Incorporating Time Delays into the Logical Analysis of Gene Regulatory Networks*

Heike Siebert and Alexander Bockmayr

DFG Research Center MATHEON,
Freie Universität Berlin, Arnimallee 3, D-14195 Berlin, Germany
siebert@mi.fu-berlin.de, bockmayr@mi.fu-berlin.de

Abstract. Based on the logical description of gene regulatory networks developed by R. Thomas, we introduce an enhanced modelling approach that uses timed automata. It yields a refined qualitative description of the dynamics of the system incorporating information not only on ratios of kinetic constants related to synthesis and decay, but also on the time delays occurring in the operations of the system. We demonstrate the potential of our approach by analysing an illustrative gene regulatory network of bacteriophage λ.

1 Introduction

When modelling a gene regulatory network one has basically two options. Traditionally, such a system is modelled with differential equations. The equations used, however, are mostly non-linear and thus they cannot be solved analytically. Furthermore, the available experimental data is often of qualitative character and does not allow a precise evaluation of quantitative parameters for the differential model. This eventually led to the development of qualitative modelling approaches. R. Thomas introduced a logical formalism in the 1970s, which, over the years, has been further developed and successfully applied to different biological problems (see [7], [8] and references therein). The only information on a concentration of gene products required in this formalism is whether or not it is above a threshold relevant for some interaction in the network. Furthermore, parameters holding information about the ratio of production and spontaneous decay rates of the gene products are used. The values of these parameters determine the dynamical behaviour of the system which is represented as a state transition graph. Moreover, Thomas realized that a realistic model should not be based on the assumption that the time delay from the start of the synthesis of a given product until the point where the concentration reaches a threshold is the same for all the genes in the network. Neither will the time delays associated to synthesis and those associated to decay be the same. Therefore, he uses an asynchronous description of the dynamics of the system, i. e., a state in the state transition graph differs from its predecessor in one component only.

* Supported by the DFG Research Center MATHEON "Mathematics for key technologies".

In order to refine the model, we would like to incorporate information about the values of the time delays. Since precise data about the time delays is not available (in biological systems the delays will not even have an exactly determined value), the information is given in the form of inequalities that pose constraints on the time delays. So we need to keep track of time while the system evolves. A theoretical framework providing us with the necessary premises is the theory of timed automata. Each gene is equipped with a clock which is used to evaluate the conditions posed on the time delays of that particular gene during the evolution of the system. The resulting transition system is in general nondeterministic, but the additional information inserted allows for a refined view of the dynamics. Conclusions about stability of dynamical behaviour and restriction to certain behaviour in comparison to the predictions of the Thomas model become possible. Moreover, the possibility of synchronous update is not excluded under certain conditions.

In the first part of the paper we give a thorough mathematical description of the Thomas formalism in Sect. 2 and of the modelling approach using timed automata in Sect. 3 and 4. In Sect. 5, we show that, by using our approach, it is possible to obtain the state transition graph of the Thomas model. Also, we outline further possibilities our model offers. To illustrate the theoretical considerations, we analyse a simple regulatory network of bacteriophage λ in Sect. 6. In addition to the mere formal analysis, we have implemented the network using the verification tool UPPAAL. In the last section, we discuss the mathematical and biological perspectives of our approach.

2 Generalised Logical Formalism of Thomas

In this section we give a formal definition of a gene regulatory network in the sense of the modelling approach of R. Thomas (see for example [7] and [8]). We use mainly the formalisation introduced in [4].

Definition 1. *Let $n \in \mathbb{N}$. An* interaction graph *(or biological regulatory graph) IG is a labelled directed graph with vertex set $V := \{\alpha_1, \ldots, \alpha_n\}$ and edge set E. Each edge $\alpha_j \to \alpha_i$ is labelled with a sign $\varepsilon_{ij} \in \{+, -\}$ and an integer $b_{ij} \in \{1, \ldots, d_j\}$, where d_j denotes the out-degree of α_j. Furthermore, we assume that $\{b_{ij}\,;\, \exists\ \alpha_j \to \alpha_i\} = \{1, \ldots, p_j\}$ for all $j \in \{1, \ldots, n\}$ and $p_j \leq d_j$. We call $\{0, \ldots, p_j\}$ the* range *of α_j. For each $i \in \{1, \ldots, n\}$ we denote by $Pred(\alpha_i)$ the set of vertices α_j such that $\alpha_j \to \alpha_i$ is an edge in E.*

The vertices of this graph represent the genes of the gene regulatory network, the range of a vertex the different expression levels of the corresponding gene product of importance to the behaviour of the network. An edge $\alpha_j \to \alpha_i$ signifies that the gene product of α_j influences the gene α_i in a positive or negative way depending on ε_{ij} provided that the expression level of α_j is equal or above b_{ij}. Note that the values b_{ij} do not have to be pairwise distinct.

In order to describe the behaviour of a gene regulatory network we need a formal framework to capture its dynamics.

Definition 2. *Let IG be an interaction graph. A* state *of the system described by IG is a tuple $s \in S^n := \prod_{i=1}^{n}\{0,\ldots,p_i\}$. The set of* resources $R_i(s)$ *of α_i in state s is the set*

$$\{\alpha_j \in Pred(\alpha_i)\,;\,(\varepsilon_{ij} = + \wedge s_j \geq b_{ij}) \vee (\varepsilon_{ij} = - \wedge s_j < b_{ij})\}.$$

Finally, we define the set of parameters

$$K(IG) := \{K_{\alpha_i,\omega} \in \{0,\ldots,p_i\}\,;\, i \in \{1,\ldots,n\},\, \omega \subset Pred(\alpha_i)\}.$$

We call the pair $(IG, K(IG))$ a gene regulatory network.

The set of resources $R_i(s)$ gives information about the presence of activators and the absence of inhibitors for some gene α_i in state s. The value of the parameter $K_{\alpha_i,R_i(s)}$ indicates how the expression level of the gene α_i will evolve. The product concentration will increase (decrease) if the parameter value is greater (smaller) than s_i. The expression level stays the same if both values are equal.

Thomas and Snoussi used this formalism to discretize a certain class of differential equation systems (see [5]). To reflect this, the following constraint has to be posed on the parameter values:

$$\omega \subset \omega' \Rightarrow K_{\alpha_i,\omega} \leq K_{\alpha_i,\omega'} \qquad (1)$$

for all $i \in \{1,\ldots,n\}$. The condition signifies that an effective activator or a non-effective inhibitor cannot induce the decrease of the expression level of α_i. In the following we will always assume that this constraint is valid in order to compare our modelling approach with the one used by Thomas. However, we will discuss in the last section of this paper the possibilities of generalisations of the model that do not require the constraint (1).

