Temporal Constraints in the Logical Analysis of Regulatory Networks ⋆

Heike Siebert and Alexander Bockmayr

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

Abstract

Starting from the logical description of gene regulatory networks developed by R. Thomas, we introduce an enhanced modelling approach based on timed automata. We obtain a refined qualitative description of the dynamical behaviour by exploiting not only information on ratios of kinetic parameters related to synthesis and decay, but also constraints on the time delays associated with the operations of the system. We develop a formal framework for handling such temporal constraints using timed automata, discuss the relationship with the original Thomas formalism, and 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 cannot be solved analytically. Furthermore, the available experimental data is often of qualitative character and does not allow a precise determination 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 [10], [11] 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

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

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 with synthesis and those associated with 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.

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 impose 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 imposed 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. Also, the possibility of synchronous update is not excluded under certain conditions. Furthermore, our modelling approach permits the modelling of context sensitive systems. That is, interactions between the network's components are allowed to be of different character, i. e., inhibiting or activating, depending on the state of the system. The resulting framework is substantially less restrictive than the classical Thomas formalism.

The organisation of this paper is as follows. We start in Sect. 2 with a mathematical presentation of the Thomas formalism, followed by a short review of the basic concepts of timed automata in Sect. 3. In Sect. 4 we develop our new modelling framework, which is the most important contribution of this paper. In Sect. 5, we show that using our approach, it is possible to obtain the state transition graph of the original Thomas model. To illustrate the theoretical considerations, we analyse in Sect. 6 two regulatory networks of bacteriophage λ. The corresponding models have been implemented using the verification tool UPPAAL. In the last section, we discuss the mathematical and biological perspectives of our approach.

This is the long version of a paper presented at CMSB'2006 [7].

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 [10] and [11]). We use mainly the formalism introduced in [4].

2.1 Structure and Dynamics

Definition 1 *An* interaction graph *(or biological regulatory graph) $\mathcal{I}$ is a labelled directed graph with vertex set $V := \{\alpha_1, \ldots, \alpha_n\}, n \geq 1$ 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 $\{b_{ij} \mid \exists\ \alpha_j \to \alpha_i\} = \{1, \ldots, p_j\}$ 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 regulatory network, the range of a vertex the different expression levels affecting the behaviour. 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 the sign ε_{ij} and 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 $\mathcal{I}$ be an interaction graph. A* state *of the system described by $\mathcal{I}$ is a tuple $s \in S^n := \{0, \ldots, p_1\} \times \cdots \times \{0, \ldots, p_n\}$. The set of* resources *$R_i(s)$ of α_i in state s is the set*

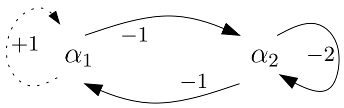
$$\{\alpha_j \in Pred(\alpha_i) \mid (\varepsilon_{ij} = + \wedge s_j \geq b_{ij}) \vee (\varepsilon_{ij} = - \wedge s_j < b_{ij})\}.$$

Finally, we define the set of (logical) parameters

$$K(\mathcal{I}) := \{K_{\alpha_i,\omega} \in \{0, \ldots, p_i\} \mid i \in \{1, \ldots, n\}, \omega \subseteq Pred(\alpha_i)\}.$$

We call the pair $(\mathcal{I}, K(\mathcal{I}))$ a gene regulatory network*.*

The set of resources $R_i(s)$ provides 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 gene α_i will evolve. The product concentration will increase (resp. decrease) if the parameter value is greater (resp. smaller) than s_i. The expression level stays the same if both values are equal.



$$
\begin{array}{llll}
K_{\alpha_1,\emptyset} = 0 & K_{\alpha_2,\emptyset} = 0 \\
K_{\alpha_1,\{\alpha_1\}} = 0 & K_{\alpha_2,\{\alpha_1\}} = 1 \\
K_{\alpha_1,\{\alpha_2\}} = 1 & K_{\alpha_2,\{\alpha_2\}} = 0 \\
K_{\alpha_1,\{\alpha_1,\alpha_2\}} = 1 & K_{\alpha_2,\{\alpha_1,\alpha_2\}} = 2
\end{array}
$$

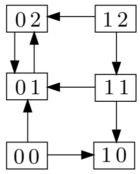


Fig. 1. Interaction graph, parameters and state transition graph of a gene regulatory network associated with bacteriophage λ.

Now, we describe the dynamics of the gene regulatory network by means of a state transition graph.

Definition 3 *The* state transition graph $\mathcal{S}_N$ *corresponding to a gene regulatory network* $N = (\mathcal{I}, K(\mathcal{I}))$ *is a directed graph with vertex set* S^n. *There is an edge* $s \to s'$ *if there is* $i \in \{1, \ldots, n\}$ *such that* $s'_i = s_i + sgn(K_{\alpha_i,R_i(s)} - s_i) \neq s_i$ *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.

Example 4 *Figure 1 shows the interaction as well as the state transition graph of a gene regulatory network comprising two genes connected by negative edges. Furthermore, each gene influences itself via a positive resp. negative loop. However, the given parameter values render the loop starting in* α_1 *ineffective with respect to the dynamics of the system. This example was given in [9] as a simplified model of a genetic network associated with the virus bacteriophage* λ. *We will give a closer look to its biological meaning and dynamical behaviour in Section 6.*

2.2 Parameter Constraints

Thomas and Snoussi used their formalism to discretise a certain class of differential equation systems (see [8]). To reflect this, the following constraint has to be imposed on the parameter values:

$$\omega \subseteq \omega' \Rightarrow K_{\alpha_i,\omega} \leq K_{\alpha_i,\omega'} \tag{1}$$

for all $i \in \{1, \ldots, n\}$. This condition signifies that an effective activator or a non-effective inhibitor cannot induce the decrease of the expression level of α_i. In biology, there are situations where this condition is not met. For instance, two substances can have activating properties when isolated from each other, but act inhibiting when combined. Thus it would be desirable to relax this restrictive condition.

