination, mutations), mRNA quantification (large scale semi-quantitative screening with DNA arrays, or more focused and more precise analysis with Quantitative Polymerase Chain Reaction), quantitative gene activity measurements (luciferase reporter genes), quantitative protein detection (use of specific antibodies, fluorescent fusion proteins), or even molecular interactions estimation (semi-quantitatively with Förster/Fluorescence Resonance Energy Transfer, double hybrid, co-precipitation). Dynamics can also be followed thanks to tools like time-lapse microscopy or Fluorescence Recovery After Photo-bleaching microscopy on living cells (for an overview of some of these techniques see [15, 28]).

2.1.2 Mathematical Modeling

Therefore, large amounts of data, of more or less qualitative nature, are now available; one of the main challenges of molecular biology is to develop methodologies for using these data to address biological questions. Because of the complexity of the networks, it is necessary to design models, describing the dynamical functioning of the GRN. Indeed, the expression of genes, the concentrations of mRNA and proteins evolve with respect to time, and possibly converge toward some steady state, some periodic behavior or some other complex dynamical attractor. The emergence of these patterns from the dynamical interactions between the elements of the network, and the comparison with experimental data, will provide new keys to the comprehension of molecular biology, and enable scientists to solve important problems.

Yet, the methods for design and analysis of GRN are still quite recent. The Boolean and discrete approaches go back to works of Kauffmann (1969) and Thomas (1973), the continuous differential approach was applied by Goodwin (1963) to GRN, and the piecewise affine models were initiated by Glass and Kauffman [22]; for references see the review by de Jong [16]. There are still many open problems, mainly due to the large number of elements in a network. It is now possible to describe the behavior of a network with dozens of genes, but what about networks with several thousands of genes? These are still not attainable, even with the power of present computers and algorithmic methods.

As we have seen, the choice of a modeling approach is dependent on the type and amount of experimental data available, and on the nature of the biological questions asked by the modeler. In this chapter, two fundamental strategies will be detailed: continuous models and logical models. The first one gives quantitative predictions but needs quantitative biological data in order to fit parameters. The second is mainly based on a correct description of the logical links between biological entities (and is for instance particularly adapted to DNA array data that describe if a given gene is on/off). Nevertheless, both approaches can be used either to simulate biological phenomena or to predict properties that are intrinsically linked to the structure of the model, such as oscillatory or switch behaviors.

2.1.3 Chapter Overview

Public

This short introduction to GRN modeling is directed at Master level students whose background is either in the biological or the mathematical sciences.

Outline

A short overview of the main mathematical tools and concepts is provided, both on continuous (ordinary differential equations or hybrid systems) (see Sect. 2.2) and discrete (see Sect. 2.3) formalisms. For each type of formalism, simple examples of how to model genetic networks are worked out in more detail. Some successful applications of these methodologies to complex networks are also described.

2.2 Continuous and Hybrid Models of Genetic Regulatory Networks

The concentrations of molecular species (such as proteins or messenger RNAs) change in response to cellular signals. In this section, the concentrations are assumed to vary in a continuous manner, and their dynamical behavior will be described by systems of ordinary differential equations or the more abstract piecewise affine

(PWA) systems. These models often assume an homogeneous distribution of the molecules over a selected volume of space and describe, for instance, the dynamics of the concentration of some protein in a population of cells.

2.2.1 Challenges

The choice of appropriate variables is one of the first steps in the construction of a model for a biological network. The network is made of nodes (proteins, or RNA) and the edges usually describe the fact that some biochemical species acts positively or negatively on the variation with respect to time of some other biochemical species. Each variable (node) will play a different role in the behavior of the system, and have different degrees of relevance. Some variables can be measured experimentally, and are thus easier to compare to the model. Other variables may be easier to control from the exterior. Large systems of differential equations will require the introduction of a large number of parameters which will be unknown and should be estimated. In general, from a theoretical point of view, large dimensional systems are difficult to analyze and can only be studied through numerical simulations. Therefore, a fundamental step is the development of model reduction methods to simplify large networks and obtain more tractable systems of lower dimension, which are more easily studied in detail.

Two classical examples are the “positive” and “negative” feedback loops, formed by variables (proteins, for instance) that influence one another in a closed circuit, or loop. A circuit with two proteins that mutually repress or activate each other is a positive loop; if one of the interactions is a repression and the other an activation, then the circuit is a negative loop. Each of these two motifs appears frequently in GRN, and has a well known dynamical behavior; they can be combined with other motifs to represent the dynamics of complex regulatory networks. The negative loop is a system that generates oscillatory behavior, while the positive loop generates one or two stable steady states, and will be analyzed in detail in the next sections.

2.2.2 Mathematical Tools

This section quickly summarizes some basic mathematical results that will be useful in the analysis of systems of ordinary differential equations. For further details see, for instance [18].

2.2.2.1 Analysis of Two-Dimensional Systems

Consider a system with two variables, $x = (x_1, x_2)^t$, where each x_i represents the concentration of some molecular species, and x evolves in the region of space where all coordinates are either zero or positive (to be called *the positive orthant*). Define $\mathbb{R}_+ = [0, +\infty)$ and the two-dimensional positive orthant as $\mathbb{R}^2_+ = \mathbb{R}_+ \times \mathbb{R}_+$. The evolution of variable x_i along time is governed by a known function $f_i : \mathbb{R}^2_+ \to \mathbb{R}_+$, which depends on both variables. Given initial values $x_0 = (x_{10}, x_{20})$, solutions $x_i(t; x_{10}, x_{20})$ for $i = 1, 2$ may be found by solving the *initial value problem*:

$$\dot{x}_1 = f_1(x_1, x_2), \quad x_1(0) = x_{10},$$
$$\dot{x}_2 = f_2(x_1, x_2), \quad x_2(0) = x_{20}.$$

A sufficient condition to guarantee that this problem has a *unique solution* is that the functions f_1 and f_2 are continuous and have bounded, continuous derivatives, with respect to both variables. The positive orthant is invariant for this system if: whenever $x_{i0}(0) \geq 0$, then $x_i(t; x_{10}, x_{20}) \geq 0$ for all $t \geq 0$ ($i = 1, 2$). The following condition guarantees *invariance of the positive orthant*:

$$x_i = 0 \;\Rightarrow\; f_i(x_1, x_2) \geq 0, \;\; i = 1, 2, \tag{2.1}$$

which means that, at the boundary of the positive orthant, the vector field is either zero or points towards the interior of the orthant, thus preventing the variables to decrease to negative numbers. From now on, it will be assumed that functions f_i satisfy the required conditions, and that solutions of the initial value problem exist, are unique, and non-negative.

For most systems the f_i are nonlinear functions, and it is not possible to obtain closed form solutions of the initial value problem. However, qualitative analysis of the phase space can give a very good idea of the general behavior of the solutions. The signs of the vector field $(f_1(z), f_2(z))$ at each point $z \in \mathbb{R}^2_+$ indicate the direction of the solution at that point: for example, if $f_1(z) < 0$ and $f_2(z) > 0$, then the variable x_1 will decrease and x_2 will increase whenever a solution passes through the point z. The *nullclines* are curves that delimit regions of the plane where the sign of the vector fields is constant:

$$\text{Nullcline } i\text{:} \quad \Gamma_i = \{x \in \mathbb{R}^2_+ : \; f_i(x) = 0\}.$$

For an example see Fig. 2.5. The points of intersection of the nullclines are called the equilibria or *steady states* of the system:

$$x^* = (x_1^*, x_2^*) \in \mathbb{R}^2_+ : \quad f_1(x_1^*, x_2^*) = 0 \;\text{ and }\; f_2(x_1^*, x_2^*) = 0.$$

A steady state is a configuration of the system where both variables remain constant, and may be stable or unstable. To characterize this stability property suppose a small perturbation is applied to the initial condition, when $x(0) = x^*$. If the solution always returns back to x^* after a while, then the steady state x^* is stable; if the solution moves away from x^* without returning to the point, then the steady state x^* is unstable. The *basin of attraction of* x^* is the set of points $x_0 \in \mathbb{R}^2_+$ such that the solution $x(t; x_0)$ converges to x^* as time approaches infinity. The *stability of a steady state* x^* can be determined by computing the *Jacobian matrix and its eigenvalues,* $\lambda_\pm$ at that point:

$$J(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{pmatrix}.$$

The *steady state* x^* *is locally stable* if all eigenvalues of $J(x^*)$ have a strictly negative real part: $\mathrm{Re}(\lambda_\pm) < 0$. For two-dimensional systems, the stability can also be established by looking at the trace and the determinant of the Jacobian matrix:

$$\mathrm{tr}(J(x)) = \frac{\partial f_1}{\partial x_1} + \frac{\partial f_2}{\partial x_2}, \quad \det(J(x)) = \frac{\partial f_1}{\partial x_1}\frac{\partial f_2}{\partial x_2} - \frac{\partial f_2}{\partial x_1}\frac{\partial f_1}{\partial x_2}.$$

The steady state x^* is locally stable if: $\mathrm{tr}(J(x^*)) < 0$ and $\det(J(x^*)) > 0$. Geometrically speaking, the equilibria in dimension two can be classified into saddle (one positive and one negative real eigenvalue), stable sink (two real negative eigenvalues), unstable sink (two real positive eigenvalues), stable focus (two complex conjugate eigenvalues with negative real part), unstable focus (two complex conjugate eigenvalues with positive real part), plus the non-generic cases.

2.2.2.2 Analysis of n-Dimensional Systems

This analysis can be extended to general systems of ordinary differential equations. Consider now a system with n variables $x = (x_1, \ldots, x_n)^t \in \mathbb{R}_+^n$, $f = (f_1, \ldots, f_n)^t$ with $f : \mathbb{R}_+^n \to \mathbb{R}^n$ and

$$\dot{x} = f(x), \quad x(0) = x_0. \tag{2.2}$$

For large n, it becomes difficult to perform the stability analysis for a general set of parameters, and so the steady states, the Jacobian matrix and its egenvalues will typically be computed numerically, for given sets of parameters. As for the two-dimensional systems, existence and uniqueness of solutions of Eq. (2.2) are guaranteed by sufficient conditions on f: each f_i is continuously differentiable. The invariance of the positive orthant may be checked by condition in Eq. (2.1) for $i = 1, \ldots, n$.

The *nullclines* corresponding to each variable can be similarly computed: $\Gamma_i = \{x \in \mathbb{R}_+^n : f_i(x) = 0\}$. The *steady states* are given by all points x^* such that $f_i(x^*) = 0$, for $i = 1, \ldots, n$. The *Jacobian matrix* is again obtained by computing the partial derivatives of f_i:

$$J(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}.$$

The eigenvalues of the Jacobian matrix at equilibria are computable, at least numerically. *Local asymptotic stability* of x^* arises if all eigenvalues have a strictly negative real part. *Global stability* may be established through a Lyapunov function (but they are not easy to find). A Lyapunov function for system $\dot{x} = f(x)$ is a continuously differentiable function $V : \mathbb{R}^n_+ \to \mathbb{R}_+$ satisfying $V(x) \geq 0$ for all $x \in \mathbb{R}^n_+$ with $V(x) = 0$ if and only if $x = x^*$, and $\frac{\partial V}{\partial x}\dot{x} \leq 0$.