To conclude this section, we describe the dynamics of the gene regulatory network by means of a state transition graph.

Definition 3. *The* state transition graph STG_N *corresponding to a gene regulatory network N is a directed graph with vertex set S^n. There is an edge $s \rightarrow s'$ if there is $i \in \{1,\ldots,n\}$ such that $|s'_i - K_{\alpha_i,R_i(s)}| = |s_i - K_{\alpha_i,R_i(s)}| - 1$ and $s_j = s'_j$ for all $j \in \{1,\ldots,n\} \setminus \{i\}$.*

The above definition reflects the use of the asynchronous update rule, since a state differs from a successor state in one component only. If s is a state such that an evolution in more than one component is indicated, then there will be more than one successor of s. Note that s is a steady state if s has no outgoing edge.

A gene regulatory network comprising two genes connected with a positive and a negative edge and the resulting state transition graph are given in Fig. 1. We use this simple example to illustrate the construction of the timed automaton representing the network in Sect. 4.

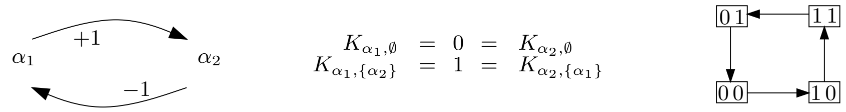


Fig. 1. Interaction graph, parameters and state transition graph of a simple gene regulatory system.

3 Timed Automata

In this section we formally define a timed automaton. We mainly use the definitions and notations given in [1].

To introduce the concept of time in our system, we consider a set $C := \{c_1, \ldots, c_n\}$ of real variables which behave according to the differential equations $\dot{c}_i = 1$. These variables are called *clocks*. They progress synchronously and can be reset to zero under certain conditions. We define the set $\Phi(C)$ of *clock constraints* φ by the grammar

$$\varphi := c \le q \mid c \ge q \mid c < q \mid c > q \mid \varphi_1 \wedge \varphi_2 \,,$$

where $c \in C$ and q is a rational constant.

A *clock interpretation* is a function $u : C \to \mathbb{R}_{\ge 0}$ from the set of clocks to the non-negative reals. For $\delta \in \mathbb{R}_{\ge 0}$, we denote by $u + \delta$ the clock interpretation that maps each $c \in C$ to $u(c) + \delta$. For $Y \subset C$, we indicate by $u[Y := 0]$ the clock interpretation that maps $c \in Y$ to zero and agrees with u over $C \setminus Y$. A clock interpretation u satisfies a clock constraint φ if $\varphi(u) = true$. The set of all clock interpretations is denoted by $\mathbb{R}^C_{\ge 0}$.

Definition 4. *A* timed automaton *is a tuple $(L, L^0, \Sigma, C, I, E)$, where L is a finite set of* locations*, $L^0 \subset L$ is the set of* initial locations*, Σ is a finite set of* events *(or labels), C is a finite set of* clocks*, $I : L \to \Phi(C)$ is a mapping that labels each location with some clock constraint which is called the* invariant *of the location, and $E \subset L \times \Sigma \times \Phi(C) \times 2^C \times L$ is a set of* switches*.*

A timed automaton can be represented as a directed graph with vertex set L. The vertices are labelled with the corresponding invariants and are marked as initial locations if they belong to L^0. The edges of the graph correspond to the switches and are labelled with an event, a clock constraint called *guard*, which specifies when the switch is enabled, and a subset of C giving the clocks that are reset to zero with this switch. While switches are instantaneous, time may elapse in a location. To describe the dynamics of such an automaton formally, we use the notion of a transition system.

Definition 5. *Let A be a timed automaton. The (labelled)* transition system *T_A associated with A is a tuple $(Q, Q^0, \Gamma, \to)$, where Q is the set of states $(l, u) \in L \times \mathbb{R}^C_{\ge 0}$ such that u satisfies the invariant $I(l)$, Q^0 comprises the states $(l, u) \in Q$ where $l \in L^0$ and u ascribes the value zero to each clock, and $\Gamma := \Sigma \cup \mathbb{R}_{\ge 0}$. Moreover, $\to \subset Q \times \Gamma \times Q$ is defined as the set comprising*

- $(l, u) \xrightarrow{\delta} (l, u + \delta)$ *for* $\delta \in \mathbb{R}_{\geq 0}$ *such that for all* $0 \leq \delta' \leq \delta$ *the clock interpretation* $u + \delta'$ *satisfies the invariant* $I(l)$*, and*
- $(l, u) \xrightarrow{a} (l', u[R := 0])$ *for* $a \in \Sigma$ *such that there is a switch* (l, a, φ, R, l') *in* E*,* u *satisfies* φ*, and* $u[R := 0]$ *satisfies* $I(l')$*.*

The elements of $\rightarrow$ *are called* transitions*.*

The first kind of transition is a state change due to elapse of time, while the second one is due to a location-switch and is called discrete. Again we can visualise the object T_A as a directed graph with vertex set Q and edges corresponding to the transitions given by $\rightarrow$. We will use terminology from graph theory with respect to T_A. Note, that by definition the set of states may be infinite and that the transition system is in general nondeterministic, i.e., a state may have more than one successor. Moreover, it is possible that a state is the source for edges labelled with a real value as well as for edges labelled with events. However, although every discrete transition corresponds to a switch in A, there may be switches in A that do not lead to a transition in T_A. That is due to the additional conditions placed on the clock interpretations. Furthermore, we obtain a modified transition system by considering only the location vectors as states, dropping all transitions labelled with real values, but keeping every discrete transition of T_A. We call it the discrete (or symbolic) transition system of A.

4 Modelling with Timed Automata

In order to model a gene regulatory network as a timed automaton, we first introduce components that correspond to the genes of the network. They constitute the building blocks that compose the automaton, representing the network much in the same way n timed automata are integrated to a product automaton (see [1]).

In the remaining part of the paper, let $N = (IG, K(IG))$ be a gene regulatory network comprising the genes $\alpha_1, \ldots, \alpha_n$.

Constructing the components. Let $i \in \{1, \ldots, n\}$. We define the component $A_i := (L_i, L_i^0, \Sigma_i, C_i, I_i, E_i)$ corresponding to α_i according to the syntax of timed automata. In addition we will label the locations with a set of switch conditions.