The definition of the state transition graph representing the dynamics of a system does not have to be altered when dropping the parameter constraints (1). However, the interpretation of the corresponding interaction graph, and as a consequence that of the definition of the resources, becomes more difficult. Condition (1) ensures that the way two components influence each other, i. e., activation or inhibition, does not depend on the state of the system. For instance, if there is a positive edge from α_i to α_j, then the increase of the expression level of α_i can never lead to a decrease of the expression level of α_j. In this sense, condition (1) formalises the intuitive interpretation of the signs in the interaction graph. A source of a positive (resp. negative) interaction will never act as an inhibitor (resp. activator) of the corresponding target. The interaction graph is global, not depending on the current state of the system.

The formalism becomes much more flexible when allowing local interaction graphs that describe the interactions occurring between the components of the system in a given state. The local view allows for modelling systems including components acting as both activators or inhibitors, depending on the state of the other components (as mentioned above) or even depending on their own expression level (e. g. activating at low concentration levels, inhibiting at high levels). A notion of local interaction graphs has been proposed in [6].

The modelling approach we introduce in the following sections is also of local character and thus has the potential to capture the situations described in the preceding paragraph. Consequently, we do not require the constraints given in (1). We continue to use the formal definitions given in Section 2.1, in order to illustrate the development of our approach starting from the Thomas formalism. Furthermore, we will assume that condition (1) holds when comparing the dynamical behaviour resulting from both approaches.

3 Timed Automata

In this section we formally introduce timed automata. 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 that 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

$$\varphi ::= c \leq q \,|\, c \geq q \,|\, c < q \,|\, c > q \,|\, \varphi_1 \wedge \varphi_2 \,,$$

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

A *clock interpretation* is a function $u : C \to \mathbb{R}_{\geq 0}$ from the set of clocks to the non-negative reals. For $\delta \in \mathbb{R}_{\geq 0}$, we denote by $u + \delta$ the clock interpretation that maps each $c \in C$ to $u(c) + \delta$. For $Y \subseteq C$, 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 $\varphi(u) = \mathit{true}$. The set of all clock interpretations is denoted by $\mathbb{R}^{C}_{\geq 0}$.

Definition 5 *A* timed automaton *is a tuple $(L, L^0, \Sigma, C, I, E)$, where L is a finite set of* locations*, $L^0 \subseteq L$ 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 called the* invariant *of the location, and $E \subseteq L \times \Sigma \times \Phi(C) \times 2^C \times L$ 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* specifying when the switch is enabled, and a subset of C comprising 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 6 *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 $(l, u) \in L \times \mathbb{R}^{C}_{\geq 0}$ 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 $\Gamma := \Sigma \cup \mathbb{R}_{\geq 0}$. Moreover, $\to \subseteq Q \times \Gamma \times Q$ 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 $\to$ 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 $\to$. 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.

Finally, 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 this 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 dynamics much in the same way n timed automata are integrated to represent a product automaton (see [1]).

In the following, let $N = (\mathcal{I}, K(\mathcal{I}))$ be a gene regulatory network comprising the genes $\alpha_1, \ldots, \alpha_n$. We will illustrate each step of the modelling process with the example introduced in Fig. 1.

4.1 Constructing the Components

For $i \in \{1, \ldots, n\}$ 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 the location set 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 $k \in \{1, \ldots, p_i\}$. The location α_i^k represents a situation where gene α_i maintains expression level k. We call such a location *regular*. If the superscript is $k+$ (resp. $k-$), the expression level is k but the concentration of the gene product tends to increase (resp. decrease). Those locations are called *intermediate*. We define $L_i^0 := \{\alpha_i^k \mid k \in \{0, \ldots, p_i\}\}$.

Example 7 *As shown in Fig. 2, the component A_1 corresponding to gene α_1 of the example given in Fig. 1 contains two regular locations signifying its expression levels 0 and 1. Component A_2 has three regular locations, namely α_2^0, α_2^1 and α_2^2, since three distinct expression levels are associated with gene α_2.*

In order to measure time delays, we need to know when a gene starts the process of increasing or decreasing its expression level. We achieve this by introducing the intermediate locations. For example, the location α_1^{0+} represents the situation that gene α_1 is in the process of changing its expression

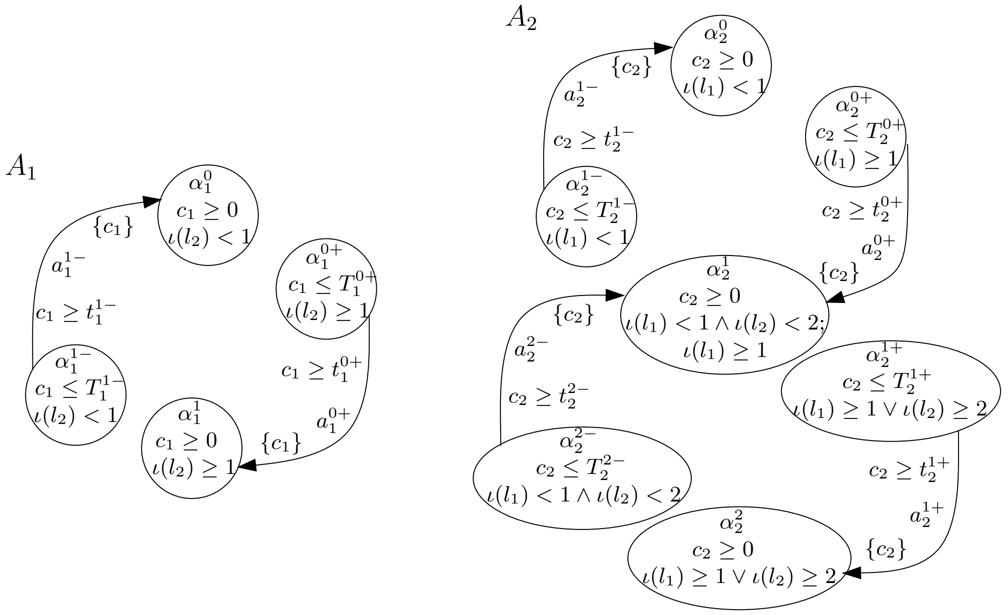


Fig. 2. Components A_1 and A_2 representing the genes α_1 and α_2 in Figure 1.

level from zero to one. Note that the expression level in this location is still 0, indicated by the number in the superscript.

