Analysing formal models of genetic regulatory networks with delays

Jamil Ahmad* and Olivier Roux

IRCCyN UMR CNRS 6597,
BP 92101, 1 rue de la Noë, 44321 Nantes Cedex 3, France
E-mail: jamil.ahmad@irccyn.ec-nantes.fr
E-mail: olivier.roux@irccyn.ec-nantes.fr
*Corresponding author

Gilles Bernot, Jean-Paul Comet and Adrien Richard

Laboratoire I3S, (UNSA & CNRS UMR 6070),
Les Algorithmes, bât. Euclide B, BP.121,
06903 Sophia Antipolis Cedex, France
E-mail: bernot@unice.fr
E-mail: comet@unice.fr
E-mail: richard@unice.fr

Abstract: In this paper, we propose a refinement of the modelling of biological regulatory networks based on the discrete approach of René Thomas. We refine and automatise the use of delays of activation/inhibition in order to specify which variable is more quickly affected by a change of its regulators. The formalism of linear hybrid automata is well suited to allow such refinement. We then use HyTech for two purposes:

- to find automatically all paths from a specified initial state to another one
- to synthesise constraints on the delay parameters in order to follow any specific path.

Keywords: regulatory networks; gene networks; modelling; delays; path; path algorithm; hyTech; cycles; constraints; bioinformatics.

Reference to this paper should be made as follows: Ahmad, J., Roux, O., Bernot, G., Comet, J-P. and Richard, A. (2008) 'Analysing formal models of genetic regulatory networks with delays', *Int. J. Bioinformatics Research and Applications*, Vol. 4, No. 3, pp.240–262.

Biographical notes: Jamil Ahmad is doing his PhD in Bioinformatics at Institut de Recherche en Communications et en Cybernétique de Nantes, Ecole Centrale de Nantes. His area of study is temporal modelling and verification of biological regulatory networks.

Olivier Roux is a Professor of Computer Science at Ecole Centrale de Nantes. He is member of the UMR 6597 IRCCyN and a former member of the Institut Universitaire de France. He is presently Chargé de mission at the French Research Ministry. For several years, he has been involved in research in the areas of Model-checking, hybrid systems and timed properties of complex

systems. Currently, his research focuses on bioinformatics, and modelling and analysis of dynamical complex systems (particularly biological systems).

Gilles Bernot is a Professor in Bioinformatics at University of Sophia Antipolis, France. He is member of the UMR 6070 I3S laboratory. Previously, he was Professor at Genopole in Evry, the French 'Genetics Valley', where he was the founder codirector of the Epigenomics Project. He is Vice-President of the French National Council of Universities in Computer Science. He has been Director of the first bioinformatics research laboratory of Genopole from 1998 to 2004. From 1992 to 1998, he was Professor in formal methods for software engineering and from 1987 to 1992 he was Assistant Professor at the Ecole Normale Supérieure in Paris.

Jean-Paul Comet is a Professor in Bioinformatics at University of Sophia Antipolis, France where he is member of the UMR 6070 I3S laboratory. From 2000 to 2007, he has been Assistant Professor at Genopole in Evry, the French 'Genetics Valley' and member of the FRE 2873 IBISC laboratory and of the Epigenomics Project of Genopole Evry (France). His research interests are in bioinformatics including sequence analysis, expression data analysis and modelling of complex biological systems.

Adrien Richard received his PhD in bioinformatics at the University of Evry, France, in 2006. He is presently CNRS research associate at University of Sophia Antipolis, France, where he is member of the UMR 6070 I3S laboratory. His research interests include discrete dynamical systems and modelling of biological networks.

1 Introduction to Biological Regulatory Networks

Biologists often represent their knowledge on a biological system in terms of graphs (de Jong, 2002). Biological Regulatory Networks (BRN) represent interactions among biological entities. For example, genetic regulatory networks are graphs where vertices represent genes or regulatory products e.g., RNA, proteins and edges represent interactions among them. These interactions are further directed (regulators are distinct from targets) and signed (+ for activation, − for inhibition).

It is now clear for researchers that the semantics of a biological regulatory system and more generally an interaction system, is encoded in the dynamics of the system and not only in the structure of this system. Biologists often need to use the previously described regulatory graphs as a basis for generating dynamical models using either continuous representation or discrete ones.