Locations: We define L_i as the set comprising the elements α_i^k for $k \in \{0, \ldots, p_i\}$, α_i^{k+} for $k \in \{0, \ldots, p_i - 1\}$, and α_i^{k-} for $k \in \{1, \ldots, p_i\}$. A location α_i^k represents that gene α_i has expression level k and that there will be no change in the expression level in the given situation. We call those locations *regular*. If the superscript is $k+$ ($k-$), the expression level is k but the concentration of the gene product tends to increase (decrease). Those locations are called *intermediate*. We define $L_i^0 := \{\alpha_i^k \,;\, k \in \{0, \ldots, p_i\}\}$.

Events: The events in Σ_i correspond to the intermediate locations. We set $\Sigma_i := \{a_i^{k+}, a_i^{m-} \,;\, k \in \{0, \ldots, p_i - 1\}, m \in \{1, \ldots, p_i\}\}$. They will be used later on to identify certain discrete transitions starting in the intermediate locations.

Clocks: For each gene we use a single clock, so $C_i := \{c_i\}$.

Invariants: We define the mapping $I_i : L_i \to \Phi(C_i)$ as follows. Every regular location α_i^k is mapped to $c_i \geq 0$ (evaluating to *true*). For each intermediate location $\alpha_i^{k\varepsilon}$, $\varepsilon \in \{+,-\}$, we choose $t_i^{k\varepsilon} \in \mathbb{Q}_{\geq 0}$ and set $I_i(\alpha_i^{k\varepsilon}) = (c_i \leq t_i^{k\varepsilon})$. The value $t_i^{k\varepsilon}$ signifies the maximal time delay occurring before the expression level of α_i changes to $k+1$, if $\varepsilon = +$, or to $k-1$, if $\varepsilon = -$. During that time a change in the expression level of α_i may yet be averted, if the expression levels of the genes influencing α_i change.

Switches: To specify the guard conditions on the switches, we choose the variables concerning time $t_i^{(k,k+1)}, t_i^{(k+1,k)} \in \mathbb{Q}_{\geq 0}$ for all $k \in \{0, \ldots, p_i - 1\}$. There are two kinds of switches in E_i. For all $k \in \{0, \ldots, p_i - 1\}$, we have $(\alpha_i^{k+}, a_i^{k+}, \varphi_i^{k+}, \{c_i\}, \alpha_i^{k+1}) \in E_i$, where $\varphi_i^{k+} = (c_i \geq t_i^{(k,k+1)})$. Furthermore, for $k \in \{1, \ldots, p_i\}$, the switch $(\alpha_i^{k-}, a_i^{k-}, \varphi_i^{k-}, \{c_i\}, \alpha_i^{k-1})$ with $\varphi_i^{k-} = (c_i \geq t_i^{(k,k-1)})$ is in E_i. The given time constraints determine the minimal time delay before a change in the expression level can occur. Choosing the time constants associated with the guards smaller than those associated with the invariants of the corresponding intermediate location leads to indeterministic behaviour of the system in that location.

Switch conditions: To each location in L_i we assign some conditions which later will be used to define the switches of the timed automaton of the gene regulatory network. These conditions concern the locations of all components A_j, $j \in \{1, \ldots, n\}$. We interpret locations as integer values by using the function $\iota : \bigcup_{j \in \{1,\ldots,n\}} L_j \to \mathbb{N}_0$ that maps the locations α_j^k, α_j^{k+} and α_j^{k-} to k.

Let $k \in \{1, \ldots, p_i - 1\}$. We define the logical conditions Λ_i^k and $\overline{\Lambda}_i^k$ as follows. For every $\alpha_j \in Pred(\alpha_i)$ and l_j a location of A_j let

$$\lambda_i^{\alpha_j}(l_j) := \begin{cases} \iota(l_j) \geq b_{ij}\,, & \varepsilon_{ij} = + \\ \iota(l_j) < b_{ij}\,, & \varepsilon_{ij} = - \end{cases} \qquad \overline{\lambda}_i^{\alpha_j}(l_j) := \begin{cases} \iota(l_j) < b_{ij}\,, & \varepsilon_{ij} = + \\ \iota(l_j) \geq b_{ij}\,, & \varepsilon_{ij} = - \end{cases}.$$

Let $\omega_1, \ldots, \omega_{m_i^1}, \upsilon_1, \ldots, \upsilon_{m_i^2}$ be the subsets of $Pred(\alpha_i)$ such that the parameter inequalities $K_{\alpha_i,\omega_h} > k$ for all $h \in \{1, \ldots, m_i^1\}$ as well as $K_{\alpha_i,\upsilon_h} < k$ for all $h \in \{1, \ldots, m_i^2\}$ hold. Let $l \in \prod_{j \in \{1,\ldots,n\}} L_j$. Then we define $\lambda_i^{\omega_h}(l) := \bigwedge_{\alpha_j \in \omega_h} \lambda_i^{\alpha_j}(l_j)$ and $\overline{\lambda}_i^{\upsilon_h}(l) := \bigwedge_{\alpha_j \in Pred(\alpha_i) \setminus \upsilon_h} \overline{\lambda}_i^{\alpha_j}(l_j)$. Finally, we set

$$\Lambda_i^k(l) := \bigvee_{h \in \{1,\ldots,m_i^1\}} \lambda_i^{\omega_h} \qquad \text{and} \qquad \overline{\Lambda}_i^k(l) := \bigvee_{h \in \{1,\ldots,m_i^2\}} \overline{\lambda}_i^{\upsilon_h}\,.$$

We define Λ_i^0 and $\overline{\Lambda}_i^{p_i}$ accordingly.

Now, we assign all locations α_i^k, $k \in \{1, \ldots, p_i - 1\}$ the conditions Λ_i^k and $\overline{\Lambda}_i^k$. The location α_i^0 ($\alpha_i^{p_i}$) is labelled with Λ_i^0 ($\overline{\Lambda}_i^{p_i}$) only. Furthermore, we associate with the location α_i^{k+} the condition $\Psi_i^{k+} := \neg \Lambda_i^k$ for all $k \in \{0, \ldots, p_i - 1\}$, and allot to the location α_i^{k-} the condition $\Psi_i^{k-} := \neg \overline{\Lambda}_i^k$ for all $k \in \{1, \ldots, p_i\}$.