2.2.2.3 Different Timescales: Tikhonov's Theorem

Systems in the form of Eq. (2.2) whose variables evolve at different timescales can often be simplified. The main idea is to separate the system into "fast" and "slow" variables, and assume that the "fast" variables reach a (quasi) "steady state". This method allows reducing system in Eq. (2.2) to a new system with less variables, but with essentially the same dynamical behavior. This method can be applied only under appropriate conditions (briefly stated below) which are known as Tikhonov's Theorem (see, for instance, [27]). Let $x \in \mathbb{R}^p_+$, $y \in \mathbb{R}^q_+$, and $\varepsilon \ll 1$ be a small real number. Consider a system of the form

$$\begin{cases} \dot{x} = f(x, y, \varepsilon), \\ \varepsilon\dot{y} = g(x, y, \varepsilon), \\ (x(0), y(0)) = (x_0, y_0), \end{cases} \tag{2.3}$$

with f and g sufficiently smooth, under the following hypotheses:

- **H1** (slow manifold): there exists a unique solution, $y = \tilde{g}(x)$, sufficiently smooth, of $g(x, y, 0) = 0$; the matrix $\partial g/\partial y(x, \tilde{g}(x), 0)$ has all eigenvalues with strictly negative real part;
- **H2** (reduced system): the scalar system $\dot{x} = f(x, \tilde{g}(x), 0)$, $x(0) = x_0$ has a solution $x^0(t)$ on an interval $[0, T]$ ($0 < T < \infty$);
- **H3**: y_0 is in the basin of attraction of the steady state $\tilde{g}(x_0)$ of the fast system $\dot{\xi} = g(x, \xi, 0)$.

If hypotheses H1-H3 are satisfied, the system in Eq. (2.3) admits a solution $(x^\varepsilon(t), y^\varepsilon(t))$ on $[0, T]$; in addition, $\lim_{\varepsilon \to 0^+} x^\varepsilon(t) = x^0(t)$ and $\lim_{\varepsilon \to 0^+} y^\varepsilon(t) = y^0(t) = \tilde{g}(x^0(t))$, uniformly on time on any closed interval contained in $(0, T]$.

The variables y are "faster", since $\dot{y}$ evolves very rapidly when compared to $\dot{x}$. Hypothesis H1 means that y evolves rapidly to a *quasi steady state* value, $y = \tilde{g}(x)$, depending only on x. This quasi steady state evolves on the slow time scale.

2.2.2.4 General Piecewise Affine Systems

The model has the general form

$$\dot{x}_i = f_i(x) - \gamma_i x_i, \qquad 1 \leq i \leq n, \tag{2.4}$$

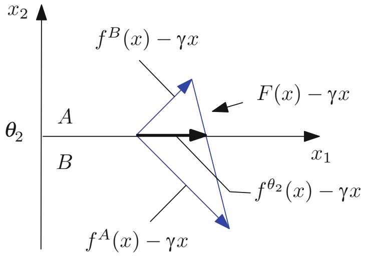


Fig. 2.3 Sliding mode solution. These type of solutions may occur when the vector fields in regions A and B point in opposite directions (f^A, f^B). At the boundary of A and B (the segment $x_2 = \theta_2$), the vector field may be defined as a convex combination of the two vector fields: $F(x) = \alpha f^A(x) + (1 - \alpha) f^B(x)$. The values of α range between $[0, 1]$, forming the convex hull cone. A sliding mode solution, with vector field $f^{\theta_2}(x)$, can be found by setting $x_2 = \theta_2$ and $F_2(x_1, \theta_2) - \gamma\theta_2 = 0$, and computing the appropriate value for α

where $x = (x_1, \ldots, x_n)^t$ is a non-negative vector of variables. The non-negative quantities $f_i(x)$ and $\gamma_i x_i$ represent production and loss (or transformation) rates for each variable x_i. The functions $f_i : \mathbb{R}_+^n \to \mathbb{R}_+$ will be constant in rectangular regions of the state space whose boundaries will be called *thresholds*. The $(n-1)$-dimensional hyperplanes defined by these thresholds partition the state space into hyper-rectangular regions which are called *domains* or boxes (see an example in Sect. 2.2.3.3). For any domain D, the function $f(x) = (f_1(x), \ldots, f_n(x))$ is constant for all $x \in D$, and it follows that the PWA system can be written as an affine vector field $\dot{x} = f^D - \gamma x, \ \ x \in D$ where f^D is constant in D.

The point $\phi(D) = \gamma^{-1} f^D \in \Omega$ is called the *focal point* for the flow in D, and globally attracts the trajectories until they reach the boundaries of the domain. The focal points define the possible transitions associated with the domain D; *the transition graph* describes these transitions and gives the qualitative behavior of the system. This graph can be efficiently computed, and its properties analyzed (see the example in Sect. 2.2.3.3).

On the thresholds, the solutions have to be appropriately defined, typically through a construction due to Filippov. This construction considers all the solutions, as if the step function could take all the values of the interval $[0, 1]$ on the threshold. To be more explicit, let $n = 2$ and consider two regular domains, A and B, separated by one threshold ($x_2 = \theta_2$), as in Fig. 2.3. Let $\overline{\text{co}}$ denote the closed convex hull of a set of vector fields. We define the differential inclusion

$$\dot{x} \in H(x), \tag{2.5}$$

with

$$H(x) = \overline{\text{co}} \left\{ f^D(x) - \gamma x : \ D^s \in \partial D \right\}, \text{if } x \in D^s, \text{a switching domain}$$

where $f^D - \gamma x$ is the vector field of the system on a regular domain D. In Fig. 2.3, there are only two regular domains (A and B) whose boundary contains the switching domain defined by $x_2 = \theta_2$, and hence the convex hull $H(x)$ is the cone formed by the two vectors $f^A(x) - \gamma x$ and $f^B(x) - \gamma x$.

A solution of Eq. (2.5) on $[0, T]$ *in the sense of Filippov* is an absolutely continuous (w.r.t. t) function $\xi(t; x_0)$ such that $\xi(0; x_0) = x_0$ and $d\xi/dt \in H(\xi)$ for almost all $t \in [0, T]$. It may give rise to a sliding mode along the plane defined by the threshold. See for instance [9] for a deeper analysis of solutions of PWA systems.

2.2.3 Methodological Developments

In general, there are n molecular species in a system, $x = (x_1, \ldots, x_n)^t \in \mathbb{R}^n_+$, and the changes in the concentration of species i result from a balance between production or synthesis processes ($g_i(x) \geq 0$) and degradation or transformation ($d_i(x) \geq 0$) into other species. Each species is thus characterized by an equation of the form:

$$\frac{dx_i}{dt} = g_i(x) - d_i(x). \tag{2.6}$$

The main problem is the choice of appropriate mathematical expressions of $g_i(x)$ and $d_i(x)$. These will depend on the modeling formalism and should reflect the observed dynamical behavior. For instance, for PWA systems, $g_i(x) = f_i(x)$ and $d_i(x) = \gamma_i x_i$, as seen in Eq. (2.4).

2.2.3.1 Modeling Transcription and Translation

In the classical sequence of events, transcription is initiated upon binding of a transcription factor to the gene. Besides transcription factors, other proteins may bind to specific sites of the gene and contribute to enhance (respectively, reduce) the transcription rate. Such proteins are called *activators* (respectively, *repressors*). In general, the binding of m molecules of protein A to the promoter site of a gene (D) to form a new complex (C), is represented as

$$D + mA \underset{k_2}{\overset{k_1}{\rightleftarrows}} C, \tag{2.7}$$

and can be modeled through the *law of mass-action*, where each reaction rate is proportional to a product of concentrations according to the stoichiometry:

$$\begin{cases} \dot{C} &= k_1 D A^m - k_2 C, \\ \dot{D} &= -\dot{C}. \end{cases} \tag{2.8}$$

If no other reactions take place, there is conservation of mass $D + C = D_T$ (the total amount of promoter sites, free or bound, remains constant), which can be used to reduce the number of variables. One can view A as an external variable. Using the observation that binding processes are typically faster than transcription, the system in Eq. (2.8) can be simplified through a process similar to Tikhonov's method, by setting the equations for C and D at "quasi steady state" ($\dot{C} \approx 0$), to obtain:

$$\begin{cases} C &= D_T \frac{A^m}{\theta_A^m + A^m}, \\ D &= D_T - C = D_T \frac{\theta_A^m}{\theta_A^m + A^m}, \end{cases} \tag{2.9}$$

with $\theta_A = (k_2/k_1)^{1/m}$. The amount of mRNA molecules (denoted M) produced may depend both on the concentration of DNA sites bound to an activator or repressor, and on the amount of free DNA sites. For simplicity, it will be assumed that the effect of activators and repressors can be modeled independently. Since binding of A to D is fast, the most natural form for the production of mRNA is a linear dependence on D and C. In addition, mRNA decays at a constant rate γ_M, which yields the equation:

$$\dot{M} = \alpha_0 D + \alpha_1 C - \gamma_M M. \tag{2.10}$$

In the case of an activator, the contribution of C to mRNA production is much larger than that of D, which can be represented by specifying $\alpha_1 \gg \alpha_0$. Defining $\kappa_0 = \alpha_0 D_T$ and $\kappa_1 = (\alpha_1 - \alpha_0) D_T$, substitution of the quasi-steady state expressions for C and D in Eq. (2.9) into Eq. (2.10) yields:

$$\dot{M} = \kappa_0 + \kappa_1 \frac{A^m}{\theta_A^m + A^m} - \gamma_M M. \tag{2.11}$$

In the case of a repressor, the contribution of C to mRNA production is much smaller than that of D ($\alpha_1 \ll \alpha_0$). Now define $\kappa_0 = \alpha_1 D_T$ and $\kappa_1 = (\alpha_0 - \alpha_1) D_T$, to obtain:

$$\dot{M} = \kappa_0 + \kappa_1 \frac{\theta_A^m}{\theta_A^m + A^m} - \gamma_M M \tag{2.12}$$

In either case, the parameter κ_0 ($\ll \kappa_1$) denotes a residual or basal activity. For further modeling aspects see also [3].

The next step is translation, or protein synthesis from mRNA. This is itself a sequence of several intermediate steps, and can be modeled as a linear function of the mRNA, together with a natural decay term:

$$\dot{P} = \kappa_2 M - \gamma_P P. \tag{2.13}$$

The translation and transcription system in Eqs. (2.10)–(2.13) can be further simplified by using the fact that mRNA degrades faster than protein, or $\gamma_M \gg \gamma_P$.

Consider the case of an activator as in Eq. (2.11) and change the time variable to $\tau = \gamma_P t$, to obtain:

$$\begin{cases} \frac{dM}{d\tau} &= \frac{\kappa_0}{\gamma_P} + \frac{\kappa_1}{\gamma_P} \frac{A^m}{\theta_A^m + A^m} - \frac{\gamma_M}{\gamma_P} M, \\ \frac{dP}{d\tau} &= \frac{\kappa_2}{\gamma_P} M - P. \end{cases} \tag{2.14}$$

For a fixed value of A, Tikhonov's theorem can now be applied with $y = M, x = P$, $\varepsilon = \gamma_P/\gamma_M$ and with $f(x, y, \varepsilon) = \frac{\kappa_2}{\gamma_P} y - x$, $g(x, y, \varepsilon) = \frac{\kappa_0}{\gamma_M} + \frac{\kappa_1}{\gamma_M} \frac{A^m}{\theta_A^m + A^m} - y$. Substituting the quasi steady state expression for mRNA into the protein Eq. (2.14), and rewriting the system in the original time variable, obtains the reduced system:

$$\dot{P} = \tilde{\kappa}_0 + \tilde{\kappa}_1 \frac{A^m}{\theta_A^m + A^m} - \gamma_P P, \tag{2.15}$$

where $\tilde{\kappa}_0 = \kappa_2 \kappa_0 / \gamma_M$ and $\tilde{\kappa}_1 = \kappa_2 \kappa_1 / \gamma_M$. This yields a dynamical equation for the protein concentration, directly dependent on the amount of activator (A). From now on, all the intermediate steps (the binding of A to the promoter and synthesis of mRNA) can be left out of the model.