- In differential models the activity of each gene is represented by a concentration x_i of the associated RNA or proteins, and the evolutions of the vector of all concentrations $x = (x_i)_{i\in[1,n]}$ obey a differential equation system $\mathrm{d}x/\mathrm{d}t = f(x)$. Observation leads biologists to consider non-linear models with some strong threshold effects. The derivation of the dynamics from the interaction graph is not trivial even if the type of each interaction is known, because a lot of parameters have to be inferred, and a small modification of a parameter can lead to a strong change in the dynamics.

- In discrete models, the threshold effects are highlighted and allow modellers to discretise the concentrations. The first approach has been based on drastic discretisation since all genes can be either *on* (present) or *off* (absent) (Thomas, 1978). This boolean approach has been generalised into a multi-valued approach (Snoussi, 1989; Thomas, 1991), in which logical identification of all steady states (Snoussi and Thomas, 1993; Devloo et al., 2003) becomes possible.
  The dynamics of these networks are based on abstraction of continuous-time switching networks which are a special type of hybrid systems as studied in, for example, control theory. Such continuous-time switching networks have been used to model dynamics in, for example, the sporulation network of *Bacillus subtilis* (de Jong et al., 2004). The derivation of the dynamics from the interaction graph remains difficult even if the number of possible models is now finite. Since the formalism consists essentially in the discretisation of the continuous differential equation system, the state space is divided into set of domains representing the symbolic qualitative states of the network.
  The transitions between the different states depend on discrete parameters that play the role of limits of the solutions of the differential equation system of each domain in the continuous space. These limits are sometimes called attractors or targets (Bernot et al., 2004).

The modelling activities then focus on the determination of parameters of the model which lead to a dynamic coherent with the specification (formal translation of experimental facts). Formal verification is not possible in the general framework of differential equation systems. In de Jong et al. (2003) focus on a particular discrete model and use model checking in order to verify if the temporal properties are satisfied. Bernot et al. (2004) proposed to consider all possible parameterisations, to generate all possible dynamics, to call for each of them a model checker for verification and to select only models which lead to a dynamic coherent with the specification. The enormous number of models limits this brute force approach.

One can also notice that the transition systems obtained in the formalism of Thomas (1978) or of de Jong et al. (2003) are not deterministic: they abstract all possible continuous trajectories but they introduce some traces which do not correspond to continuous ones. This is due to a complete and total abstraction of time. To overcome this point Adélaïde and Sutre (2004) showed that under some conditions of equality of degradation constants, this abstraction can lead to a dynamic which does not present the same drawback.

The work of Hill et al. (1998) and Kauffman (2003) is a major contribution towards the dynamical stability and unstability (chaos) of BRN as we are also interested in the analysis of the behaviours which lead to cycles.

In this paper we propose to refine activation and inhibition delays (Thomas, 1973) in the formalism of René Thomas following (Thomas and Kaufman, 2001) where delays have been introduced to study traces closer to the experimental facts. After having briefly presenting René Thomas modelling in Section 2, we introduce in Section 3 the refinement of the concept of delays. In Section 4, a pedagogical example is introduced whereas Section 5 is devoted to two biological examples: we show that phage lambda choice between lytic and lysogenic pathways is controlled by delay values and that T-cell activation and anergy system can be analysed in the same modelling framework. These examples allow us to present an algorithm for searching paths between two specified states (Section 6). Finally, we show how this algorithm can be helpful for parameter synthesis (Section 7). In Section 8, we discuss some problems raised by the presence of unstable cycles. Section 9 is devoted to conclusion.

2 Approach of René Thomas

In a directed graph $G = (V, A)$, we note $G^-(v)$ and $G^+(v)$ the set of predecessors and successors of a node $v \in V$ respectively.