The conditions defined above correspond to the the set of resources used in the formalism of Thomas and thus play a key role in the dynamics of the system. If the condition Λ_i^k is met, the gene α_i will start producing its product at a higher rate. This is represented by a transition to the location α_i^{k+} (see the definition of the switches of the timed automaton A defined below). However, it is possible that some change in the expression levels of genes influencing α_i occurs while α_i has not yet reached the location α_i^{k+1}. If those changes are such that the condition Ψ_i^{k+} is satisfied, then the premises for α_i to reach the expression level $k+1$ are no longer given, and it will return to the location α_i^k (again see the definition of the switches of A below). The conditions $\overline{\Lambda}_i^k$ and Ψ_i^{k-} are used similarly for the decrease of the expression level.

Note, that whenever $\omega_{h_1} \subset \omega_{h_2}$ for sets ω_h defined as above, condition $\lambda_i^{\omega_{h_2}}$ can be deleted from the expression Λ_i^k due to the constraints (1) on the parameter values. A corresponding statement holds for the sets υ_h.

Formally, the components defined above are timed automata. However, it does not make sense to evaluate their behaviour isolated from each other. This becomes apparent when looking at the graph representation. Most locations in the automaton A_i are not connected by edges. Every path in the graph contains at most one edge. Figure 2 illustrates this observation. It shows the components A_1 and A_2 corresponding to the genes α_1 and α_2 in Fig. 1. Each component comprises the regular locations α_i^0 and α_i^1 and the intermediate locations α_i^{0+} and α_i^{1-}, represented as circles in the graph. The first line below the location identifier in a circle is the corresponding invariant, the second line shows the corresponding switch condition. Since both genes have only two expression levels, each location is only labelled with one switch condition. For example, we have $2^{Pred(\alpha_1)} = \{\emptyset, \{\alpha_2\}\}$, $\lambda_1^{\alpha_2}(l_2) = (\iota(l_2) < 1)$ and $\overline{\lambda}_1^{\alpha_2}(l_2) = (\iota(l_2) \geq 1)$. Since $K_{\alpha_1,\{\alpha_2\}} = 1$, we have $\lambda_1^{\{\alpha_2\}} = (\iota(l_2) < 1)$ and $\Lambda_1^0(l) = (\iota(l_2) < 1)$. Furthermore, $K_{\alpha_1,\emptyset} = 0$ and thus $\overline{\lambda}_1^{\emptyset}(l) = (\iota(l_2) \geq 1) = \overline{\Lambda}_1^1$. The switches are represented as directed edges from the first to the last component of the switch. They are labelled with the guard, the event, and the set of clocks that are to be reset.

Modelling the network. In this paragraph, we construct the timed automaton $A_N := (L, L^0, \Sigma, C, I, E)$ representing the network N by means of the components $A_1, \ldots, A_n$ in the following way. We define $L := \prod_{i \in \{1,\ldots,n\}} L_i$, $L^0 := \prod_{i \in \{1,\ldots,n\}} L_i^0$ and $\Sigma := \{a\} \cup \bigcup_{i \in \{1,\ldots,n\}} \Sigma_i$. Here a will signify a general event, which is used to indicate that the switch is defined by means of the switch conditions of the components A_i (see below). A location in L is called *regular*, if all of its components are regular, and *intermediate* otherwise. Furthermore, we define the set of clocks $C := \bigcup_{i \in \{1,\ldots,n\}} C_i$ and $I : L \to \Phi(C)$, $(l_1, \ldots, l_n) \mapsto (I_1(l_1) \wedge \cdots \wedge I_n(l_n))$. The set of switches $E \subset L \times \Sigma \times \Phi(C) \times 2^C \times L$ is comprised of the following elements:

- For every $i \in \{1, \ldots, n\}$ and every switch $(l_i, a_i, \varphi_i, R_i, l_i') \in E_i$ the tuple $(h, a_i, \varphi_i, R_i, h')$, with $h, h' \in L$, $h_j = h_j'$ for all $j \neq i$, $h_i = l_i$ and $h_i' = l_i'$, is a switch in E.
- Let $(l, a, \varphi, R, l') \in L \times \Sigma \times \Phi(C) \times 2^C \times L$ with $\varphi := \mathit{true}$. Let J be the largest subset of $\{1, \ldots, n\}$ such that for each l_j, $j \in J$, one of the switch conditions

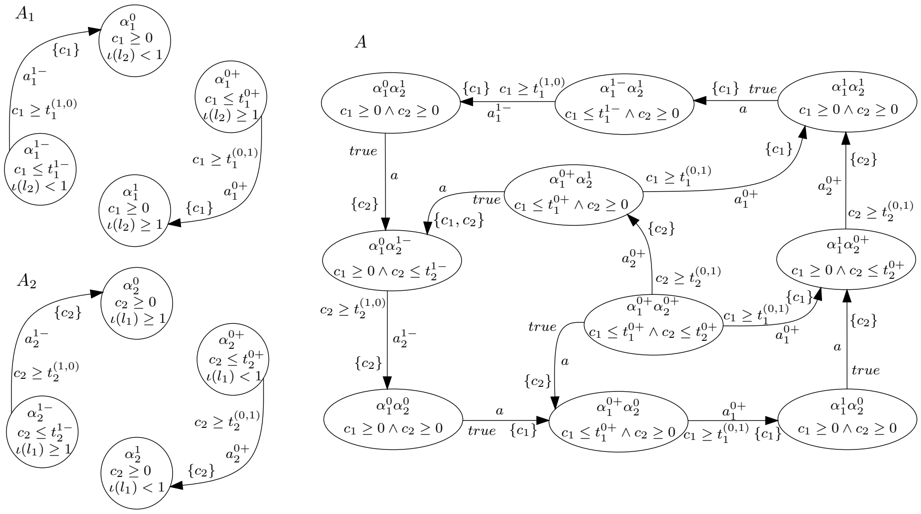


Fig. 2. On the left, components A_1 and A_2 representing the genes α_1 and α_2 in Figure 1. On the right, a section of the timed automaton A constructed from A_1 and A_2.

associated with l_j is true. Assume R comprises the clocks c_j, $j \in J$. Let $l_i = l'_i$ for all $i \notin J$, and let, for all $j \in J$,

$$l'_j = \begin{cases} \alpha_j^{k-}\ , & l_j = \alpha_j^k \quad \text{for some } k \text{ and } \overline{\Lambda}_j^k(l) = true \\ \alpha_j^{k+}\ , & l_j = \alpha_j^k \quad \text{for some } k \text{ and } \Lambda_j^k(l) = true \\ \alpha_j^k\ , & l_j = \alpha_j^{k+} \quad \text{for some } k \text{ and } \Psi_j^{k+}(l) = true \\ \alpha_j^k\ , & l_j = \alpha_j^{k-} \quad \text{for some } k \text{ and } \Psi_j^{k-}(l) = true \end{cases} \tag{2}$$

Then (l, a, φ, R, l') is a switch in E.