The expression $h^+(x, \theta, m) = x^m/(\theta^m + x^m)$ (or *Hill function*) is known to fit well to synthesis and activity rates. Similarly, the inhibition function can be represented as: $h^-(x, \theta, m) = 1 - h^+(x, \theta, m) = \theta^m/(\theta^m + x^m)$. For gene regulatory networks, the exponent m is considered to be "large" ($m \geq 2$), according to experimental data [40]. Note that the qualitative form of $h^+(x, \theta, m)$ remains essentially unchanged for $m \geq 2$, with the same maximal and half-maximal values ($\max(h^-) = 1$ and $h^\pm(\theta, \theta, m) = 1/2$), the only difference being the steepness of the function around the value θ. For large m, the parameter θ has therefore a special meaning: it is a threshold value below which there is practically no activity and above which activity is (almost) maximal. In the limit as m tends to infinity, the Hill function becomes a step function, as described in Sect. 2.2.3.3.

2.2.3.2 Continuous Differential Systems for Genetic Network Models

To illustrate the modeling and analysis of complex GRN, consider a regulatory motif that appears frequently in genetic networks: two genes that mutually inhibit themselves or, more precisely, the protein A encoded by gene a represses transcription of gene b, and vice-versa (Fig. 2.4). The concentration of each protein can be described by $\dot{x}_j = \kappa_j M_j - \gamma_j x_j$, and each mRNA by an expression as in Eq. (2.12):

$$\dot{M}_j = \kappa_{j0} + \kappa_{j1} h^-(x_i, \theta_i, m_i) - \gamma_{Mj} M_j, \text{ for } j, i \in \{1, 2\} \text{ and } j \neq i. \tag{2.16}$$

Using the quasi-steady state assumption for the protein and mRNA equations, the system can be reduced to the dynamics of the protein concentrations, $\dot{x}_i = f_i(x_1, x_2)$ with (renaming constants):

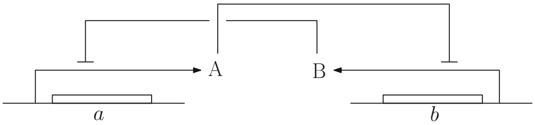


Fig. 2.4 Mutual inhibition between two genes. The white rectangles represent genes a and b, which are transcribed and then translated into the respective proteins, A and B (this is represented by the *arrows*). Each of these proteins inhibits the transcription of the gene of the other protein (A⊣B, B⊣A)

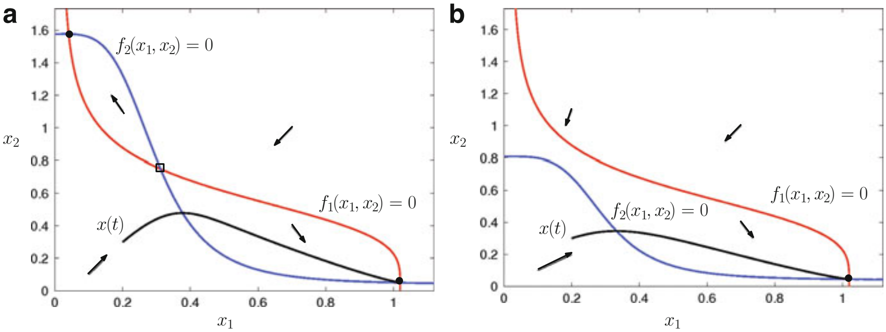


Fig. 2.5 Phase plane for the system of Eq. (2.17), for two different sets of parameters: (**a**) bistability, or (**b**) a single steady state. Stable steady states are marked by a *black circle*, and the unstable steady state by an *open rectangle*. In each region delimited by the nullclines ($f_1(x) = 0$ in *red*, $f_2(x) = 0$ in *blue*), the sign of the vector field of each coordinate is unchanged. One solution is shown in *black*. Parameter values for case (**a**): $\kappa_{10} = 0.02$, $\kappa_{11} = 1$, $\theta_1 = 0.3$, $\gamma_1 = 1$, $\kappa_{20} = 0.05$, $\kappa_{21} = 2$, $\theta_2 = 0.6$, $\gamma_1 = 1.3$, $m_1 = m_2 = 4$; for case (**b**) the only difference is $\kappa_{21} = 1$

$$\begin{cases} \dot{x}_1 &= \kappa_{10} + \kappa_{11}\frac{\theta_2^{m_2}}{\theta_2^{m_2}+x_2^{m_2}} - \gamma_1 x_1, \\ \dot{x}_2 &= \kappa_{20} + \kappa_{21}\frac{\theta_1^{m_1}}{\theta_1^{m_1}+x_1^{m_1}} - \gamma_2 x_2. \end{cases} \tag{2.17}$$

(Note that, in this section, the function f_i denotes the full right-hand side of the $\dot{x}_i$ equation, as in Sect. 2.2.2.1.) The dynamics of this system depend on the values of the parameters. The nullclines and steady states for system in Eq. (2.17) are represented in Fig. 2.5. Two distinct generic cases can be identified:

(a) m_i large, $\theta_i < (\kappa_{i0} + \kappa_{i1})/\gamma_i$, for all $i = 1, 2$. There are two stable and one unstable steady states. The stable steady states can be intuitively understood: one of the proteins is present at high concentration hence inhibiting transcription of the second gene, and implying that the second protein is only present in low concentration. This co-existence of two stable steady states is called *bistability*;

(b) m_i large, $\theta_i > (\kappa_{i0} + \kappa_{i1})/\gamma_i$, for some $i = 1, 2$. There is a unique steady state, which is stable.

The stability of the steady states obtains by studying the Jacobian matrix. In this example, it is clear that the equation of each nullcline, $f_i(x_1, x_2) = 0$, implicitly defines a function $x_2 = \tilde{f}_i(x_1)$. Thus the derivatives of f_i and $\tilde{f}_i$ are related by the Implicit Function Theorem:

$$0 = \frac{\partial f_i}{\partial x_1} + \frac{\partial f_i}{\partial x_2}\frac{dx_2}{dx_1} = \frac{\partial f_i}{\partial x_1} + \frac{\partial f_i}{\partial x_2}\frac{d\tilde{f}_i}{dx_1}. \tag{2.18}$$

The Jacobian matrix and its determinant can thus be written:

$$J(x) = \begin{pmatrix} -\frac{\partial f_1}{\partial x_2}\frac{d\tilde{f}_1}{dx_1} & \frac{\partial f_1}{\partial x_2} \\ -\frac{\partial f_2}{\partial x_2}\frac{d\tilde{f}_2}{dx_1} & \frac{\partial f_2}{\partial x_2} \end{pmatrix} \quad \text{and} \quad \det(J) = \frac{\partial f_1}{\partial x_2}\frac{\partial f_2}{\partial x_2}\left(\frac{d\tilde{f}_2}{dx_1} - \frac{d\tilde{f}_1}{dx_1}\right). \tag{2.19}$$

Therefore, its trace and determinant at a steady state x^* are

$$\begin{aligned} \mathrm{tr}(J^*) &= -(\gamma_1 + \gamma_2), \\ \det(J^*) &= \gamma_2 \kappa_{11} \frac{m_2 \theta_2^{m_2} (x_2^*)^{m_2 - 1}}{(\theta_2^{m_2} + (x_2^*)^{m_2})^2} \left(\frac{d\tilde{f}_2}{dx_1}(x^*) - \frac{d\tilde{f}_1}{dx_1}(x^*)\right), \end{aligned} \tag{2.20}$$

where $d\tilde{f}_i/dx_1(x^*)$ denote the slope of the curves $\tilde{f}_i$ at x^*. It is clear that the trace is always negative. For the steady states near one of the axis (one of the proteins at low concentration), it holds that $0 > d\tilde{f}_2/dx_1(x^*) > d\tilde{f}_1/dx_1(x^*)$, and therefore the determinant is positive – these are stable steady states. The middle steady state is unstable, since the opposite inequality holds and the determinant is negative.

This example is also known as *the bistable switch*, as only an external stimulus can force the system to evolve, or switch, from one steady state to the other (see discussion on Sect. 2.2.4).

2.2.3.3 Piecewise Affine Systems for Genetic Network Models

As seen above, the analysis of the dynamics of a dynamical system described by differential equations can be quite complicated in dimension greater than two. We are looking for a more algorithmic approach, easily implementable on a computer. We will consider a *qualitative* description of the bistable switch, corresponding to the case $m_i \to \infty$ where sigmoidal functions h^- become step functions. This is an approximation of the "real" system, done for an easier comprehension.

The formalism is as described in Sect. 2.2.2.4. The functions f_i now represent the dependence of the rate of synthesis of a protein encoded by gene i on the concentrations x_j of the other proteins in the cell. The term $\gamma_i x_i$ represents the degradation rate of protein x_i. The functions $f_i : \mathbb{R}_+^n \to \mathbb{R}_+$ can be written as

$$f_i(x) = \sum_{l \in I} \kappa_{il} b_{il}(x), \tag{2.21}$$

where $\kappa_{il} > 0$ is a rate parameter, $b_{il} : \mathbb{R}^n_+ \to \{0,1\}$ is a boolean-valued regulation function, and I is an index set. The regulation functions b_{il} capture the conditions under which the protein encoded by gene i is synthesized at a rate κ_{il}. These conditions are written down as combinations (sums of products) of step functions $s^+, s^- : \mathbb{R}_+ \times \mathbb{R}_+ \to \{0,1\}$, where $s^+(x_j, \theta^i_j) = 1$ if $x_j > \theta^i_j$, and $s^+(x_j, \theta^i_j) = 0$ if $x_j < \theta^i_j$, and $s^-(x_j, \theta^i_j) = 1 - s^+(x_j, \theta^i_j)$. The parameters θ^i_j are threshold concentrations.

This class of PWA systems was first introduced by Glass and Kauffman [22], and is widely used for modeling genetic regulatory networks [9, 17, 22, 34]. Step functions are not defined at threshold points, but solutions of the system "across" or "along" a threshold can still be defined in the sense of Filippov, as the solutions of differential inclusions, as shown in Sect. 2.2.2.4 and Fig. 2.3.

In the PWA formalism, the bistable system in Eq. (2.17) is defined inside the (invariant) set $\Omega = [0, \kappa_1/\gamma_1] \times [0, \kappa_2/\gamma_2]$. Assuming for the sake of simplicity that $\kappa_{10} = \kappa_{20} = 0$, one gets the equations:

$$\begin{cases} \dot{x}_1 &= \kappa_1 s^-(x_2, \theta_2) - \gamma_1 x_1, \\ \dot{x}_2 &= \kappa_2 s^-(x_1, \theta_1) - \gamma_2 x_2. \end{cases} \tag{2.22}$$

The space of state variables Ω is now divided into four boxes, or *regular domains*, where the vector field is uniquely defined:

$$\begin{aligned} B_{00} &= \{x \in \mathbb{R}^2_+ :\ 0 < x_1 < \theta_1,\ 0 < x_2 < \theta_2\} \\ B_{01} &= \{x \in \mathbb{R}^2_+ :\ 0 < x_1 < \theta_1,\ \theta_2 < x_2 < \kappa_2/\gamma_2\} \\ B_{10} &= \{x \in \mathbb{R}^2_+ :\ \theta_1 < x_1 < \kappa_1/\gamma_1,\ 0 < x_2 < \theta_2\} \\ B_{11} &= \{x \in \mathbb{R}^2_+ :\ \theta_1 < x_1 < \kappa_1/\gamma_1,\ \theta_2 < x_2 < \kappa_2/\gamma_2\}. \end{aligned}$$

In addition, there are also *switching domains*, where the system is defined only as a differential inclusion, corresponding to the segments where each of the variables is at a threshold ($x_i = \theta_i$ and $x_j \in [0, \kappa_j/\gamma_j]$). In each of the four regular domains, the differential system is affine, and simple to study. In B_{00} for instance $\dot{x}_1 = \kappa_1 - \gamma_1 x_1,\ \dot{x}_2 = \kappa_2 - \gamma_2 x_2$, and the solution can easily be written and converges exponentially towards a steady state $(\kappa_1/\gamma_1, \kappa_2/\gamma_2)$. If we suppose that $\theta_i < \frac{\kappa_i}{\gamma_i}$, then this steady state is *outside* B_{00}, and the solution will switch to another system when it crosses one of the thresholds. This succession of possible transitions will result in a *transition graph*, describing the possible sequences of boxes.