Definition 1: *A biological regulatory network, or BRN for short, is a tuple* $\mathcal{G} = (V, A, l, s, t, K)$ *where*

- (V, A) *is a directed graph denoted by G,*
- l *is a function from* V *to* $\mathbb{N}$
- s *is a function from* A *to* $\{+, -\}$,
- t *is a function from* A *to* $\mathbb{N}$ *such that, for all* $u \in V$, *if* $G^+(u)$ *is not empty then* $\{t(u, v) \mid v \in G^+(u)\} = \{1, \dots, l(u)\}$.
- $K = \{K_v \mid v \in V\}$ *is a set of maps: for each* $v \in V$, K_v *is a function from* $2^{G^-(v)}$ *to* $\{0, \dots, l(v)\}$ *such that* $K_v(\omega) \leq K_v(\omega')$ *for all* $\omega \subseteq \omega' \subseteq G^-(v)$.

The map l describes the domain of each variable v: if $l(v) = k$, the abstract concentration on v holds its value in $\{0, 1, \dots, k\}$. Similarly, the map s represents the sign of the regulation ($+$ for an activation, $-$ for an inhibition).

$t(u, v)$ is the threshold of the regulation from u to v: this regulation takes place iff the abstract concentration of u is above $t(u, v)$, in such a case the regulation is said active. The condition on these thresholds states that each variation of the level of u induces a modification of the set of active regulations starting from u. For all $x \in [0, \dots, l(u) - 1]$, the set of active regulations of u, when the discrete expression level of u is x, differs from the set when the discrete expression level is $x + 1$.

Finally, the map K_v allows us to define what is the effect of a set of regulators on the specific target v. If this set is $\omega \subseteq G^-(v)$, then, the target v is subject to a set of regulations which makes it to evolve towards a particular level $K_v(\omega)$.

Definition 2 (States): *A state* μ *of a BRN* $\mathcal{G} = (V, A, l, s, t, K)$ *is a function from* V *to* $\mathbb{N}$ *such that* $\mu(v) \in \{0, \dots, l(v)\}$ *for all variables* $v \in V$. *We denote* $E^{\mathcal{G}}$ *the set of states of* $\mathcal{G}$.

When $\mu(u) \geq t(u, v)$ and $s(u, v) = +$, we say that u is a resource of v since the activation takes place. Similarly when $\mu(u) < t(u, v)$ and $s(u, v) = -$, u is also a resource of v since the inhibition does not take place (the absence of the inhibition is treated as an activation).

Definition 3 (Resource function): *Let* $\mathcal{G} = (V, A, l, s, t, K)$ *be a BRN. For each* $v \in V$ *we define the resource function* $\omega_v : E^{\mathcal{G}} \to 2^{G^-(v)}$ *by:*

$$\omega_v(\mu) = \{u \in G^-(v) \mid (\mu(u) \geq t(u, v) \text{ and } s(u, v) = +) \text{ or } (\mu(u) < t(u, v) \text{ and } s(u, v) = -)\}.$$

As said before, at state μ, $K_v(\omega_v(\mu))$ gives the level towards which the variable v tends to evolve. We consider three cases,

- if $\mu(v) < K_v(\omega_v(\mu))$ then v can increase by one unit
- if $\mu(v) > K_v(\omega_v(\mu))$ then v can decrease by one unit
- if $\mu(v) = K_v(\omega_v(\mu))$ then v cannot evolve.

Definition 4 (Signs of derivatives): *Let $\mathcal{G} = (V, A, l, s, t, K)$ be a BRN and $v \in V$. We define $\alpha_v : E^{\mathcal{G}} \to \{+1, 0, -1\}$ by*

$$\alpha_v(\mu) = \begin{cases} +1 & if\, K_v(\omega_v(\mu)) > \mu(u) \\ 0 & if\, K_v(\omega_v(\mu)) = \mu(u) \\ -1 & if\, K_v(\omega_v(\mu)) < \mu(u) \end{cases} . \tag{1}$$

The signs of derivatives show the tendency of the solution trajectories and these will be used in the temporal model that we will build in Section 3.