Events: The events in Σ_i correspond to the intermediate locations. We set $\Sigma_i := \{a_i^{k+}, a_i^{m-} \mid k \in \{0, \dots, p_i - 1\}, m \in \{1, \dots, p_i\}\}$. These events will be used later on to identify certain discrete transitions starting in the intermediate locations, namely those that result in a change of expression level. For example, the event a_1^{0+} signifies that the expression level of gene α_1 changes from zero to one.

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

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*). That is, the question whether or not the system remains in a regular location does not depend on the clock values. Now, we make the first step in incorporating time delays. Since it is not realistic to assign an exact time delay to a biological process such as change of expression level, we rather use an interval bounded by a maximal and minimal time delay. 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 before the expression level of α_i changes to $k+1$, if $\varepsilon = +$, or to $k-1$, if $\varepsilon = -$.

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

Switch conditions: In a last step, we have to incorporate information of network interactions and parameters. We label each location with conditions concerning the expression levels of the interacting genes. If the conditions are met, a change in the location is indicated. Note that in general these conditions can only be evaluated in the network context, since information about the current location of all interacting genes is needed.

To formulate the switch conditions, we need to know how to obtain from a location the expression level of the corresponding gene. We use 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 $k \in \{1, \ldots, p_i - 1\}$ and consider a location of A_i that represents expression level k. First we determine the resources (see Def. 2) that influence the behaviour of A_i in this location. For every $\alpha_j \in Pred(\alpha_i)$ and l_j a location of A_j let

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

Thus, if $\lambda_i^{\alpha_j}(l_j)$ evaluates to *true*, then the location l_j of A_j represents a resource of α_i. If the negation is true, then l_j does not represent a resource. In order to find out whether or not a location change in A_i is indicated, we have to consider the parameter values that determine the dynamics of the system. 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.

Example 8 *In our example from Fig. 1 we obtain for α_2 and $k = 1$ the sets $\omega_1 = \{\alpha_1, \alpha_2\}$, $\upsilon_1 = \emptyset$ and $\upsilon_2 = \{\alpha_2\}$.*

Now we combine the above inequalities to derive conditions that allow us to check whether the composed system is in a state that indicates a change in the expression level of α_i. Let $l \in L_1 \times \cdots \times L_n$, where l_i is the chosen location in A_i, and $\rho = \omega_h$ or $\rho = \upsilon_h$. Then we define

$$\lambda_i^\rho(l) := \bigwedge_{\alpha_j \in \rho} \lambda_i^{\alpha_j}(l_j) \quad \wedge \bigwedge_{\alpha_j \in Pred(\alpha_i) \setminus \rho} \overline{\lambda}_i^{\alpha_j}(l_j).$$

If $\lambda_i^{\omega_h}(l)$ is true for some ω_h, then an increase of expression level of gene α_i is indicated. If $\lambda_i^{\upsilon_h}(l)$ is satisfied for some υ_h, then the expression level will decrease.

Example 9 *For our running example and with ω_1, υ_1 and υ_2 as calculated above we obtain $\lambda_2^{\omega_1}(l) = (\iota(l_1) < 1) \wedge (\iota(l_2) < 2)$, $\lambda_2^{\upsilon_1}(l) = (\iota(l_1) \geq 1) \wedge (\iota(l_2) \geq 2)$ and $\lambda_2^{\upsilon_2}(l) = (\iota(l_1) \geq 1) \wedge (\iota(l_2) < 2)$. Therefore, if the system is in state $l' = (\alpha_1^{0+}, \alpha_2^1)$, for instance, the condition $\lambda_2^{\omega_1}(l')$ is true and the expression level of α_2 should increase, while for $l = (\alpha_1^1, \alpha_2^1)$ condition $\lambda_2^{\upsilon_2}(l)$ is satisfied and indicates expression level decrease. Note that condition $\lambda_2^{\upsilon_1}(l)$ is not satisfied.*

In order to induce a corresponding change in expression level, it is sufficient if the condition $\lambda_i^{\omega_h}(l)$ resp. $\lambda_i^{\upsilon_h}(l)$ holds for some ω_h resp. υ_h. Due to this observation we set

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

We define Λ_i^{0+} and $\Lambda_i^{p_i-}$ accordingly.

Now, we assign all locations α_i^k, $k \in \{1, \ldots, p_i - 1\}$ the conditions Λ_i^{k+} and Λ_i^{k-}. The location α_i^0 resp. $\alpha_i^{p_i}$ is labelled with Λ_i^{0+} resp. $\Lambda_i^{p_i-}$ only, since the location represents the lowest resp. highest expression level possible. Furthermore, we want to check in an intermediate location whether the condition that led to the process of changing the expression level is still valid. If that is not the case, the system should not remain in that location Thus, we associate with location α_i^{k+} the condition $\neg\Lambda_i^{k+}$ for all $k \in \{0, \ldots, p_i - 1\}$, and allot to location α_i^{k-} the condition $\neg\Lambda_i^{k-}$ for all $k \in \{1, \ldots, p_i\}$.

All the above considerations on how the switch conditions should influence the behaviour of the system will be realised in the definition of the timed automaton representing the network dynamics.

Formally speaking, the components defined above are timed automata. However, it does not make sense to evaluate their behaviour in isolation 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. The behaviour of the gene regulatory network is captured when allowing the components to interact, the rules of interaction given by the switch conditions.

4.2 Simplifying the Switch Conditions

The definition of the switch conditions reflects the local character of our modelling approach. The boolean value resulting from the evaluation of the switch conditions obviously depends on the state the system is in. This allows for a flexible description of interactions between the components of the network as already discussed in Section 2.2.