Although the formal definition of the switches looks quite complicated, the actual meaning is straightforward. A location change occurs when the current state of locations allows for a change. The switch conditions Λ_j^k, $\overline{\Lambda}_j^k$, Ψ_j^{k+} and Ψ_j^{k-} carry the information which conditions, depending on the current location of A, the expression levels of the genes influencing α_j have to satisfy in order to induce a change in the expression level of α_j (see the remarks to the switch conditions of the components A_i). Furthermore, changes in the expression level of a gene happen gradually. That is, for every two locations l, l' connected by a switch we have $|\iota(l_i) - \iota(l'_i)| \leq 1$ for all $i \in \{1, \ldots, n\}$. The event a is used to identify the switches that include the checking of the switch conditions of some location.

The timed automaton A representing the gene regulatory network in Fig. 1 is partially presented in Fig. 2. The figure includes all regular locations of A as well as all locations that are the target of an edge (representing a switch) starting in a regular location. Moreover, we chose two locations that render interesting switches. All switches of A starting in a location displayed in Fig. 2 are presented in the figure. We show the construction of the switches exemplarily for the location $(\alpha_1^{0+}, \alpha_2^1)$. First, we note that $(\alpha_1^{0+}, a_1^{0+}, (c_1 \geq t_1^{(0,1)}), \{c_1\}, \alpha_1^1)$ is a switch in A_1. Thus $((\alpha_1^{0+}, \alpha_2^1), a_1^{0+}, (c_1 \geq t_1^{(0,1)}), \{c_1\}, (\alpha_1^1, \alpha_2^1))$ is a switch in A. Now we check the switch conditions in $l = (\alpha_1^{0+}, \alpha_2^1)$. The condition $\Psi_1^{0+}(l) = (\iota(l_2) \geq 1)$ is true in l as is the condition $\overline{\Lambda}_2^1(l) = (\iota(l_1) < 1)$. Thus, $J = \{1, 2\}$. We obtain the target location l' of the switch according to (2). Since $\Psi_1^{0+}(l)$ is true and $l_1 = \alpha_1^{0+}$, we get $l'_1 = \alpha_1^0$, and since $\overline{\Lambda}_2^1(l)$ is true and $l_2 = \alpha_2^1$, we get $l'_2 = \alpha_2^{1-}$. The guard condition for the switch is *true*, it is labelled with a and both clocks c_1 and c_2 are reset.

The associated transition system. Let $T_A = (Q, Q^0, \Gamma, \rightarrow)$ be the transition system associated with A. Note that the above definition of the first kind of switch in E reflects the use of the asynchronous update of the expression levels in the transition system. More precisely, although more than one component of the discrete state may change in one step (via switches labelled with a), a change in the expression level will only occur in one component at most. We will refine the system in one aspect, which leads to a smaller set of possible transitions. Whenever $(l, u) \in Q$ is a state such that there is some transition $(l, u) \xrightarrow{a} (l', v)$ for some state $(l', v) \in Q$, then we delete every transition of the form $(l, u) \xrightarrow{\delta} (l, u + \delta)$ regardless of the value of δ. We call a an *urgent* event. That is to say, whenever some transition is labelled with the urgent event a, it is not possible for time to elapse further in location l. However, there may be further discrete transitions starting in (l, u), which allows for synchronous (in the temporal sense) update (see the example in Sect. 6). If we want to avoid that, we delete all other transitions starting in (l, u), and call a an *overriding* event. In the following we consider the more general case that a is urgent.

Furthermore, note that a transition labelled with a never leads to a change in the expression levels of the genes, and that J in the definition of the second kind of switch is chosen maximal. Thus, if a path in T_A starts in a regular location and its first transition is labelled with a, then the second transition in the path will not be labelled with a.

Here, some discrete state $l \in L$ is called a steady state if T_A does not contain a discrete transition starting in (l, u), for all clock interpretations u.

To analyse the dynamics of the gene regulatory network we consider the paths in T_A that start in some initial state in Q^0. Questions of interest are for example if a steady state is reachable from a given initial location via some path in T_A. We will discuss the analysis of T_A in a later section.

5 Comparison of the Models

In this section, we aim to show that on the one hand the information inherent in the state transition graph as defined in Definition 3 can also be obtained from the transition system of a suitable automaton. On the other hand, the modelling approach via timed automata offers possibilities to incorporate information about gene regulatory networks that cannot be included in the Thomas model, and thus leads to a refined view on the dynamics of the system.

Let S_N be the state transition graph corresponding to N and A the timed automaton derived from N, as explained in the preceding section. We set all time variables needed to define A to zero, that is $t_i^{k\varepsilon}, t_i^{(k,k+1)}, t_i^{(k+1,k)} = 0$ for all $i \in \{1, \ldots, n\}$, $\varepsilon \in \{+, -\}$. Thus every guard condition evaluates to *true* and time does not elapse in the intermediate locations.

Now, we derive a graph G from T_A as follows. First we identify locations of A_i which represent the same expression level, i.e., for $k \in \{1, \ldots, p_i - 1\}$ we define $v_k^{\alpha_i} := \{\alpha_i^k, \alpha_i^{k+}, \alpha_i^{k-}\}$, $v_0^{\alpha_i} := \{\alpha_i^0, \alpha_i^{0+}\}$ and $v_{p_i}^{\alpha_i} := \{\alpha_i^{p_i}, \alpha_i^{p_i-}\}$. Let $V^{\alpha_i} := \{v_k^{\alpha_i}\ ;\ k \in \{0, \ldots, p_i\}\}$ and $V := \prod_{i \in \{1,\ldots,n\}} V^{\alpha_i}$ be the vertex set of G. Furthermore, there is an edge $v \to w$, if $v \neq w$ and if there is a path in T_A from some state (l, u), such that l is regular, to a state (l', u') satisfying $l'_i \in w_i$ for all i, such that every discrete state on the path other than l' is an element of $\prod_{i \in \{1,\ldots,n\}} v_i$. The condition to start in a regular state l makes sure that the first discrete transition occurring is labelled with a. This excludes the possibility of a change of expression level that does not correspond to the parameter values. We can drop the condition, if we declare a an overriding event.