The state graph of BRN represents the set of the states that a BRN can adopt with transitions among them deduced from the previous rules:

Definition 5 (State graph): *Let $\mathcal{G} = (V, A, b, s, t, K)$ be a BRN. The state graph of $\mathcal{G}$ is a directed graph $\mathbf{G} = (E^{\mathcal{G}}, T)$ with $(\mu, \mu') \in T$ if there exists $v \in V$ such that:*

$$\alpha_v(\mu) \neq 0 \quad and \quad \mu'(v) = \mu(v) + \alpha_v(\mu) \quad and \quad \mu(u) = \mu'(u),\ \forall u \in V \backslash \{v\}.$$

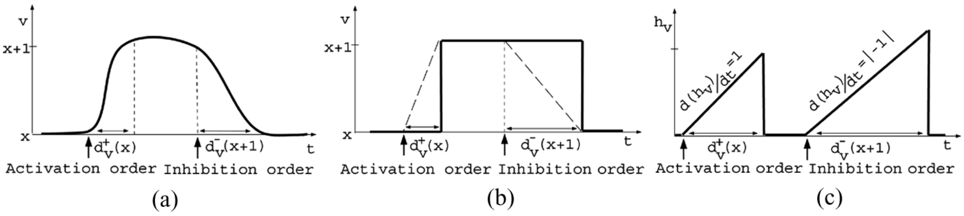
3 Refinement of the concept of delays

In the semantics that we use, a state can have several successors, each of them corresponding to the evolution of the discrete expression level of a unique gene (the dynamic is asynchronous). We refine and automatise the use of delays in the model of René Thomas presented in Section 2. To be more precise, when an order of activation/inhibition arrives, the biological machinery starts to increase/decrease the corresponding protein concentration, but this action takes time. We use two types of parameters, $d_v^+(x)$ and $d_v^-(x)$, to represent the time delay required to change the expression level of a gene v from the level x to $x+1$ and from a level x to $x-1$, respectively, as shown in Figure 1. Then, we add to each variable v a continuous clock h_v whose slope at state μ is $|\alpha_v(\mu)|$. At a given state μ, if $\alpha_v(\mu) = +1$ (resp. $\alpha_v(\mu) = -1$), then, when h_v reaches $d_v^+(\mu(v))$ (resp. $d_v^-(\mu(v))$), the level of v becomes $\mu(v) + 1$ (resp. $\mu(v) - 1$) and the clock h_v is reset.

The temporal model described above belongs to the class of the so-called stopwatch automata (Cassez and Larsen, 2000) which is a specific type of Linear Hybrid Automata (LHA) (Alur et al., 1992, 1995). LHA are finite state automata augmented with real variables whose values evolve continuously in a discrete state. The values of the continuous variables can be affected by discrete transitions between discrete states. LHA implies that the solutions to the differential equations are lines. Linear hybrid automata can be subject to a reachability analysis to verify that a given set of assertions is true. However, in general, the reachability problem for linear hybrid automata is undecidable Thomas et al. (1998).

In the following, we present a method allowing to synthesise constraints on delays which are necessary and sufficient for the system to follow a given path of the state graph. Before, the temporal model presented here is illustrated with a pedagogical example and two biological examples.

Figure 1 The actual evolution of a gene's expression (a) the discrete model (b) along with the temporal model (c)



(a) (b) (c)

4 Preliminary example of a Toy gene regulatory network

Consider the BRN involving three genes, a, b and c, which interact according to the signed graph of Figure 2, where $l(a) = l(b) = l(c) = 1$, and where $K_a(\{c\}) = K_b(\{a\}) = K_c(\{b\}) = 1$ and $K_a(\{\}) = K_b(\{\}) = K_c(\{\}) = 0$. From this BRN, we obtain the table of resources (Table 1) and state graph as shown in Figure 3.

Figure 2 Example of a BRN

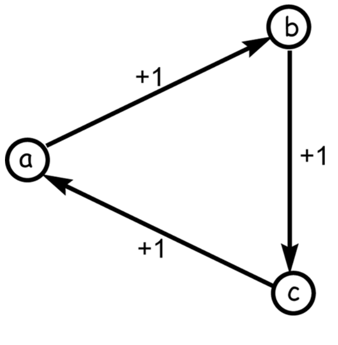


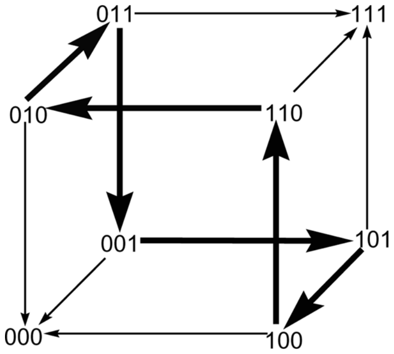
Table 1 gives the set of resources $\omega_v(\mu)$ of gene $v \in \{a, b, c\}$, and the corresponding parameter $K_v(\omega_v(\mu))$, according to the state μ of the BRN. This table helps the reader to reconstruct the state graph representing the dynamics of the BRN. The edges in the graph give the possible transitions between states which can occur after certain time delays. The circuit $(010, 011, 001, 101, 100, 110, 010)$ represents an unstable circuit in the dynamics and the two states not involved in the circuit are the only two stable states.