For the bistable switch, there are two classical stable steady states, P_1 and P_2, and an unstable Filippov equilibrium point, P_3, analogous to a saddle point (see Fig. 2.6):

$$P_1 = \left(\frac{\kappa_1}{\gamma_1}, 0\right), \quad P_2 = \left(0, \frac{\kappa_2}{\gamma_2}\right), \quad P_3 = (\theta_1, \theta_2).$$

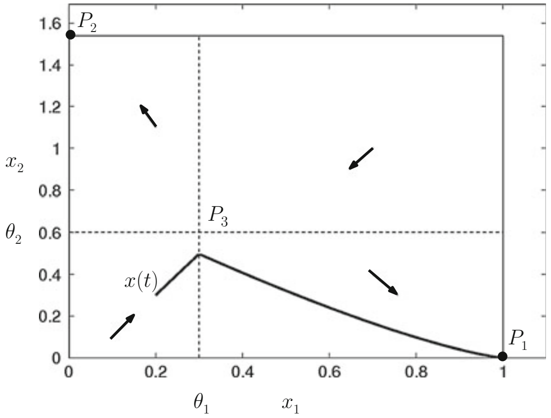


Fig. 2.6 Phase plane for the piecewise linear system of Eq. (2.22), with all parameters as in Fig. 2.5a (except $\kappa_{10} = \kappa_{20} = 0$ and $m_i = \infty$). The nullclines cannot be defined as in the continuous model (2.17) but, instead, the threshold values θ_1, θ_2 divide the plane into four rectangular regions, where the vector field is constant. There are still two stable steady states (P_1,P_2), but the unstable steady state is now defined as an unstable Filippov equilibrium (P_3). One solution is shown in black, which may be compared to that shown in Fig. 2.5a

P_1 and P_2 belong to the boundary of their respective domains (B_{10} and B_{01}), so that any trajectory entering one of these domains remains there. In contrast, trajectories starting in B_{00} or B_{11} will switch to another domain. This leads to the following transition graph for the bistable switch:

$$\begin{tikzcd}
\mathbf{01} & 11 \arrow[l] \arrow[d] \\
00 \arrow[u] \arrow[r] & \mathbf{10}
\end{tikzcd}$$

where P_1 is represented by $\mathbf{10}$, P_2 by $\mathbf{01}$, and P_3 is not represented in this diagram, as it is located in the middle, at the boundary of the four regular domains. This discrete abstraction (in the sense of hybrid systems) is a qualitative description of the behavior of the dynamical system. It can be used to check some qualitative properties of the system. Software exist that are able to compute the graph and check some of its properties, with *model checking* techniques.

2.2.3.4 Towards Control of Genetic Networks

An important problem is to be able to lead the system to a prescribed behavior. In control theory, the input represents the actions that a user (here a biologist) is able to exert on the system. From an experimental point of view, one common manipulation is to change the synthesis rate of messenger RNA by addition of a plasmid (a small unit of DNA that replicates within a cell independently of the chromosomal DNA). Therefore, it is reasonable to suppose that the input will act on the synthesis rates. The system with inputs can thus be written:

$$\dot{x}_1 = u\kappa_1 s^-(x_2, \theta_2) - \gamma_1 x_1, \quad \dot{x}_2 = u\kappa_2 s^-(x_1, \theta_1) - \gamma_2 x_2. \tag{2.23}$$

Moreover, different hypotheses can be made concerning the control u: it can take continuous values, or only qualitative values, e.g. two values $u_{\min}, u_{\max}$ and the value $u = 1$ corresponding to no control. The effective application of the control also depends on the measurements we are able to do, which can be continuous or qualitative. The ultimate goal can be to render the point P_1 (or P_2) stable in the whole space (see, for instance, [10]). This poses new and original problems to both mathematicians (to design the input laws) and to biologists (to implement them in the cell). This approach is used in synthetic biology, where artificially built biological circuits realize some tasks [21].

2.2.4 Success Stories

2.2.4.1 The Bistable Switch

The example studied in Sect. 2.2.3.2 is a frequently observed motif in biological networks (for instance, the system governing transcription of the *lac* operon contains a similar positive loop). These positive loops are typically observed composed with other motifs to form larger networks, but a very successful experiment by Gardner et al. in [21] showed that such a system can be synthetically implemented in a cell. Gardner et al. constructed plasmids containing two genes coding for proteins that repress each other, and inserted these plasmids in a strain of the bacterium *Escherichia Coli*. Transcription of each gene could be further controlled by an inducer, so that the whole system could be re-set. A mathematical model was used to determine appropriate intervals for some of the parameters, such as the maximal transcription rates, and concentrations of inducers. The experiments measure the expression of one of the genes, which we will call A, and show that the synthetic circuit indeed behaves as a bistable switch: following induction with inducer 1, gene A will be highly expressed, and stably maintain this expression for several hours after the inducer has been removed. Re-setting the system by means of inducer 2 will cause gene A to lower its expression to near zero and remain thus (until a new inducer is applied). Therefore, the synthetically constructed system has the capacity to stably exist in two distinct modes, corresponding to high or low expression of gene A, while the inducers are used to force the system to switch between these two modes. This is a clear observation of bistability in a genetic network.

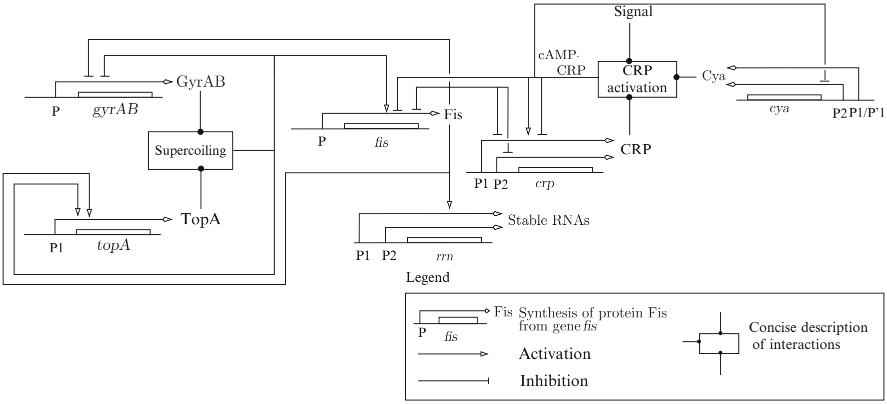


Fig. 2.7 Genetic network, including proteins and regulations that come into play during a nutritional stress response in *E.coli*: CRP activation module (Cya, CRP, Fis), DNA topology module (GyrAB, TopA, Fis), stable RNA output module (Rrn) (Adapted from [34])

2.2.4.2 The Carbon Starvation Response in *Escherichia coli*

One of the successful applications of the PWA formalism is the study of the genetic network that regulates nutritional stress response in *Escherichia Coli*. The model was developed by Ropers et al. [34] to describe the dynamics of a family of genes that regulate the carbon starvation response in *E.coli* (Fig. 2.7): *crp* (x_c), *cya* (x_y), *fis* (x_f), *gyrAB* (x_g), *topA* (x_t), and *rrn* (x_r). Nutritional stress is represented by an input $u \in \{0, 1\}$: $u = 0$ if carbon is present (no stress), and $u = 1$ in the absence of carbon. The PWA equations are shown in Table 2.1, and their mathematical study can be found in [24].

For the case $u = 1$, the asymptotic dynamics of the system in Table 2.1 satisfies:

$$x_c(t) \to \frac{\kappa_c^1 + \kappa_c^2 + \kappa_c^3}{\gamma_c} > \theta_c^3 > \theta_c^2; \quad x_y(t) = \theta_y^3 \text{ (in finite time)};$$
$$x_f(t) \to 0; \quad x_g(t) = \theta_g^2 \text{ (in finite time)}; , \quad x_t(t) \to 0. \tag{2.24}$$

Therefore, solutions converge to an equilibrium point in the sense of Filippov. In practice, there are sliding modes along the planes $x_g = \theta_g^2$ and $x_y = \theta_y^3$.

For the case $u = 0$, the asymptotic dynamics of the system in Table 2.1 can be reduced to the equations on x_g and x_f with:

1. $x_c(t) \to \frac{\kappa_c^1}{\gamma_c}$ and $x_y(t) \to \frac{\kappa_y^1+\kappa_y^2}{\gamma_y}$, after some finite time;
2. Sliding mode along the plane $x_t = \theta_t^1$ with the solution eventually jumping down to the region $x_t < \theta_t^1$, and staying there;

Table 2.1 Example piecewise affine model and parameter inequalities

$$
\begin{aligned}
\dot{x}_c &= \kappa_c^1 + \kappa_c^2 s^-(x_f,\theta_f^2)s^+(x_c,\theta_c^1)s^+(x_y,\theta_y^1)s^+(u,\theta_u) + \kappa_c^3 s^-(x_f,\theta_f^1) - \gamma_c x_c\\
\dot{x}_y &= \kappa_y^1 + \kappa_y^2[1 - s^+(x_c,\theta_c^3)s^+(x_y,\theta_y^3)s^+(u,\theta_u)] - \gamma_y x_y\\
\dot{x}_f &= \kappa_f^1[1 - s^+(x_c,\theta_c^1)s^+(x_y,\theta_y^1)s^+(u,\theta_u)]s^-(x_f,\theta_f^5)\\
&\quad +\kappa_f^2 s^+(x_g,\theta_g^1)s^-(x_t,\theta_t^2)s^-(x_f,\theta_f^5) \times [1 - s^+(x_c,\theta_c^1)s^+(x_y,\theta_y^1)s^+(u,\theta_u)] - \gamma_f x_f\\
\dot{x}_g &= \kappa_g^1[1 - s^+(x_g,\theta_g^2)s^-(x_t,\theta_t^1)]s^-(x_f,\theta_f^4) - \gamma_g x_g\\
\dot{x}_t &= \kappa_t^1 s^+(x_g,\theta_g^2)s^-(x_t,\theta_t^1)s^+(x_f,\theta_f^4) - \gamma_t x_t\\
\dot{x}_r &= \kappa_r^1 s^+(x_f,\theta_f^3) + \kappa_r^2 - \gamma_r x_r
\end{aligned}
$$

$$
\begin{aligned}
&0 < \theta_c^1 < \tfrac{\kappa_c^1}{\gamma_c} < \tfrac{\kappa_c^1+\kappa_c^2}{\gamma_c} < \theta_c^2 < \theta_c^3 < \tfrac{\kappa_c^1+\kappa_c^3}{\gamma_c}\\
&0 < \theta_y^1 < \tfrac{\kappa_y^1}{\gamma_y} < \theta_y^2 < \theta_y^3 < \tfrac{\kappa_y^1+\kappa_y^2}{\gamma_y}\\
&0 < \theta_f^1 < \tfrac{\kappa_f^1}{\gamma_f} < \theta_f^2 < \theta_f^3 < \theta_f^4 < \theta_f^5 < \tfrac{\kappa_f^1+\kappa_f^2}{\gamma_f}\\
&0 < \theta_g^1 < \theta_g^2 < \tfrac{\kappa_g}{\gamma_g}\\
&0 < \theta_t^1 < \theta_t^2 < \tfrac{\kappa_t}{\gamma_t}
\end{aligned}
$$