Theorem 1. *The graphs S_N and G are isomorphic.*

Proof. We define $f : S^n \to V$, $(s_1, \ldots, s_n) \mapsto (v_{s_1}^{\alpha_1}, \ldots, v_{s_n}^{\alpha_n})$. Then it is easy to see that f is a bijection.

Let $s \to s'$ be an edge in S_N. We have to show that $f(s) \to f(s')$ is an edge in G. Set $v := f(s)$ and $w := f(s')$.

According to the definition of the state transition graph, there is a $j \in \{1, \ldots, n\}$ such that $|s'_j - K_{\alpha_j, R_j(s)}| = |s_j - K_{\alpha_j, R_j(s)}| - 1$ and $s_i = s'_i$ for all $i \in \{1, \ldots, n\} \setminus \{j\}$. Thus, $v_i = w_i$ for all $i \neq j$, and $v_j \neq w_j$.

First we consider the case that $s_j < K_{\alpha_j, R_j(s)}$. It follows that $s_j \neq p_j$, and thus $\alpha_j^{s_j}, \alpha_j^{s_j+} \in v_j$, and $s'_j = s_j + 1$. We choose $l \in L$ such that $l_i = \alpha_i^{s_i}$ for all $i \in \{1, \ldots, n\}$, thus $l \in \prod_{i \in \{1,\ldots,n\}} v_i$ is regular. Furthermore, we choose the clock interpretation u that assigns each clock the value zero.

We have $R_j(s) \subset Pred(\alpha_j)$ and, by definition, we know that $\lambda_j^{R_j(s)}(l)$, and thus the switch condition $\Lambda_j^{s_j}(l)$, is true. It follows that there is a switch $(l, a, \varphi, R, \tilde{l}) \in E$ with $\varphi = true$, $\tilde{l}_j = \alpha_j^{s_j+}$ and $\tilde{l}_i \in v_i$ for all $i \neq j$. Thus we find a transition $(l, u) \xrightarrow{a} (\tilde{l}, u)$. Since time is not allowed to elapse in intermediate locations, and since no transition starting in $(\tilde{l}, u)$ is labelled with a according to the observations we made on page 9, every possible transition starting in $(\tilde{l}, u)$ will lead to a state that differs from $(\tilde{l}, u)$ in one component of the location vector only. Moreover, we have $(\alpha_j^{s_j+}, a_j^{s_j+}, \varphi_i^{s_j+}, \{c_j\}, \alpha_j^{s_j+1}) \in E_j$ and thus there is a transition $(\tilde{l}, u) \to (l', u)$ labelled with $a_j^{s_j+}$, with $l'_j = \alpha_j^{s_j+1} \in w_j$ and $l'_i = \tilde{l}_i \in v_i = w_i$ for $i \neq j$. It follows that $f(s) = v \to w = f(s')$ is an edge in G.

The case that $s_j < K_{\alpha_j, R_j(s)}$ and thus $s'_j = s_j - 1$ can be treated analogously.

Now let $v \to w$ be an edge in G. We set $s := f^{-1}(v)$ and $s' := f^{-1}(w)$. According to the definition there is a path $((l^1, u^1), \ldots, (l^m, u^m))$ in T_A such that l_1 is regular, $l_i^j \in v_i$ for all $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, m-1\}$ and $l_i^m \in w_i$ for all $i \in \{1, \ldots, n\}$. Since $l^1 \neq l^m$, there is some discrete transition in the path. Since every component of l^1 is regular, and thus the only discrete transition starting there is labelled by a, and since a is an urgent event, we can deduce that $(l^1, u^1) \to (l^2, u^2)$ is labelled by a. Then l^2 has at least one component which is an intermediate location. Let $J \subset \{1, \ldots, n\}$ be such that l_j^2 is an intermediate location for all $j \in J$, and l_i^2 is a regular location for all $i \notin J$. Then $l_i^2 = l_i^1$ for all $i \notin J$. Since time is not allowed to elapse in the intermediate locations, the transition from (l^2, u^2) to (l^3, u^3) has to be discrete. Moreover, we know that the transition is not labelled by a, since the first transition of the path is already labelled that way. It follows that there is $j \in J$ such that l_j^3 is regular, $l_j^3 \neq l_j^2$, and $l_i^3 = l_i^2$ for all $i \neq j$. Furthermore, the expression levels of gene α_j in location l_j^1 and in location l_j^3 differ. We can deduce that $l_j^3 \notin v_j$ and thus $l_j^3 \in w_j$, $m = 3$ and $w_i = v_i$ for all $i \neq j$. We have $l_j^1 = \alpha_j^{s_j}$ and $l_j^3 = \alpha_j^{s'_j}$ and $|s_j - s'_j| = 1$.

We first consider the case that $s'_j = s_j + 1$, i. e., $l_j^1 = \alpha_j^{s_j}$, $l_j^2 = \alpha_j^{s_j+}$ and $l_j^3 = \alpha_j^{s_j+1}$. Since there is a transition from (l^1, u^1) to (l^2, u^2), we can deduce that the switch condition $\Lambda_j^{s_j}(l^1)$ evaluates to *true*. Thus, there exists a subset ω of $Pred(\alpha_j)$ such that $K_{\alpha_j,\omega} > s_j$ and $\lambda_j^\omega(l^1)$ is true. By definition of the resources, we have $R_j(s) \supset \omega$ and thus $K_{\alpha_j,R_j(s)} \geq K_{\alpha_j,\omega} > s_j$ It follows that $|s_j - K_{\alpha_j,R_j(s)}| - 1 = K_{\alpha_j,R_j(s)} - s_j - 1 = K_{\alpha_j,r_j(s)} - s'_j = |s'_j - K_{\alpha_j,R_j(s)}|$ and thus that $s \to s'$ is an edge in the state transition graph S_N.