However, given a concrete model, it is often possible to simplify the switch conditions. Assume, for instance, that the Snoussi condition (1) holds and define sets $\omega_1, \ldots, \omega_{m_i^1}, \upsilon_1, \ldots, \upsilon_{m_i^2}$ as in the preceding section. Then it is sufficient to define $\lambda_i^{\omega_h}(l) := \bigwedge_{\alpha_j \in \omega_h} \lambda_i^{\alpha_j}(l_j)$, since the addition of another resource never results in a smaller parameter value. An increase of expression level is indicated if $\bigwedge_{\alpha_j \in \omega_h} \lambda_i^{\alpha_j}(l_j)$ is true regardless of the expression level of predecessors of α_i not contained in ω_h. For the same reason we can define $\lambda_i^{\upsilon_h}(l) := \bigwedge_{\alpha_j \in Pred(\alpha_i) \setminus \upsilon_h} \overline{\lambda}_i^{\alpha_j}(l_j)$.

Moreover, whenever $\omega_{h_1} \subseteq \omega_{h_2}$ for sets ω_h, then $\lambda_i^{\omega_{h_1}}(l)$ is true if $\lambda_i^{\omega_{h_2}}(l)$ is true. Since condition (1) implies that $K_{\alpha_i,\omega_{h_2}} \geq K_{\alpha_i,\omega_{h_1}} > k$, we can delete condition $\lambda_i^{\omega_{h_2}}(l)$ from the expression $\Lambda_i^{k+}(l)$. Analogously, if $\upsilon_{h_1} \subseteq \upsilon_{h_2}$, we can delete the condition $\lambda_i^{\upsilon_{h_1}}(l)$ from the expression $\Lambda_i^{k-}(l)$. Figure 2 shows the components A_1 and A_2 corresponding to the genes of our running example, which satisfies condition (1). The switch conditions are given in the simplified form explained above.

In general, any inequality concerning the expression level of the location the inequality is associated with can be evaluated immediately. For example, the location α_2^2 in Fig. 2 is labelled with the switch condition $\iota(l_1) \geq 1 \vee \iota(l_2) \geq 2$. Since $\iota(\alpha_2^2) \geq 2$, the switch condition in that location is always true, regardless of the state of A_1.

4.3 Modelling the Network

Combining the components $A_1, \ldots, A_n$, we now construct the timed automaton $A_N := (L, L^0, \Sigma, C, I, E)$ representing the whole network N. We define $L := L_1 \times \cdots \times L_n$, $L^0 := L_1^0 \times \cdots \times L_n^0$ and $\Sigma := \{a\} \cup \bigcup_{i \in \{1,\ldots,n\}} \Sigma_i$. Here a stands for 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 $E \subseteq L \times \Sigma \times \Phi(C) \times 2^C \times L$ 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. That is, we preserve the switches of the components.
- Let $(l, a, \varphi, R, l') \in L \times \Sigma \times \Phi(C) \times 2^C \times L$ with $\varphi := true$. Let J be the set of those $j \in \{1, \ldots, n\}$ such that for each $l_j, j \in J$ one of the associated switch conditions 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 $j \in J$,

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

  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+}, Λ_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 motivation of switch conditions of 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 require checking the switch conditions of some location.

Example 10 *Figure 3 shows a part of the automaton derived from the components A_1 and A_2 of our running example. Let us consider the location (α_1^0, α_2^0). There is no switch in A_1 starting in location α_1^0, and neither is there a switch in A_2 starting in α_2^0 as can be seen in Fig. 2. So there is no switch starting in (α_1^0, α_2^0) that originates from the switches of the components.*

Let us now evaluate the switch conditions in α_1^0 and α_2^0 given in Fig. 2. We have $\iota(\alpha_1^0) = \iota(\alpha_2^0) = 0 < 1$. Thus both switch conditions are true in (α_1^0, α_2^0). According to (2) we obtain the switch $((\alpha_1^0, \alpha_2^0), a, true, \{c_1, c_2\}, (\alpha_1^{0+}, \alpha_2^{0+}))$. This switch represents the following situation. If both genes α_1 and α_2 have expression level zero, neither one inhibits the other. Therefore, both genes start to increase their expression level. This is represented by the state $(\alpha_1^{0+}, \alpha_2^{0+})$.

For both components of $(\alpha_1^{0+}, \alpha_2^{0+})$ there exist switches in A_1 resp. A_2. They are preserved in the automaton A and we obtain the switches $((\alpha_1^{0+}, \alpha_2^{0+}), a_1^{0+}, (c_1 \geq t_1^{0+}), \{c_1\}, (\alpha_1^1, \alpha_2^{0+}))$ and $((\alpha_1^{0+}, \alpha_2^{0+}), a_2^{0+}, (c_2 \geq t_2^{0+}), \{c_2\}, (\alpha_1^{0+}, \alpha_2^1))$. They represent the crossing of the corresponding threshold between expression level zero and one. Furthermore, we have to check the switch conditions in α_1^{0+}