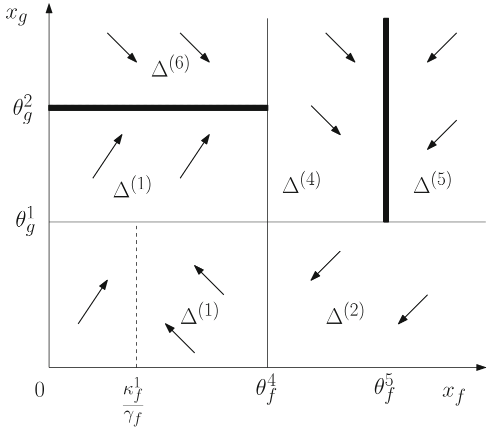


Fig. 2.8 Asymptotic behavior of the PWA in the (x_f, x_g) plane, for the case $u = 0$. *Thick black lines* indicate sliding modes [24]

3. Damped oscillations around the point $x_g = \theta_g^1$ and $x_f = \theta_f^4$. It is shown that all trajectories will asymptotically converge to this point, which is an equilibrium in the sense of Filippov;
4. $x_r(t) \to \frac{\kappa_r^1+\kappa_r^2}{\gamma_r}$ following the solution x_f.

There are also sliding modes along the segments: $x_g = \theta_g^2$ with $x_f < \theta_f^4$ and $x_g > \theta_g^1$ with $x_f = \theta_f^5$ (Fig. 2.8).

The PWA formalism allowed a more rigorous analysis of the complex network of carbon starvation response in *Escherichia Coli*. Major participants were identified as well as their roles in the presence or absence of nutritional stress. This PWA network could be further analyzed using a constraint-based method [14] to evaluate and improve the structure of the network under the parameter constraints on Table 2.1.

This example illustrates the sequence of events delineated in the introduction: the external or environmental signal is the presence ($u = 0$) or absence ($u = 1$) of a nutrient source, which may trigger the activation of the transcription of some genes, with production of the respective mRNA and proteins. Depending on the external signal, the response of the system is different, leading to low (respectively, high) expression of gene *fis* if nutrient is absent (respectively, present). Similar conclusions hold for the remaining genes, and many of the predictions have been experimentally observed.

2.3 Discrete Models of GRN

2.3.1 Challenges

In the previous section, the concentrations of molecular species are handled in continuous frameworks, using differential equations and it is shown that *some regions* (domains) of the space of state variables can be identified so that a more abstract continuous modeling framework can be applied, namely the piecewise affine systems. Going further into abstraction, many biological questions can be answered by only looking at the successive boxes that the cells under study can follow, forgetting the precise state in each box. Such models are called *discrete* models, as the state of a variable at a given time can be described by an integer value: the number of the interval containing the continuous state.

There are several motivations to consider qualitative models that forget the precise continuous state into a box:

- Numerous biological questions are themselves of qualitative nature;
- The *in vivo* measurement capabilities offer a resolution that does not allow to validate or refute a very precise value for the continuous parameters of a differential equation;
- Discrete descriptions can be easily modeled and simulated by computers, even when hundreds of variables are involved, allowing to experiment large genetic networks *in silico*.

These biological considerations have motivated the discrete approach proposed by René Thomas (presented in the next section).

Perhaps more importantly, discrete models can be studied using powerful techniques from computer science:

- *Combinatorial approaches*, often based on graph theory, are able to establish general laws about the link between the form of the interaction graph and the dynamic behavior of the system (e.g. there are behaviors that are unreachable for some interaction graphs);

- *Formal logics*, mainly temporal logics, can be used to automatically perform complex reasonnings about a given discrete model, so that discrete parameter values can often be *deduced* from behaviors observed *in vivo*.

All in all, discrete models are particularly well suited to perform qualitative reasonning in a computer aided manner and they help biologists to elucidate open questions about the functioning of many gene networks. It finally appears that computer *reasonning* capabilities are at least as useful as *simulation* capabilities. Discrete modeling is consequently able to provide the biologists with quick helpful information about open problems (possible behaviors, refutation of hypotheses, missing variables or missing interactions...); they are able to quickly prune some inconsistent intervals for the parameters in continuous models; they are also able to *suggest experimental plans* optimized to check a biological hypothesis.

2.3.2 Methodological Developments

2.3.2.1 René Thomas' Logical Method

R. Thomas' logical method consists in modeling the qualitative behavior of a gene network under the form of a finite state transition graph. This state transition graph is built from the interaction graph of the network together with logical parameters that describe the combined effects of switch-like interactions.

More precisely, the starting point of Thomas logical method is an *interaction* (or *regulatory*) *graph* G. The vertices, denoted from 1 to n, correspond to genes, and each arc $i \rightarrow j$ is associated with sign s_{ij} (Fig. 2.9). If s_{ij} is positive (resp. negative), it means that the protein encoded by i activates (resp. inhibits) the synthesis of the protein encoded by j. For every vertex i, we denote by G_i the set of *regulators* of i, that is, the set of vertices j such that $j \rightarrow i$ is an arc of G, and we denote by T_i the set of vertices regulated by i.

The **first step** of the logical method consists in associating with every vertex i a natural number b_i, called the *bound* of i, such that: $b_i \leq \mathrm{card}(T_i)$, and $b_i > 0$ if T_i is not empty. Then, $X_i = \{0, 1, \ldots, b_i\}$ corresponds to the possible *(concentration) levels* for the protein encoded by i, and $X = \prod_i X_i$ corresponds to the set of possible *(discrete) states* for the system.

The **second step** consists in associating with each interaction $i \rightarrow j$ an integer $t_{ij} \in X_i$, $t_{ij} > 0$, called the *logical threshold* of the interaction $i \rightarrow j$. It is required that, for every i, and for every integer $l \in X_i$, $l > 0$, there exists at least one interaction $i \rightarrow j$ such that $t_{ij} = l$ (condition C1). Then, at state $x = (x_1, \ldots, x_n) \in X$, we say that a regulator j of i is a *resource* of i if: $x_j \geq t_{ji}$ and $s_{ji} = +$ (effective activator), or $x_j < t_{ji}$ and $s_{ji} = -$ (ineffective inhibitor). In other words, j is a resource of i when its concentration level x_j "favors" the synthesis of the protein encoded by i. The set of resources of i at state x is denoted by $\omega_i(x)$. See Fig. 2.10 for an illustration.

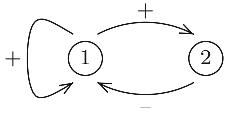
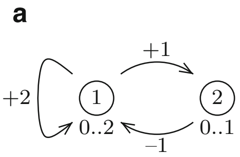


Fig. 2.9 An interaction graph



b

x	$\omega_1(x)$	$\omega_2(x)$
$(0,0)$	$\{2\}$	$\emptyset$
$(0,1)$	$\emptyset$	$\emptyset$
$(1,0)$	$\{2\}$	$\{1\}$
$(1,1)$	$\emptyset$	$\{1\}$
$(2,0)$	$\{1,2\}$	$\{1\}$
$(2,1)$	$\{1\}$	$\{1\}$

Fig. 2.10 Logical thresholds and resources. (**a**) The interaction graph of Fig. 2.9 together with the bounds $b_1 = 2$ and $b_2 = 1$, and the logical thresholds $t_{11} = 2$, $t_{12} = 1$ and $t_{21} = 1$. (**b**) The table gives the set or resources $\omega_i(x)$ of $i = 1, 2$ according to the state x of the system

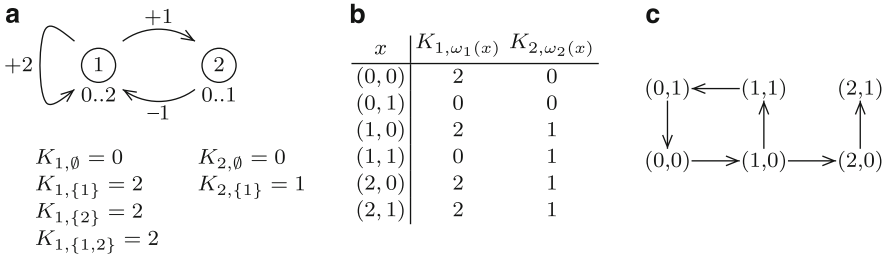
The ***third step*** *consists in associating with every vertex i and every set of regulators $\Omega \subseteq G_i$ a logical parameter $K_{i,\Omega} \in X_i$, in such a way that: for all i, and for all subsets Ω and Ω' of G_i, if $\Omega \subseteq \Omega'$ then $K_{i,\Omega} \leq K_{i,\Omega'}$* (condition C2). Intuitively, $K_{i,\Omega}$ is the level toward which i evolves (focal level) when Ω is the set of resources of i. In other words, at state x, the level of i is: increasing if $x_i < K_{i,\omega_i(x)}$; stable if $x_i = K_{i,\omega_i(x)}$; and decreasing if $x_i > K_{i,\omega_i(x)}$. The signs of the interactions of G are taken into account through the condition C2, which states that the focal level of i increases ($K_{i,\Omega} \leq K_{i,\Omega'}$) when its resources increase ($\Omega \subseteq \Omega'$), that is, when there are more activators and less inhibitors (so that the resources of i favor effectively the synthesis of the protein encoded by gene i).

More precisely, once logical parameters have been given, the behavior of the system is described by a directed graph, called *asynchronous state graph*, and defined by: the set of vertices is X; for every state x and every vertex i such that $x_i \neq K_{i,\omega_i(x)}$, there is an arc (or *transition*) from x to the state x' defined by:

$$x'_i = \begin{cases} x_i + 1 \text{ if } x_i < K_{i,\omega_i(x)} \\ x_i - 1 \text{ if } x_i > K_{i,\omega_i(x)} \end{cases} \quad \text{and} \quad x'_j = x_j \quad \forall j \neq i.$$

See Fig. 2.11 for an illustration.

If every variable is stable at state x (that is if $x_i = K_{i,\omega_i(x)}$ for every i), then x has no outgoing transition in the asynchronous state graph, and it corresponds to a *stable state* of the system. More generally, the *attractors* of the system are the smallest non-empty subsets of states $A \subseteq X$ that we cannot leave, that is, such that for every transition $x \rightarrow y$ of the state graph, if $x \in A$ then $y \in A$. So $\{x\}$ is an attractor if and only if x is a stable state. Attractors that are not stable states (attractors of size at least two) are called *cyclic attractors*, because once the system is inside such an attractor, it cannot reach a stable state, and thus, it necessarily describes sustained



$K_{1,\emptyset} = 0$ $\quad K_{2,\emptyset} = 0$
$K_{1,\{1\}} = 2$ $\quad K_{2,\{1\}} = 1$
$K_{1,\{2\}} = 2$
$K_{1,\{1,2\}} = 2$

x	$K_{1,\omega_1(x)}$	$K_{2,\omega_2(x)}$
$(0,0)$	2	0
$(0,1)$	0	0
$(1,0)$	2	1
$(1,1)$	0	1
$(2,0)$	2	1
$(2,1)$	2	1

Fig. 2.11 (**a**) The interaction graph of Fig. 2.9 together with bounds, logical thresholds and logical parameters. (**b**) The table gives the focal point of $i = 1, 2$ according to the state of the system. This table results from the one of Fig. 2.10 and the parameter values given in (**a**). (**c**) The asynchronous state graph resulting from the data given in (**a**). This asynchronous state graph can be easily built from the table given in (**b**)

oscillations. It is easy to show that from any initial state, there always exists a path leading to an attractor (and so, there always exists at least one attractor). It is in this weak sense that one can consider that attractors perform an attraction. The state graph of Fig. 2.11 contains a unique attractor, the stable state $(2,1)$, and indeed, from every initial state, there exists a path leading to this unique attractor.