The case that $s'_j = s_j - 1$ can be treated analogously. □

For the above proof we used the most basic version of a timed automaton representing the network in question. Furthermore, we simplified the transition system T_A. Obviously, our modelling approach is designed to incorporate additional information about the biological system, such as information about the actual values of synthesis and decay rates. Thereby we obtain a more precise idea of the dynamics of the system. For example, we may be able to discard certain paths in the state transition graph that violate conditions involving the time delays (see the example presented in the next section). Furthermore, we can evaluate stability and feasibility of a certain behaviour, i. e., a certain path in the discrete transition system, in terms of clock interpretations that allow for that behaviour. The stricter the conditions the clock interpretations have to satisfy to permit a certain behaviour, the less allowance is made for fluctuations in the actual time delays of the genes involved.

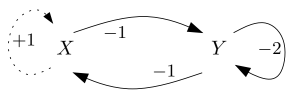
The intermediate locations give supplementary information about the behaviour of the genes. For instance, it is possible to distinguish between a gene keeping the same expression level because there is no change in the expression levels of the genes influencing it, and the same behaviour due to alternating opposed influences. In the first case, the gene stays in the regular location representing the expression level, in the second case it also traverses the corresponding intermediate locations.

Moreover, although this model uses asynchronous updates, it also allows for synchronous updates in the sense that two discrete transitions may occur at the same point in time. This may lead to paths in the transition system that are not incorporated in the state transition graph of the Thomas formalism.

To clarify the above considerations we give an illustrative example in the next section.

6 Bacteriophage λ

Temperate bacteriophages are viruses that can act in two different ways upon infection of a bacterium. If they display the lytic response, the virus multiplies, kills and lyses the cell. However, in some cases the viral DNA integrates into the bacterial chromosome, the viral genome becomes harmless for the so-called lysogenic bacterium. In [6], the formalism of Thomas is used to describe and analyse the genetic network associated with this behaviour. Figure 3 shows the simplified model they propose. We denote with X the gene cI and with Y the gene cro of the bacteriophage lambda. The choice of the thresholds and parameter values is based on experimental data. They render the loop starting in X ineffective with respect to the dynamics. Thus we will omit it in the modelling of the timed automaton. The resulting state transition graph shows two possible behaviours. The steady state in $(1, 0)$ can be related to the lysogenic, the cycle comprising the states $(0, 1)$ and $(0, 2)$ to the lytic behaviour.



$$\begin{array}{llll} K_{X,\emptyset} = 0 & K_{Y,\emptyset} = 0 \\ K_{X,\{X\}} = 0 & K_{Y,\{X\}} = 1 \\ K_{X,\{Y\}} = 1 & K_{Y,\{Y\}} = 0 \\ K_{X,\{X,Y\}} = 1 & K_{Y,\{X,Y\}} = 2 \end{array}$$

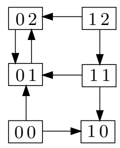


Fig. 3. Model of a network of bacteriophage λ in the Thomas formalism.

Now let us analyse that network modelled as a timed automaton. The component corresponding to X is of the same form as A_1 in Fig. 2. Since Y influences X as well as itself, the corresponding component is slightly more complex. Both components are shown in Fig. 4. Furthermore, the figure displays graphs, which are condensed versions of the different transition systems derived from the timed automaton combining X and Y. With the exception of graph (c), the vertices of the graphs represent the expression levels of the genes, which correspond to the integer value of the superscript of the location. For instance, the states (X^0, Y^{1-}) and (X^{0+}, Y^1) are both represented by $(0, 1)$. We analyse the dynamics of the system starting only from regular states. Thus, edges as well as paths in the graphs from some vertex $(j_1\, j_2)$ to a vertex $(i_1\, i_2)$ signify that the system can evolve from (X^{j_1}, Y^{j_2}) to a state where X and Y have expression level i_1 and

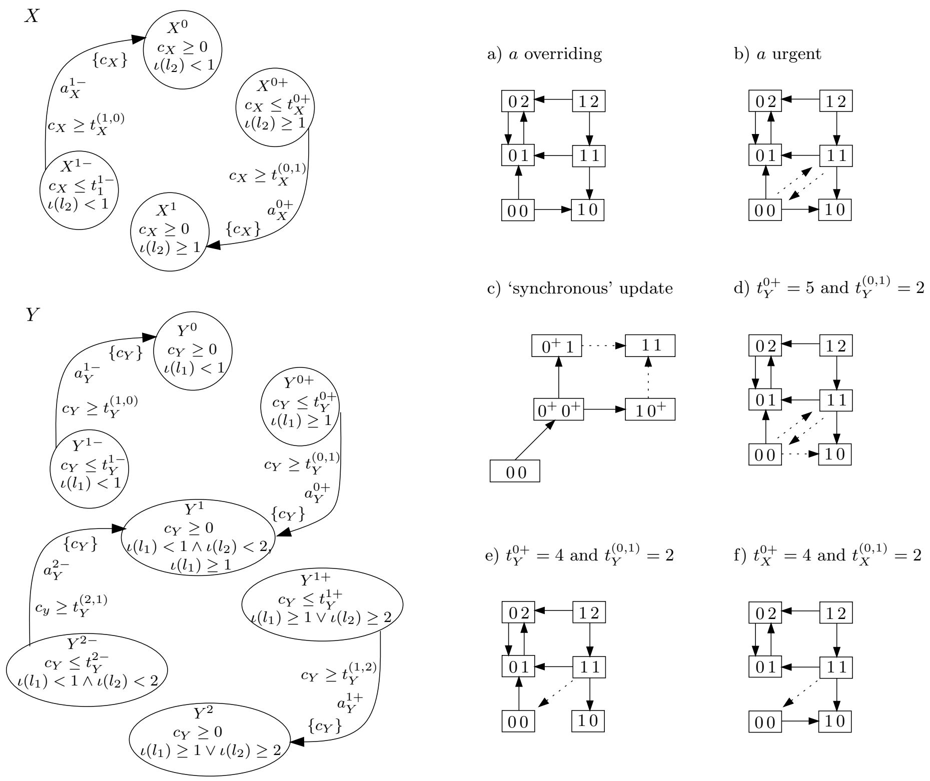


Fig. 4. The components X and Y representing the corresponding genes of the network in Figure 3. On the right, graphs representing the dynamical behaviour of the system derived from the transition systems resulting from different specifications of the model. Unless otherwise stated a is an urgent event and we set $t_Z^{k+} = t_Z^{k-} = 10$ and $t_Z^{(j,l)} = 5$ for all $Z \in \{X, Y\}$ and $k, j, l \in \{0, 1, 2\}$.

i_2 respectively. Thereby it traverses states with expression levels corresponding to the vertices in the path, provided there is an actual point in time in which the genes have those expression levels. Again graph (c) is an exception to this representation and its analysis will clarify the distinction made.