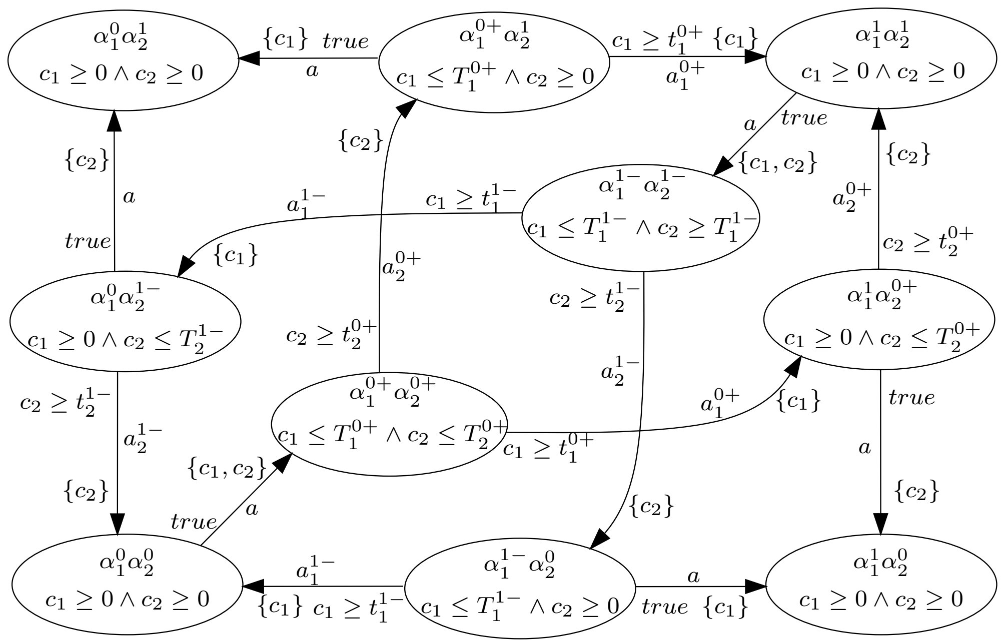


Fig. 3. A part of the timed automaton A constructed from A_1 and A_2 given in Fig. 2.

and α_2^{0+}. Neither one is satisfied, so there is no switch starting in $(\alpha_1^{0+}, \alpha_2^{0+})$ labelled with a.

From state $(\alpha_1^1, \alpha_2^{0+})$ there is the switch $((\alpha_1^1, \alpha_2^{0+}), a_2^{0+}, (c_2 \geq t_2^{0+}), \{c_2\}, (\alpha_1^1, \alpha_2^1))$, which is preserved from A_2, as well as the switch $((\alpha_1^1, \alpha_2^{0+}), a, true, \{c_2\}, (\alpha_1^1, \alpha_2^0))$. The latter reflects the fact that the switch condition in α_2^{0+} is met, while the condition in α_1^1 is not true in state $(\alpha_1^1, \alpha_2^{0+})$.

Note that at this point we have not yet evaluated any of the time constraints placed on switches and locations. In the next step we have to determine which of the paths in the graph representing the automaton A correspond to possible dynamical behaviours.

4.4 The Associated Transition System

Let $T_A = (Q, Q^0, \Gamma, \rightarrow)$ be the transition system associated with A as defined in Section 3. 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 $(l, u) \xrightarrow{a} (l', v)$ to some state $(l', v) \in Q$, 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). To illustrate this, we take a look at location $(\alpha_1^{0+}, \alpha_2^1)$ in Fig 3 and assume we are in a state $(\alpha_1^{0+}, \alpha_2^1, \tau_1, \tau_2)$. Since there is a switch labelled with a starting in this location, we obtain two possible transitions starting in this state, namely $(\alpha_1^{0+}, \alpha_2^1, \tau_1, \tau_2) \xrightarrow{a} (\alpha_1^0, \alpha_2^1, 0, \tau_2)$ and $(\alpha_1^{0+}, \alpha_2^1, \tau_1, \tau_2) \xrightarrow{\alpha_1^{0+}} (\alpha_1^1, \alpha_2^1, 0, \tau_2)$. If we want to put even stronger emphasis on the switches derived from the switch conditions, we delete all transitions other than that labelled with a starting in (l, u), and call a an *overriding* event. Thus, in our example above only the switch mentioned first would remain. Unless otherwise stated, we assume in the following 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 the set 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. This can be interpreted as follows. Starting in a regular state, all commands for a location change caused by the current distribution of resources, evaluated via the switch conditions, happen simultaneously. This leads to a location where each component is either in a location indicating increase or decrease of expression level or remains in the regular location of the original state. In this new state no switch condition of a component location is satisfied, since the expression levels of all components stayed the same. Thus, all transitions starting in the new location are either due to elapse of time or discrete transitions originating from some component A_i.

Again we are able to identify steady states of the system by the lack of outgoing edges. Here, a 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 from Definition 3 can also be obtained from the transition system T_A of a suitable timed automaton A. 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 $\mathcal{S}_N$ be the state transition graph corresponding to N and A the timed automaton derived from N. We set $T_i^{k\varepsilon}, t_i^{k\varepsilon} = 0$ for all $i \in \{0, \dots, n\}$, $\varepsilon \in \{+, -\}$. Thus every guard condition evaluates to *true* and time does not elapse in the intermediate locations.

Next we derive a graph G from T_A. First we identify locations of A_i representing the same expression level, i. e., for $k \in \{1, \dots, p_i - 1\}$ 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 $V^{\alpha_i} := \{v_k^{\alpha_i} \mid k \in \{0, \dots, p_i\}\}$ and $V := V^{\alpha_1} \times \dots \times V^{\alpha_n}$ 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 $v_1 \times \dots \times v_n$. The condition to start in a regular state l ensures 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.

Now, we need to show that $\mathcal{S}_N$ is contained in G. For the sake of completeness we prove the following stronger statement.

Theorem 11 *Suppose that the parameter constraints (1) associated with the Thomas formalism are satisfied. Then the graphs $\mathcal{S}_N$ and G are isomorphic.*

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

Let $s \to s'$ be an edge in $\mathcal{S}_N$. 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 edges in $\mathcal{S}_N$, there is a $j \in \{1, \dots, n\}$ such that $s'_j = s_j + sgn(K_{\alpha_j, R_j(s)} - s_j) \neq s_j$ and $s_i = s'_i$ for all $i \in \{1, \dots, n\} \setminus \{j\}$. 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 $i \in \{1, \dots, n\}$, thus $l \in v_1 \times \dots \times v_n$ is regular. Furthermore, we choose the clock interpretation u that assigns each clock the value zero.

We have $R_j(s) \subseteq Pred(\alpha_j)$ 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 made in the preceding section, every 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 $J \subseteq \{1, \ldots, n\}$ 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 $K_{\alpha_j, R_j(s)} \geq K_{\alpha_j, \omega} > s_j$ according to condition (1). It follows that $s'_= s_j + 1 = s_j + sgn(K_{\alpha_j, R_j(s)} - s_j)$ and thus that $s \to s'$ is an edge in the state transition graph $\mathcal{S}_N$.