2.3.2.2 Relationships with the Continuous Approaches

The dynamics of a network whose interaction graph is G may be described, using the piecewise affine model (see Sects. 2.2.3.3 and 2.2.2.4), by the following system:

$$\dot{x}_i = \kappa_i + \sum_{j \in G_i} \kappa_{ji} \cdot s^{(s_{ji})}(x_j, \theta_{ji}) - \gamma_i \cdot x_i \qquad (i = 1, \dots, n), \tag{2.25}$$

where: κ_i and γ_i are the "basal" synthesis rate and the degradation rate of i; G_i is the set of regulators j of i; κ_{ji}, θ_{ji} and $s_{ji} \in \{+,-\}$ are the synthesis rate, the quantitative threshold and the sign associated with the interaction $j \to i$; s^+ and s^- are the step functions defined in Sect. 2.2.2.4. We will now describe how to obtain, from the quantitative parameters $\kappa_i, \kappa_{ij}, \gamma_i$ and θ_{ij}, the qualitative parameters b_i, t_{ij} and $K_{i,\Omega}$ describing an asynchronous state graph abstracting the system (2.25). First, for all i, let $\Theta_i = \{\theta_{ij} | i \in G_j\}$ be the set of "out-going" quantitative thresholds of i, and set

$$b_i = card(\Theta_i) \qquad \textbf{(first step)}. \tag{2.26}$$

Then, consider the resulting set of discrete states $X = \prod_{i=1}^{n} \{0, 1 \dots, b_i\}$, and the discretization mapping

$$d : \mathbb{R}^n \to X, \quad d(x) = (d_1(x_1), \dots, d_n(x_n)), \quad d_i(x_i) = \text{card}(\{\theta \in \Theta_i | x_i \geq \theta_i\}).$$

Using this discretization map, let us define the logical thresholds and logical parameters as follows: for every i, every j such that $i \to j$ is an arc of G, and every $\Omega \subseteq G_i$,

$$t_{ij} = d_i(\theta_{ij}) \quad \textbf{(second step)}, \qquad K_{i,\Omega} = d_i(\kappa_i + \sum_{j \in \Omega} \kappa_{ji}/\gamma_i) \quad \textbf{(third step)}. \tag{2.27}$$

In this way, conditions C1 and C2 are satisfied, and Snoussi [35] proved that the resulting asynchronous state graph has the following property[1]:

Theorem 1 ([35]). *There exists a transition from the discrete state $x \in X$ to the discrete state $x' \in X$ if and only if there exists a solution ξ of the differential system (2.25), and $t' \in \mathbb{R}$, such that $d(\xi(t)) = x$ for all $0 \leq t < t'$ and $d(\xi(t')) = x'$ (i.e. there exists a solution ξ starting in the regular domain $\mathrm{d}^{-1}(x)$ that reaches the regular domain $\mathrm{d}^{-1}(x')$). Consequently: all solutions ξ such that, for all t, there exists at most one i such that $\xi_i(t) \in \Theta_i$, (and thus almost all solutions), are represented by a path in the asynchronous state graph.*

So each differential system of the form (2.25) is abstracted by an asynchronous state graph that can be built from G using the logical method. And conversely, thanks to the conditions C1 and C2, every asynchronous state graph S built from G with the logical method (from the parameters b_i, t_{ij} and $K_{i,\Omega}$) is the abstraction of an infinite set of differential systems of the form (2.25). (More precisely, S is an abstraction of (2.25) for every κ_{ij}, γ_i and θ_{ij} verifying the equalities (2.26) and (2.27).) Thus, the logical method may be seen as a *constructive* method for abstracting the infinite number of differential systems of the form (2.25) into a *finite* number of asynchronous state graphs.

2.3.2.3 Positive and Negative Circuits

In practice, while G may often be established from experimental data, (see for instance the database RegulonDB [20] about gene interactions in *E. coli*,) the bounds b, the logical thresholds t and the logical parameters K remain most often unknown. An interesting question is then: *which dynamical properties of a gene network can be inferred from its interaction graph, in the absence of information on the value of these parameters?* This question can be partially solved by studying positive and negative circuits of G.

A *positive* (resp. *negative*) *circuit* of G is a directed cycle containing an even (resp. odd) number of negative arcs. The interaction graph of Fig. 2.9 contains a positive circuit of length one ($1 \to 1$) and a negative circuit of length two ($1 \to 2 \to 1$).

[1] Provided that $(\kappa_i + \sum_{j \in \Omega} \kappa_{ji}/\gamma_i) \notin \Theta_i$ for all $\Omega \subseteq G_i$.

René Thomas highlighted the predominant (dynamical) role of positive and negative circuits by stating the following two rules [39]: (1) *A necessary condition for the presence of several stable states is the presence of a positive circuit in G.* (2) *A necessary condition for the presence of sustained oscillations is the presence of a negative circuit in G.*

These rules are "transversal" to the considered modeling framework in the sense that they have been proved for differential models [12, 23, 26, 29, 36–38], Boolean models [4,5,30] and discrete models [32,33]. The obvious interest of these two rules is that they relate the rather simple information contained in the interaction graph G of a network to its much more complex dynamical behaviors. In addition, multistationarity and sustained oscillations are, from a biological point of view, important dynamical properties, respectively associated with differentiation processes and homeostasis phenomena.

Inside Thomas' logical method, Thomas' rules take the following form:

Theorem 2 ([32, 33]).

1. *If G has no positive circuit, then for all bounds b, logical thresholds t, and logical parameters K, the resulting asynchronous state graph has at most one attractor.*
2. *If G has no negative circuit, then for all bounds b, logical thresholds t, and logical parameters K, the resulting asynchronous state graph has no cyclic attractor.*

In fact, if G has no positive (resp. negative) circuit, then every associated asynchronous state graph contains a unique attractor (resp. at least one stable state). These are simple consequences of the above theorem and the basic observation, already mentioned, that a state graph has always at least one attractor.

Most often, real interaction graphs contains positive and negative circuits, so that the previous theorem cannot be applied to obtain information on the dynamics of the system. However, the following theorem, which extends the first point of the previous theorem, can always be used (in the worst case, take $I = \{1, \ldots, n\}$).

Theorem 3 ([4,31]). *If I is a set of vertices such that every positive circuit of G has at least one vertex in I, then the asynchronous state graph resulting from the bounds b, logical thresholds t and logical parameters K contains at most $\prod_{i \in I}(b_i + 1)$ attractors.*

This theorem shows that the number of attractors is small when positive circuits are highly connected. The number of positive circuits is not the relevant parameter: if there is one million of positive circuits, but if all these circuits contain a vertex i with $b_i = 1$, then there are at most two attractors. Note also that the upper bound is tight in some cases. For instance, if G consists in a single vertex (vertex 1) with a positive arc $1 \to 1$, and if $b_1 = t_{11} = K_{1,\{1\}} = 1 > K_{1,\emptyset} = 0$, then the resulting asynchronous state graph has $2 = b_1 + 1$ attractors (that are stable states).

2.3.2.4 Formal Methods

SMBioNet. Given an interaction graph G, the number of asynchronous state graphs that we can build with the logical method is finite. For instance, 90 different asynchronous state graphs can be built from the interaction graph of Fig. 2.9.[2]

A natural question is then: *How to find, among this finite set of asynchronous state graphs, those that are coherent with biological observations (or hypothesis) on the dynamics of the system?*

Formal methods are useful to perform automatically such a selection. First, a *temporal logic* can be used to translate the dynamical observations into a temporal formula Φ, which can be handled by a computer. Then, one can use model-checking algorithms in order to check automatically if a given state graph (a model) satisfies or not formula Φ. Hence, to solve the question, the following (basic) approach can be used: *enumerate the different asynchronous state graphs, and select those that satisfy Φ using model-checking techniques*.

This enumerative approach has been implemented in a software called SMBioNet [8]. The temporal logic used is the well known Computational Tree Logic (CTL) [19], and the verification step is performed with the model-checker called NuSMV [11]. The Computational Tree Logic is briefly presented in the next paragraph. An illustration, on a real case, of this logic and the enumerative approach is then given in Sect. 2.3.3.

But before going further, let us briefly discuss the enumerative approach. The obvious limitation is that the number of state graphs to enumerate (which increases exponentially with the number of components) is often too huge to consider networks with more than ten genes or so. The obvious interest is that temporal logic and model checking allow us to handle automatically rather complex dynamical properties, and that the method is exhaustive: *all* the state graphs associated with G that are consistent with Φ are reported. (For other applications of formal methods in the context of gene regulatory networks, see for instance [6, 7, 13, 14] and the references therein.)

Computational Tree Logic. In an asynchronous state graph, a given state has generally several successors. So without additional information, all the successors are possible next states: the dynamical description is undeterministic. In other words, given an initial state x, the possible evolutions of the system are given by the set of paths starting from x, and these paths may be seen as a tree rooted at x.

[2] For the interaction graph of Fig. 2.9, we have $b_1 \in \{1,2\}$, $t_{11}, t_{12} \in \{1, b_1\}$ and $b_2 = \theta_{21} = 1$. If $b_1 = 1$, then $t_{11} = t_{12} = 1$, and there are 18 possible instantiations of the parameters K, which lead to a set of 18 different asynchronous state graphs. If $b_1 = 2$ there are two cases. First, if $t_{11} = 1 < t_{21} = 2$, there are 60 possible instantiations of the parameters K, which lead to a set $\mathcal{S}$ of 42 different asynchronous state graphs. Second, if $t_{11} = 2 > t_{21} = 1$, there are 60 possible instantiations of the parameters K, which lead also to a set of 42 different asynchronous state graphs, but 12 of them are contained in $\mathcal{S}$. Hence, the total number of asynchronous state graphs is $18 + 42 + 42 - 12 = 90$.

The Computational Tree Logic allows the formulation of properties on this tree. It is thus well adapted to formulate dynamical properties on undeterministic discrete dynamical systems. In particular, one can express possibilities in the future. For instance, the formula $\text{EF}(l_i = 0)$ expresses that "it is possible to reach a state in which the level of the ith component is 0", and the formula $\text{EG}(l_i = 0)$ expresses that "it is possible that the ith component stay for ever at the level 0".

Computational Tree Logic is defined in two steps. The first step consists in defining the *syntax* of the logic, i.e. rules for constructing formulas. The second step consists in defining the *semantic* of the logic, i.e. meaning of formulas.