We specify our model by choosing values for the maximal and minimal time delays. Set $t_Z^{k+} = t_Z^{k-} = 10$ and $t_Z^{(j,l)} = 5$ for all $Z \in \{X, Y\}$ and $k, j, l \in \{0, 1, 2\}$. That is to say, the time delays for synthesis and decay are all in the same range regardless of the gene and the expression level. If we declare a to be an overriding event, we avoid the possibility that there is a path from $(0, 0)$ to $(1, 1)$ in the graph derived from the corresponding transition system. It is shown in Fig. 4 (a) and matches the state transition graph in Fig. 3. In (b), a is again an urgent event. We obtain two opposite edges between $(0, 0)$ and $(1, 1)$. However, there are very strict conditions posed on the time delays in order for the system to traverse those edges, which we drew dotted for that reason. To clarify the situation, we follow the path from $(0,0)$ to $(1,1)$ via the intermediate states shown in (c). A switch labelled with a leads to $(0^+,0^+)$. Assuming that X reaches the next expression level faster than Y after a time delay $5 \leq r_X \leq 10$, we reach $(1,0^+)$. In that situation two switches are enabled. One is labelled by a and leads to $(1,0)$. Since time is not allowed to pass, whenever the actual time r_Y that Y needs to reach the expression level 1 differs from r_X, that switch is taken. Only in the case that both time delays are exactly equal, the system will move via the switch labelled by a_Y^{0+} to $(1,1)$. Analogous considerations apply to the path via $(0^+,1)$. It follows that although the states $(0,0)$ and $(1,1)$ form a cycle in the graph, it is not plausible that the system will traverse that cycle. Once in the cycle, even the slightest perturbation of one of the time delays suffices for the system to leave the cycle. It is unstable.

These considerations apply not only to the edges representing synchronous update. In Fig. 4 (d) we change the values for t_Y^{0+} and $t_Y^{(0,1)}$ signifying that the synthesis of Y is usually faster than that of X. The system can reach the state $(1,0)$ only if Y needs the maximal and X the minimal time to change their expression level. So, usually we would expect the system to reach the cycle comprising $(0,1)$ and $(0,2)$, corresponding to the lytic behaviour of the bacteriophage. If we know that Y is always faster than X in reaching the expression level 1, we can eliminate the edge leading from $(0,0)$ to $(1,0)$, and that to $(1,1)$, altogether, as shown in (e). There is no clock interpretation that satisfies the posed conditions. If we reverse the situation of X and Y, we eliminate the edges from $(0,0)$ to $(0,1)$ and $(1,1)$ as shown in (f). In that case, the system starting in $(0,0)$ will always reach the steady state $(1,0)$ representing the lysogenic response of the bacteriophage. The incorporation of data concerning the time delays can thus lead to a substantial refinement of the analysis of the dynamical behaviour.

We have implemented the above system in UPPAAL[1], a tool for analysing systems modelled as networks of timed automata (see [3]). Since UPPAAL uses product automata in the sense of the definition in [1], we had to make some modifications in the modelling of the components. Primarily, we converted the switch conditions to actual switches, which synchronise via the input of an external component, that ensures the desired mechanisms of update of the system. With the model checking engine of UPPAAL we verified the dynamical properties of the different specifications of our model mentioned above.

7 Perspectives

In this paper, we introduced a discrete modelling approach that extends the established formalism of Thomas by incorporating constraints on the time delays occurring in the operations of biological systems. We addressed some of the advantages this kind of model offers, but naturally there is much room for future work. One of the most interesting possibilities the model provides is the evaluation of feasibility and stability of certain behaviours of the system by means of

[1] `http://www.uppaal.com`

the constraints posed on the time delays. We may find cycles in the transition system (implying homeostatic behaviour of the real system), the persistence of which requires that equalities for time delays are satisfied. It is highly unlikely that a biological system will sustain a behaviour which does not allow for the slightest perturbance in its temporal processes. A cycle persisting for a range of values for each time delay will be a lot more stable. The merit of such considerations was already mentioned by Thomas (see [8]). It calls for a thorough analysis with mathematical methods as well as testing with substantial biological examples.

Furthermore, it seems worthwhile to relax some of the conditions posed by the Thomas formalism. Dropping constraint (1) would allow for a combination of genes to have a different influence (inhibition, activation) on the target gene than each would have on its own. It also could be advantageous to allow a gene product to influence a target gene depending on its concentration. For instance, it may be activating in low but inhibiting in high concentrations. That translates to the formalism by allowing multiple edges in the interaction graph.

We would like to close with some remarks regarding the analysis of the dynamics of our model. The theory of timed automata provides powerful results concerning analysis and verification of the model by means of model checking techniques. For example, CTL and LTL model checking problems can be decided for timed automata (see [2]). However, we face the state explosion problem and moreover the task to phrase biological questions in terms suitable for model checking. A thorough study of problems and possibilities of applying model checking techniques to answer biologically relevant questions using the modelling framework given in this paper seems necessary and profitable.

References

1. R. Alur. Timed Automata. In *Proceedings of the 11th International Conference on Computer Aided Verification*, volume 1633 of *LNCS*, pages 8–22. Springer, 1999.
2. R. Alur, T. Henzinger, G. Lafferriere, and G. Pappas. Discrete abstractions of hybrid systems. In *Proceedings of the IEEE*, 2000.
3. J. Bengtsson and W. Yi. Timed Automata: Semantics, Algorithms and Tools. In *Lecture Notes on Concurrency and Petri Nets*, volume 3098 of *LNCS*, pages 87–124. Springer, 2004.
4. G. Bernot, J.-P. Comet, A. Richard, and J. Guespin. Application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic. *J. Theor. Biol.*, 229:339–347, 2004.
5. E. H. Snoussi. Logical identification of all steady states: the concept of feedback loop characteristic states. *Bull. Math. Biol.*, 55:973–991, 1993.
6. D. Thieffry and R. Thomas. Dynamical behaviour of biological regulatory networks - II. Immunity control in bacteriophage lambda. *Bull. Math. Biol.*, 57:277–297, 1995.
7. R. Thomas and R. d'Ari. *Biological Feedback*. CRC Press, 1990.
8. R. Thomas and M. Kaufman. Multistationarity, the basis of cell differentiation and memory. II. Logical analysis of regulatory networks in terms of feedback circuits. *Chaos*, 11:180–195, 2001.