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

In 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, e.g. about the actual values of synthesis and decay rates. Thereby we can obtain a more precise description 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 in the next section). Furthermore, we can evaluate stability and feasibility of a certain behaviour, i. e., a 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 latter 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. That is due to the fact that transitions labelled with a never result in a change of expression level. But, on the other hand those transitions are the only ones resulting in a change in more than one component of the location vector. Thus a change of expression level for more than one component cannot happen in a single transition. However, it is possible that two transitions resulting in change of expression level are executed successively, i. e., time does not elapse between those transitions. 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, rendering the viral genome harmless for the so-called lysogenic bacterium.

6.1 Two Genes Model

In [9], the formalism of Thomas is used to describe and analyse the genetic network associated with this behaviour. Figure 1 shows the simplified model they propose, which we already used as a running example in the preceding

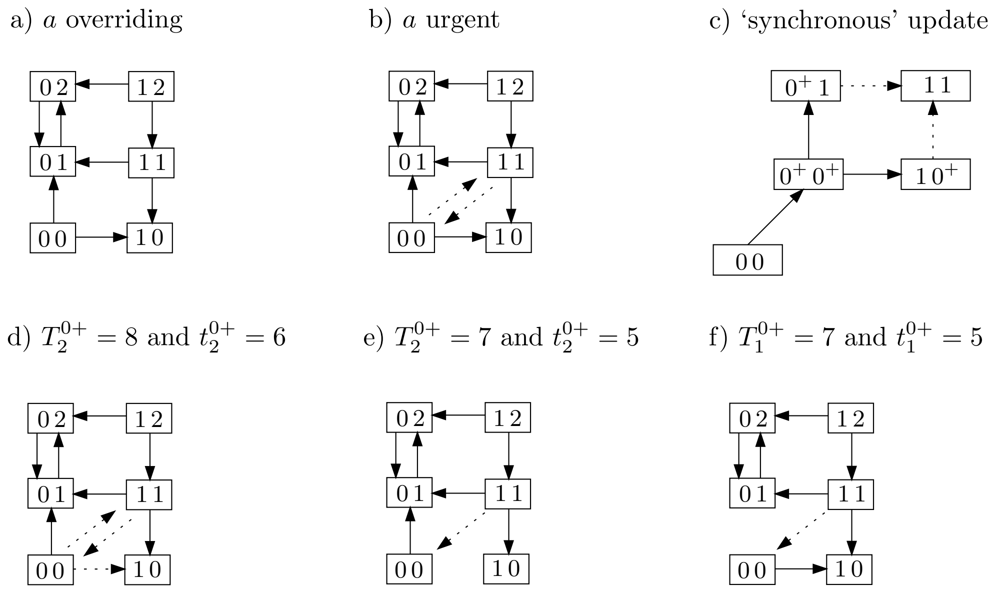


Fig. 4. Graphs representing the dynamical behaviour of the system derived from the transition systems resulting from different specifications of the model. Unless otherwise stated a is an urgent event and we set $T_i^{k+} = T_i^{k-} = 10$ and $t_i^{k+} = t_i^{k-} = 8$ for all $i \in \{1, 2\}$ and $k \in \{0, 1, 2\}$.

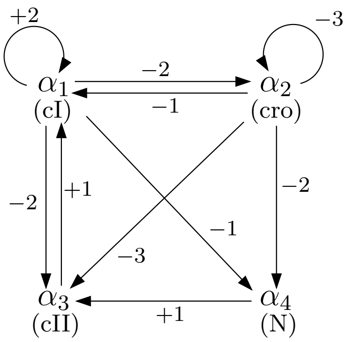
sections. The gene α_1 corresponds to the gene cI and α_2 to the gene cro of the bacteriophage λ. The choice of the thresholds and parameter values is based on experimental data. As already mentioned, they render the loop starting in α_1 ineffective with respect to the dynamics. The resulting state transition graph in Fig. 1 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.

Now we analyse this network modelled as a timed automaton A with components A_1 and A_2, see Fig. 2. A part of A is shown in Fig. 3, but we have not yet considered the corresponding transition system. Figure 4 displays graphs, which are condensed versions of the different transition systems derived from A. 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 location superscript. For instance, states $(\alpha_1^0, \alpha_2^{1-})$ and $(\alpha_1^{0+}, \alpha_2^1)$ 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 a vertex (j_1, j_2) to a vertex (i_1, i_2) signify that the system can evolve from $(\alpha_1^{j_1}, \alpha_2^{j_2})$ to a state where α_1 and α_2 have expression level i_1 and 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 acquire those expression levels. Again graph (c) is an exception to this representation and its analysis will clarify the distinction.

We specify our model by choosing values for the maximal and minimal time delays. Set $T_i^{k+} = T_i^{k-} = 10$ and $t_i^{k+} = t_i^{k-} = 8$ for all $i \in \{1, 2\}$ and $k \in \{0, 1, 2\}$. 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 as explained in Section 4.4. This is illustrated in Fig. 4 (a) and matches the state transition graph in Fig. 1. 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 imposed 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 α_1 reaches the next expression level faster than α_2 after a time delay $8 \leq r_1 \leq 10$, 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_2 that α_2 needs to reach the expression level 1 differs from r_1, 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 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_2^{0+} and t_2^{0+} to express that the synthesis of α_2 is usually faster than that of α_1. The system can reach the state $(1, 0)$ only if α_2 needs the maximal and α_1 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 α_2 is always faster than α_1 in reaching the expression level 1, we can altogether eliminate both the edge leading from $(0, 0)$ to $(1, 0)$, and the one leading to $(1, 1)$, as shown in (e). There is no clock interpretation satisfying the imposed conditions. If we reverse the situation of α_1 and α_2, we eliminate the edges from $(0, 0)$ to $(0, 1)$ and $(1, 1)$ as shown in (f). In this case, the system starting in $(0, 0)$ will always reach the steady state $(1, 0)$ representing the lysogenic response of the bacteriophage.