The *syntax* of CTL is inductively defined by:

- For all genes i and integers k, $(l_i = k)$, $(l_i < k)$, $(l_i > k)$, $(l_i \leq k)$ and $(l_i \geq k)$ are (*atomic*) CTL formulas.
- If ϕ and ψ are two CTL formulas then $(\neg\phi)$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \Rightarrow \psi)$, $\text{EX}(\phi)$, $\text{EF}(\phi)$, $\text{EG}(\phi)$, $\text{E}(\phi \,\text{U}\, \psi)$, $\text{AX}(\phi)$, $(\text{AF}(\phi))$, $(\text{AG}(\phi))$, and $(\text{A}(\phi \,\text{U}\, \psi))$ are CTL formulas.

The *semantic* is given by the satisfaction relation $\models$ between the states x of a given asynchronous state graphs S and the CTL formulas ϕ. The semantic of atomic formulas is the following: $x \models (l_i = k)$ if and only if $x_i = k$; $x \models (l_i < k)$ if and only if $x_i < k$ and so on. The semantic of the classical logical connectives $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), and $\Rightarrow$ (implication) is usual: for instance, $x \models \phi \wedge \psi$ if and only if $x \models \phi$ and $x \models \psi$. The other connectives, called *temporal connectives*, are made with two letters and lead to formulas that are satisfied by a state x according to the set of infinite paths of S starting from x.[3] Intuitively, E and A correspond to existential and universal quantifiers respectively: E means "for at least one path" and A "for all paths". The other letters express properties along the paths: $\text{X}(\phi)$ means that ϕ is true at the neXt step, $\text{F}(\phi)$ means that ϕ is ture in the Future; $\text{G}(\phi)$ means that ϕ is Globally true, and $(\psi \,\text{U}\, \phi)$ means that ψ is always true Until ϕ becomes true. See Fig. 2.12 for an illustration. Formally, the semantic of temporal connectives is given by:

- $x \models \text{EX}(\phi) \iff$ there exists a successor of x satisfying ϕ.
- $x \models \text{AX}(\phi) \iff$ all the successors of x satisfy ϕ.
- $x \models \text{EF}(\phi) \iff$ there exists an infinite path starting from x which contains a state satisfying ϕ.
- $x \models \text{AF}(\phi) \iff$ all the infinite paths starting from x contain a state satisfying ϕ.
- $x \models \text{EG}(\phi) \iff$ there exists an infinite path starting from x which only contains states satisfyingϕ.

[3] An infinite path of S is an infinite sequence of states $x^0 x^1 x^2, \ldots$ such that, for all $k \in \mathbb{N}$: if x^k has a successor in S, then $x^k \to x^{k+1}$ is an arc of S, and $x^k = x^{k+1}$ otherwise.

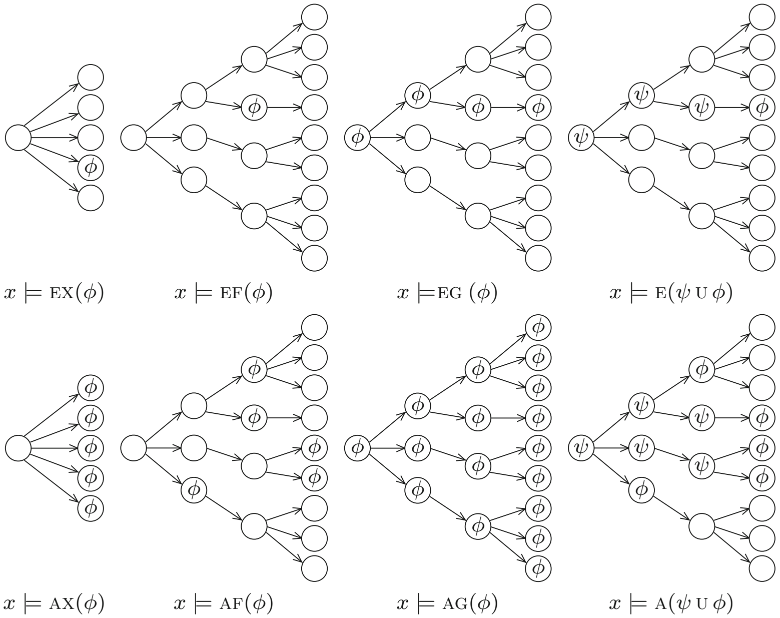

$x \models \text{EX}(\phi)$
$x \models \text{EF}(\phi)$
$x \models \text{EG}(\phi)$
$x \models \text{E}(\psi \,\text{U}\, \phi)$
$x \models \text{AX}(\phi)$
$x \models \text{AF}(\phi)$
$x \models \text{AG}(\phi)$
$x \models \text{A}(\psi \,\text{U}\, \phi)$
ϕ
ψ

Fig. 2.12 Illustration of the semantic of temporal connectives. Paths starting from the root correspond to paths starting from x. The *circles* containing ϕ (resp. ψ) corresponds to states satisfying ϕ (resp. ψ)

- $x \models \text{AG}(\phi) \iff$ all the infinite paths starting from x only contain states satisfying ϕ.
- $x \models \text{E}(\psi \,\text{U}\, \phi) \iff$ there exists both an infinite path $x^0 x^1 x^2 \dots$ with $x = x^0$ and $i \in \mathbb{N}$ such that $x^i \models \phi$ and $x^j \models \psi$ for all $j < i$.
- $x \models \text{A}(\psi \,\text{U}\, \phi) \iff$ for all infinite paths $x^0 x^1 x^2 \dots$ with $x = x^0$, there exists $i \in \mathbb{N}$ such that $x^i \models \phi$ and $x^j \models \psi$ for all $j < i$.

If all the states of a state graph S satisfy a given formula, we say that S satisfies this formula. For instance, the formula

$$(l_1 \geq 1) \Rightarrow \text{AX}(\text{AF}(l_1 \geq 1)),$$

is satisfied by S if and only if for every path $x^0, x^1, \dots$ starting from an initial state x^0, with $x_1^0 \geq 1$, there exists $t > 0$ such that $x_1^t \geq 1$. In other words, the formula means that when the level of the first component is at least one at time t, then it will be at least one at another time $t' > t$. The asynchronous state graph of Fig. 2.11 satisfies this property. The formula

$$(l_1 = 0) \Rightarrow \text{AG}(l_1 < 2)$$

is satisfied by S if and only if every path starting from a state x, with $x_1 = 0$, only contains states y such that $y_1 < 2$. In other words, the formula means that when the level of the first component is zero, then it will be always less than two. The asynchronous state graph of Fig. 2.11 does not satisfy this property, because of the presence of the path $(0, 0) \rightarrow (1, 0) \rightarrow (2, 0)$.

To summarize, the interest of CTL is twofold. Firstly, it allows to express, in a simple way, rather complex dynamical properties on undeterministic transition systems. Secondly, every CTL formula ϕ can be checked on a transition system S in a polynomial time with respect to the size of ϕ and S. (More precisely, the complexity of the verification is in $\mathcal{O}(|\phi| \cdot |S|)$ where $|\phi|$ is the number of symbols in ϕ seen as a string, and $|S|$ is the sum of the number of vertices and the number of transitions of the state graph S.) Notice though, as discussed in Sect. 2.3.2.4, that the number of state graphs grows exponentially with the number of components.

2.3.3 *Success Story: Pseudomonas aeruginosa and Cystic Fibrosis*

The bacteria *Pseudomonas aeruginosa* [25] are commonly present in the environment and secrete mucus only in lungs affected by cystic fibrosis. As it increases the respiratory deficiency of the patient, it is the major cause of mortality. Bacteria isolated from cystic fibrosis lungs continue to grow in laboratory as mucous colonies for numerous generations (mucoid phenotype). A majority of these bacteria present a mutation. Does it mean that the mutation is the cause of the passage to the mucoid state?

A majority of biologists tend to follow this hypothesis. However, the regulatory network that controls the mucus production has been elucidated (Fig. 2.13a) and the regulatory graph contains two feedback circuits among which one is a positive one (Fig. 2.13b). This positive circuit makes possible a dynamic with two attractors that would allow, from a biological point of view, an epigenetic change (stable change of phenotype without mutation) from the non-mucoid state to the mucoid one.

From a biological point of view, it is very important to determine whether the mucoidy can be induced by an epigenetic phenomenon or not. In such a case, the elimination of the anti-AlgU gene (*via* a mutation) could be favored later on because an inhibitor complex is produced, which is toxic for the bacteria.

From a modeling point of view, and because the mathematical model of mucus production system is not yet determined, this question becomes: Can we exhibit, from the interaction graph of 2.13, a dynamical model (an asynchronous state graph) presenting at least two attractors, one in which mucus is regularly produced and one in which mucus is not produced?

Assuming that AlgU activates the mucus production at its maximal level b_{AlgU}, to state that a model which *regularly produces mucus* is equivalent to the fact that

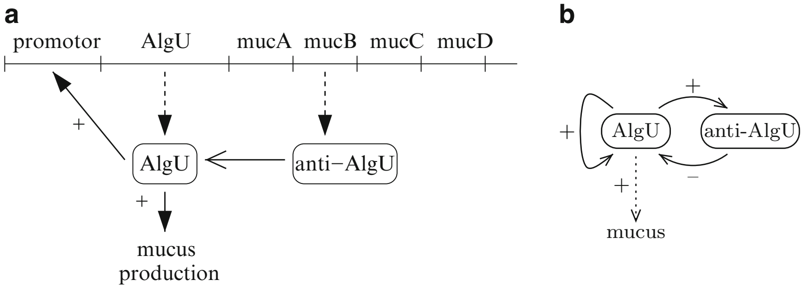


Fig. 2.13 Mucus production in *Pseudomonas aeruginosa* (**a**) The main regulatory genes (**b**) A possible interaction graph (identical to the one of Fig. 2.9)

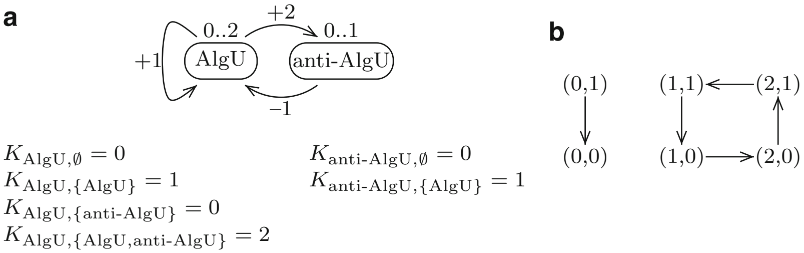

a
+2
0..2
0..1
+1
AlgU
anti-AlgU
−1
$K_{\text{AlgU},\emptyset} = 0$
$K_{\text{AlgU},\{\text{AlgU}\}} = 1$
$K_{\text{AlgU},\{\text{anti-AlgU}\}} = 0$
$K_{\text{AlgU},\{\text{AlgU},\text{anti-AlgU}\}} = 2$
$K_{\text{anti-AlgU},\emptyset} = 0$
$K_{\text{anti-AlgU},\{\text{AlgU}\}} = 1$
b
(0,1)
(1,1)
(2,1)
(0,0)
(1,0)
(2,0)

Fig. 2.14 Modeling the mucus production in *Pseudomonas aeruginosa* (**a**) The interaction graph of Fig. 2.13 together bounds, logical thresholds and logical parameters. (**b**) The resulting asynchronous state graph. It satisfies the formulas (2.28) and (2.29). It has two attractors: a cyclic attractor $\{(1,0),(2,0),(2,1),(1,1)\}$ in which mucus is regularly produced, and a stable state $(0,0)$ in which mucus is not produced

the concentration level of AlgU is repeatedly equal to b_{AlgU}. Thus this information can be expressed in CTL as:

$$(l_{\text{AlgU}} = b_{\text{AlgU}}) \Rightarrow \text{AX}(\text{AF}(l_{\text{AlgU}} = b_{\text{AlgU}})). \tag{2.28}$$

Moreover we know that the wild bacteria never produces mucus by themselves when starting from a basal state (second attractor):

$$(l_{\text{AlgU}} = 0) \Rightarrow \text{AG}(l_{\text{AlgU}} < b_{\text{AlgU}}). \tag{2.29}$$

Using SMBioNet, one shows that, among the 73 asynchronous state graphs that can be built using the logical method, from the interaction graph of Fig. 2.13, there are 17 asynchronous state graphs verifying the two previous formulas (one of them is display in Fig. 2.14). Consequently, because the set of remaining models is not empty, the epigenetic question receives a positive answer from the modeling standpoint. This epigenetic question has not only an academic interest because this prediction has been validated experimentally that could lead to new therapeutic strategies.