We have implemented the above system in UPPAAL, a tool for analysing systems modelled as networks of timed automata ([3], and `http://www.uppaal.com`). Since UPPAAL uses product automata in the sense of [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 update mechanisms of the system. Using the UPPAAL model checking engine, we verified the above mentioned dynamical properties of the different specifications of our model.



$$
\begin{array}{lcl@{\qquad}lcl}
K_{\alpha_1,\{\alpha_2\}} & = & 2 & K_{\alpha_2,\{\alpha_1\}} & = & 2 \\
K_{\alpha_1,\{\alpha_3\}} & = & 2 & K_{\alpha_2,\{\alpha_1,\alpha_2\}} & = & 3 \\
K_{\alpha_1,\{\alpha_1,\alpha_2\}} & = & 2 & & & \\
K_{\alpha_1,\{\alpha_1,\alpha_3\}} & = & 2 & K_{\alpha_3,\{\alpha_1,\alpha_2,\alpha_4\}} & = & 1 \\
K_{\alpha_1,\{\alpha_2,\alpha_3\}} & = & 2 & & & \\
K_{\alpha_1,\{\alpha_1,\alpha_2,\alpha_3\}} & = & 2 & K_{\alpha_4,\{\alpha_1,\alpha_2\}} & = & 1
\end{array}
$$

Fig. 5. Extended model of the bacteriophage λ network. Only non-zero parameter values are given

6.2 Four Genes Model

In [9] the authors furthermore discuss a more complete model of the considered bacteriophage system. They include the effects of not only genes *cI* and *cro* but also of genes *cII* and N. The resulting model is shown in Fig. 5, the parameter values are derived from experimental data and theoretical considerations (see [9] for details). Again the lytic and lysogenic behaviour can be identified in the state transition graph. The former is represented by the steady state $(2, 0, 0, 0)$, the latter by the cycle C comprising the states $(0, 2, 0, 0)$ and $(0, 3, 0, 0)$. The timed automaton model consists of four components. Component A_1 corresponding to *cI* includes three regular and four intermediate locations. Component A_2 corresponding to *cro* includes four regular and six intermediate locations, and components A_3 and A_4 corresponding to *cII* and N both have two regular and two intermediate locations. The parameter values satisfy condition (1) which simplifies the switch conditions.

Again we have implemented the model in UPPAAL and analysed the dynamical behaviour for different time constraints, starting with identical time values regardless of corresponding gene and expression level. As a first result, we note that the crucial roles of *cI* and *cro*, i. e., α_1 and α_2, are preserved in the following sense. If the values of T_1^{k+} and t_1^{k+} for $k \in \{0, 1, 2\}$ are sufficiently big in comparison with T_2^{l+} and t_2^{l+} for $l \in \{0, 1, 2, 3\}$, then the state $(2, 0, 0, 0)$ is not reachable from the initial state $(0, 0, 0, 0)$ while the cycle C is reachable. That is, if the expression level increase of *cI* is slow relative to that of *cro*, the system shows lytic behaviour. If the situation is reversed, the system shows lysogenic behaviour. These observations are in accordance with the results for the two gene model as illustrated in Fig. 4.

Now, we want to give an example how to incorporate information on the system gathered from biological experiments (see [5] for an overview). When transcription of the bacteriophage DNA is initiated, *cro* and N are the first of our model components that are expressed. This is exactly the kind of infor-

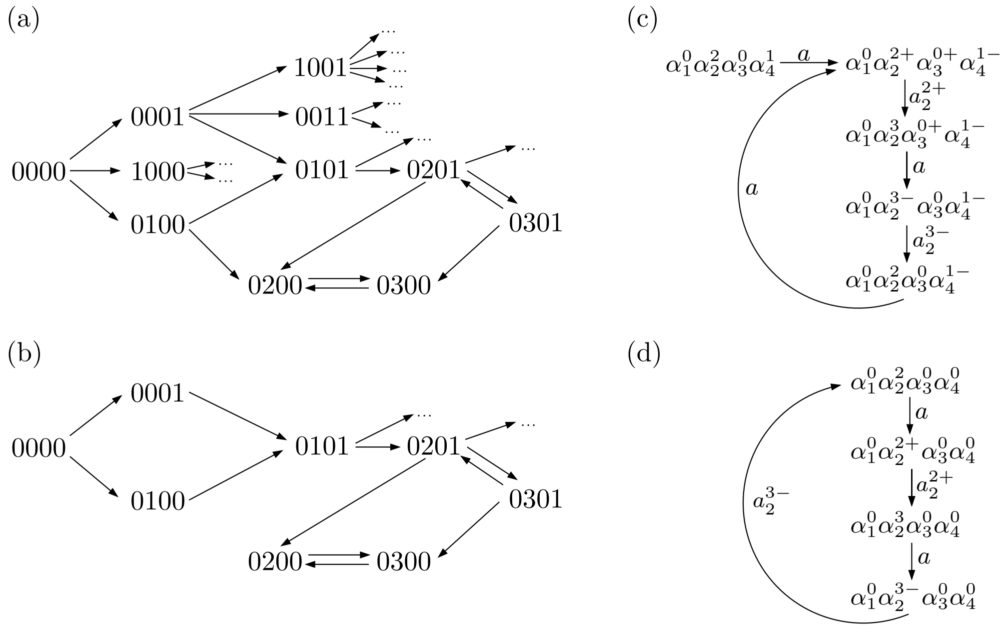


Fig. 6. Parts of the state transition graph resp. transition systems of the model given in Fig. 5

mation we need in order to obtain a more concise model. Starting in the state $(0,0,0,0)$ the given parameter values indicate expression level increase for the genes *cI*, *cro* and N. According to the given data, we choose the values T_2^{0+}, T_4^{0+}, t_2^{0+} and t_4^{0+} smaller than the values T_1^{0+} and t_1^{0+}. The impact on the dynamical behaviour of the system is illustrated in Fig. 6 (a) and (b). In (a) we see part of the state transition graph obtained from the model given in Fig. 5. Starting in the state (0,0,0,0), representing the inactivity of the bacteriophage DNA when introduced into the bacterial cell, all possible changes in expression level, i. e., for genes *cI*, *cro* and N, are taken into account. In the state transition graph of the Thomas formalism we thus have three outgoing edges. From the corresponding target states again every possible change in expression level is included in the graph, resulting in a strongly branched structure. The impact of the additional information on the time constraints given above is shown in (b). The transition from initial state $(0,0,0,0)$ to $(1,0,0,0)$ can be excluded since it violates the temporal constraints. Furthermore, we can derive that the second transition has to lead to state $(0,1,0,1)$, representing expression of the genes *cro* and N in accordance with the experimental observations. Thus, we are able to exclude sizeable parts of the state transition graph.