2.4 Outlook

A large amount of data on gene regulation is currently available. This created an upsurge of interest in modeling regulatory networks. Here, three usual approaches to modeling were presented: continuous differential systems (based on sigmoidal functions), piecewise affine systems (based on step functions), and discrete systems. On the one hand, continuous systems present a higher level of details and offer the ability to model dynamics. But on the other hand, discrete (or logical) systems are easier to analyze, need a smaller amount of data, can be deduced from qualitative observations, and thus allow modeling of larger systems. The choice then depends on the nature of input data and of the biological question under consideration. Furthermore, the well understood relationships between the continuous and discrete approaches presented here, allow one to follow a classical and simple strategy to model a new biological system: the discrete approach can be taken as a useful first step as long as the input data are qualitative, then more accurate descriptions can be achieved through continuous models based on the discrete ones when more precise input data are available.

Research on gene regulatory networks is rather active, and many research directions are relevant. From a computational point of view, it becomes crucial to develop techniques that allow the modeling of large systems, for instance using sensible model reductions or modular decompositions. Another direction consists in using experimental design approaches to select sets of experiments that are efficient to validate or to refute a model. From a biological point of view, a number of outstanding questions are open. The stochasticity and robustness of regulatory networks are not well understood. The dynamical influence of network architectures, and the evolutionary processes that produce them, are also far from being understood. Furthermore, models for the interplay between gene network and other processes, such as metabolism and cell signaling, have to be developed.

2.5 Online Resources

Several computer tools are available to help modeling and analyze genetic regulatory networks. A few examples are:

GNA (Genetic Network Analyzer)

http://www.genostar.com/en/genostar-software/gnasim.html

Modeling and simulation of GRN, using piecewise linear models.The user specifies the equations, the parameters (synthesis and degradation rates, thresholds), and inequality constraints between them.

GINsim (Gene Interaction Network simulation)

http://gin.univ-mrs.fr

Modeling and simulation of GRN, based on a discrete, logical formalism. The user may specify a model of a GRN in terms of asynchronous, multivalued logical functions.

SMBioNet (Selection of Models of Biological Networks)

http://www.i3s.unice.fr/~richard/smbionet

Modeling and model-checking analysis of GRN, based on a discrete, logical formalism (see Sect. 2.3.2.4).

The Biochemical Abstract Machine BIOCHAM

http://contraintes.inria.fr/BioCham

A modeling environment for systems biology, with features for inferring unknown model parameters from temporal logic constraints.

SBML (Systems Biology Markup Language)

http://sbml.org/Main_Page

A standard formalism for the construction and exchange of computer models of biological processes.

Acknowledgements It is a pleasure for GB, JPC and AR to thank the biologist Janine Guespin-Michel, who has actively participated to the definition of our formal logic methodology in such a way that our techniques from computer science and the SMBioNet software become truly useful for biologists. She has also been at the origin of the *Pseudomonas aeruginosa* hypothesis. The authors would also like to thank F. Cazals for his remarks and careful reading of the chapter.

List of Acronyms

DNA	Deoxyribose Nucleic Acid
mRNA	messenger Ribonucleic Acid
GRN	Gene Regulatory Networks
PWA	Piecewise affine

References

1. http://en.wikipedia.org/wiki/File:Ribosome_mRNA_translation_en.svg
2. http://http://en.wikipedia.org/wiki/File:Simple_transcription_elongation1.svg
3. U. Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman & Hall/CRC, Boca Raton, 2006.
4. J. Aracena. On the number of fixed points in regulatory boolean networks. *Bulletin of Mathematical Biology*, 70(5):1398–1409, 2008.
5. J. Aracena, J. Demongeot, and E. Goles. Positive and negative circuits in discrete neural networks. *IEEE Transactions of Neural Networks*, 15:77–83, 2004.
6. J. Barnat, L. Brim, I. Černá, S. Dražan, J. Fabrikozá, and D. Šafránek. On algorithmic analysis of transcriptional regulation by ltl model checking. *Theoretical Computer Science*, 2009.
7. G. Batt, M. Page, I. Cantone, G. Goessler, P. Monteiro, and H. de Jong. Efficient parameter search for qualitative models of regulatory networks using symbolic model checking. *Bioinformatics*, 26(18):i603–i610, 2010.
8. G. Bernot, J.-P. Comet, A. Richard, and J. Guespin. A fruitful application of formal methods to biological regulatory networks: Extending Thomas' asynchronous logical approach with temporal logic. *J. Theor. Biol.*, 229(3):339–347, 2004.

9. R. Casey, H. de Jong, and J.L. Gouzé. Piecewise-linear models of genetic regulatory networks: equilibria and their stability. *J. Math. Biol.*, 52:27–56, 2006.
10. M. Chaves and J.L. Gouzé. Exact control of genetic networks in a qualitative framework: the bistable switch example. *Automat.*, 47:1105–1112, 2011.
11. A. Cimatti, E. Clarke, E. Giunchiglia, F. Giunchiglia, M. Pistore, and M. Roven. NuSMV2: An Open Source Tool for Symbolic Model Checking. In *International Conference on Computer-Aided Verification (CAV 2002)*, 2002.
12. O. Cinquin and J. Demongeot. Roles of positive and negative feedback in biological systems. *C.R.Biol.*, 325(11):1085–1095, 2002.
13. F. Corblin, E. Fanchon, and L. Trilling. Applications of a formal approach to decipher discrete genetic networks. *BMC Bioinformatics*, 11(385), 2010.
14. F. Corblin, S. Tripodi, E. Fanchon, D. Ropers, and L. Trilling. A declarative constraint-based method for analyzing discrete genetic regulatory networks. *Biosystems*, 98(2):91–104, 2009.
15. F. Dardel and F. Képès. *Bioinformatics: genomics and post-genomics*. Wiley, Chichester, 2005.
16. H. De Jong. Modeling and simulation of genetic regulatory systems: a literature review. *Journal of computational biology*, 9(1):67–103, 2002.
17. H. de Jong, J.L. Gouzé, C. Hernandez, M. Page, T. Sari, and J. Geiselmann. Qualitative simulation of genetic regulatory networks using piecewise linear models. *Bull. Math. Biol.*, 66:301–340, 2004.
18. L. Edelstein-Keshet. *Mathematical models in Biology*. SIAM classics in applied mathematics, Philadelphia, 2005.
19. E.A. Emerson. *Handbook of theoretical computer science, Volume B : formal models and semantics*, chapter Temporal and modal logic, pages 995–1072. MIT Press, 1990.
20. S. Gama-Castro, H. Salgado, M. Peralta-Gil, A. Santos-Zavaleta, L. Muniz-Rascado, H. Solano-Lira, V. Jimenez-Jacinto, V. Weiss, J. S. Garcia-Sotelo, A. Lopez-Fuentes, L. Porron-Sotelo, S. Alquicira-Hernandez, A. Medina-Rivera, I. Martinez-Flores, K. Alquicira-Hernandez, R. Martinez-Adame, C. Bonavides-Martinez, J. Miranda-Rios, A. M. Huerta, A. Mendoza-Vargas, L. Collado-Torres, B. Taboada, L. Vega-Alvarado, M. Olvera, L. Olvera, R. Grande, E. Morett, and J. Collado-Vides. RegulonDB version 7.0: transcriptional regulation of Escherichia coli K-12 integrated within genetic sensory response units (Gensor Units). *Nucleic Acids Research*, 2010.
21. T.S. Gardner, C.R. Cantor, and J.J. Collins. Construction of a genetic toggle switch in *Escherichia coli*. *Nature*, 403:339–342, 2000.
22. L. Glass and S.A. Kauffman. The logical analysis of continuous, nonlinear biochemical control networks. *J. Theor. Biol.*, 39:103–129, 1973.
23. J.L. Gouzé. Positive and negative circuits in dynamical systems. *Journal of Biological Systems*, 6:11–15, 1998.
24. F. Grognard, J.-L. Gouzé, and H. de Jong. Piecewise-linear models of genetic regulatory networks: theory and example. In I. Queinnec, S. Tarbouriech, G. Garcia, and S. Niculescu, editors, *Biology and control theory: current challenges*, Lecture Notes in Control and Information Sciences (LNCIS) 357, pages 137–159. Springer-Verlag, 2007.
25. J. Guespin-Michel and M. Kaufman. Positive feedback circuits and adaptive regulations in bacteria. *Acta Biotheor.*, 49(4):207–18, 2001.
26. M. Kaufman, C. Soulé, and R. Thomas. A new necessary condition on interaction graphs for multistationarity. *Journal of Theoretical Biology*, 248:675–685, 2007.
27. H.K. Khalil. *Nonlinear systems*. Prentice Hall, New Jersey, 2002.
28. E. Klipp, R. Herwig, A. Howald, C. Wierling, and H. Lehrach. *Systems Biology in Practice*. Wiley-VCH, Weinheim, 2005.
29. E. Plahte, T. Mestl, and S.W. Omholt. Feedback loops, stability and multistationarity in dynamical systems. *Journal of Biological Systems*, 3:569–577, 1995.
30. E. Remy, P. Ruet, and D. Thieffry. Graphic requirement for multistability and attractive cycles in a boolean dynamical framework. *Advances in Applied Mathematics*, 41(3):335–350, 2008.
31. A. Richard. Positive circuits and maximal number of fixed points in discrete dynamical systems. *Discrete Applied Mathematics*, 157(15):3281–3288, 2009.

32. A. Richard. Negative circuits and sustained oscillations in asynchronous automata networks. *Advances in Applied Mathematics*, 44(4):378–392, 2010.
33. A. Richard and J.-P. Comet. Necessary conditions for multistationarity in discrete dynamical systems. *Discrete Applied Mathematics*, 155(18):2403–2413, 2007.
34. D. Ropers, H. de Jong, M. Page, D. Schneider, and J. Geiselmann. Qualitative simulation of the carbon starvation response in *Escherichia coli*. *Biosystems*, 84(2):124–152, 2006.
35. E.H. Snoussi. Qualitative dynamics of a piecewise-linear differential equations : a discrete mapping approach. *Dynamics and stability of Systems*, 4:189–207, 1989.
36. E.H. Snoussi. Necessary conditions for multistationarity and stable periodicity. *Journal of Biological Systems*, 6:3–9, 1998.
37. C. Soulé. Graphical requirements for multistationarity. *ComPlexUs*, 1:123–133, 2003.
38. C. Soulé. Mathematical approaches to differentiation and gene regulation. *C.R. Biologies*, 329:13–20, 2006.
39. R. Thomas. On the relation between the logical structure of systems and their ability to generate multiple steady states and sustained oscillations. In *Series in Synergetics*, volume 9, pages 180–193. Springer, 1981.
40. G. Yagil and E. Yagil. On the relation between effector concentration and the rate of induced enzyme synthesis. *Biophys. J.*, 11:11–27, 1971.