We want to point out a further advantage of our approach by taking a closer look to the dynamics presented in Fig. 6. The part of the state transition graph given in (a) contains two cycles. First, we have the cycle C representing the lytic behaviour, second the cycle C' comprising the states $(0,2,0,1)$ and $(0,3,0,1)$. According to the information inherent in the state transition graph, the system could remain in the cycle C' indefinitely. Figure 6 (c) shows the cycle in a suitable transition system derived from the timed automaton model. Starting from the regular location $(\alpha_1^0, \alpha_2^2, \alpha_3^0, \alpha_4^1)$ we enter the cycle by evaluating the switch conditions and executing a switch labelled with a. Now, α_2 has to reach its next expression level sooner than α_3 and α_4 in order to reach the state $(\alpha_1^0, \alpha_2^3, \alpha_3^{0+}, \alpha_4^{1-})$, corresponding to the state $(0, 3, 0, 1)$. Continuing in that manner, we derive time constraints that have to be satisfied if the system is to remain in the cycle. However, a close look shows that A_4 always remains in the location α_4^{1-} representing the process of gene N decreasing its expression level from one to zero. After the corresponding time delay, N will reach the expression level zero, thus forcing the system out of the cycle. Cycles with this property have already been considered in [10].

The cycle C representing the lytic behaviour, as illustrated in Fig. 6 (d), shows opposite characteristics. We enter the cycle from the regular state $(\alpha_1^0, \alpha_2^2, \alpha_3^0, \alpha_4^0)$. The given parameter values, evaluated by means of the switch conditions, indicate a change in expression level for gene *cro* only, leading to a transition to $(\alpha_1^0, \alpha_2^{2+}, \alpha_3^0, \alpha_4^1)$. In fact, all transitions in the cycle concern solely gene *cro*. The other genes remain in their respective stable regular location. Thus, we can deduce that the actual values of the time delays concerning the expression level changes from two to three and back do not influence the behaviour of the system after reaching the cycle C.

Lastly, we take another look at the key players in the network. While the genes *cI* and *cro* define the lysogenic and lytic states respectively, studies have shown the importance of *cII* in the switching process (see [5]). Since *cro* and N are the first to be expressed, the activation of *cI* transcription heavily depends on *cII*. As for the time constraints concerning *cI* and *cro* discussed above, we find that we can determine the decision between lysogeny and lytic behaviour by evaluating the *cII* time delays with respect to those of *cro*. Again, if the expression level increase of *cII* is sufficiently slow in comparison with that of *cro*, the system will reach the lytic state. In the reversed situation, we obtain lysogeny. However, the impact of *cII* cannot be correctly captured by our model. The reason for this is the following. The parameter values for *cI* reflect that *cI* can reach its highest expression level in the absence of *cro*, as well as in the presence of *cII*. Thus, if either condition is met, the system moves to the location representing increase of expression level which is labelled with the corresponding time delay. However, the rate for synthesis of *cI* in the presence of *cII* is much higher than in its absence. Thus, we need to consider distinct time delays which depend on the current expression level of *cII*, if we want to capture the resulting impact on the dynamical behaviour. This issue will be addressed again in the following section.

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 the constraints imposed 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 perturbation 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 [11]). It calls for a thorough analysis using mathematical methods as well as testing with substantial biological examples.

Furthermore, it seems worthwhile to relax some of the conditions imposed by the Thomas formalism. We already emphasised the local character of our approach regarding the interactions between the components of the network. Thus we were able to avoid the restrictions induced by the parameter constraints (1). 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 at low but inhibiting at high concentrations. Our approach clearly allows for the modelling of such a situation, however the underlying formal framework, i. e., the definitions of interaction graph, resources and/or parameter values, has to be adapted to obtain a concise description.

A related observation is that our modelling formalism does not allow one to distinguish between processes of expression level change, represented by some intermediate location, caused by different situations. That is, we cannot capture the behaviour of a system where the change of expression level from some level k to $k+1$ (or $k-1$) occurs with different time delays depending on the state of the system. This could be achieved by allowing for different intermediate locations that represent the same process of expression level change. However, some thought has to be given to the possible definition of switches between such intermediate locations.

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] A. Oppenheim, O. Kobiler, J. Stavans, D. Court, and S. Adhya. Switches in bacteriophage lambda development. *Annu. Rev. Genet.*, 39:409–429, 2005.

[6] É. Remy, P. Ruet, and D. Thieffry. Graphic requirements for multistability and attractive cycles in a boolean dynamical framework. Prépublication, 2005.

[7] H. Siebert and A. Bockmayr. Incorporating time delays into the logical analysis of gene regulatory networks. In *Computational Methods in Systems Biology, CMSB 2006, Trento, Italy*, volume 4210 of *LNCS*, pages 169–183. Springer, 2006.

[8] E. H. Snoussi and R. Thomas. Logical identification of all steady states: the concept of feedback loop characteristic states. *Bull. Math. Biol.*, 55:973–991, 1993.

[9] D. Thieffry and R. Thomas. Dynamical behaviour of biological regulatory networks - II. Immunity control in bacteriophage lambda. *Bull. Math. Biol.*, 57:277–297, 1995.

[10] R. Thomas and R. d'Ari. *Biological Feedback*. CRC Press, 1990.

[11] 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.