As we have explained in Section 3, we associate a clock h_v, with each variable $v \in \{a, b, c\}$. All clocks of the system increase continuously and simultaneously with either slope 1 or slope 0. The guard $h_v == d_v^\alpha$, where $\alpha \in \{+, -\}$ is a condition which means that the delay has accomplished. The clock of a variable is set to zero when this variable changes its abstract level. In Figure 4 all the transitions are labelled with guards and clock initialisations.

Table 1 Table of resources of the BRN of Figure 2

$\mu(a)$	$\mu(b)$	$\mu(c)$	$\omega_a(\mu)$	$\omega_b(\mu)$	$\omega_c(\mu)$	$K_a(\omega_a(\mu))$	$K_b(\omega_b(\mu))$	$K_c(\omega_c(\mu))$
0	0	0	$\{\}$	$\{\}$	$\{\}$	0	0	0
0	0	1	$\{c\}$	$\{\}$	$\{\}$	1	0	0
0	1	0	$\{\}$	$\{\}$	$\{b\}$	0	0	1
0	1	1	$\{c\}$	$\{\}$	$\{b\}$	1	0	1
1	0	0	$\{\}$	$\{a\}$	$\{\}$	0	1	0
1	0	1	$\{c\}$	$\{a\}$	$\{\}$	1	1	0
1	1	0	$\{\}$	$\{a\}$	$\{b\}$	0	1	1
1	1	1	$\{c\}$	$\{a\}$	$\{b\}$	1	1	1

Figure 3 State graph of the BRN of Figure 2



5 Biological examples

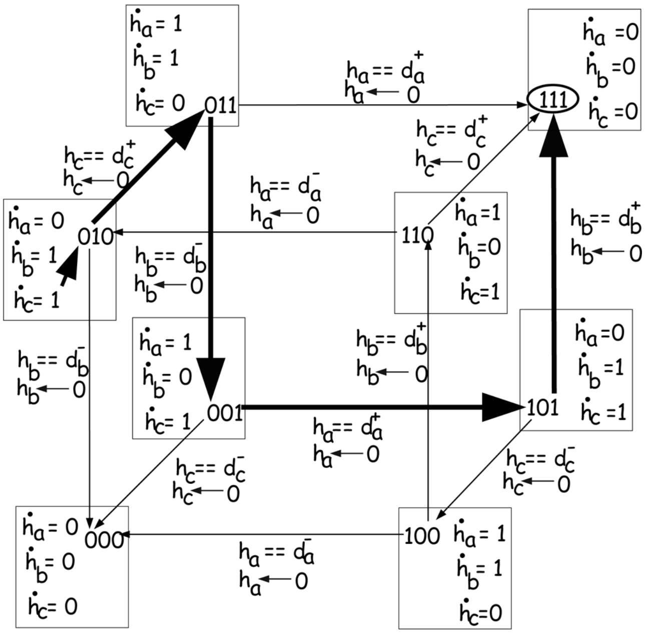
In this section, we present two biological examples. The first example is the biological regulatory network of *lambda phage* while the second one is the example of activation and anergy system of a *T-cell*. The purpose of introducing two more examples in this section is to show how useful are our modelling of BRN and its HyTech analysis for real biological regulatory systems.

5.1 Example of a Biological Regulatory Network of Lambda phage

In this sub-section, we present the example of a biological regulatory network of *lambda phage*. This example has been explained in more detail in the thesis report of Thieffry (1993) as well as in Thomas (1979). First, we briefly describe this example and then we apply our method to distinguish different paths for lytic and lysogenic cycles in Section 7.

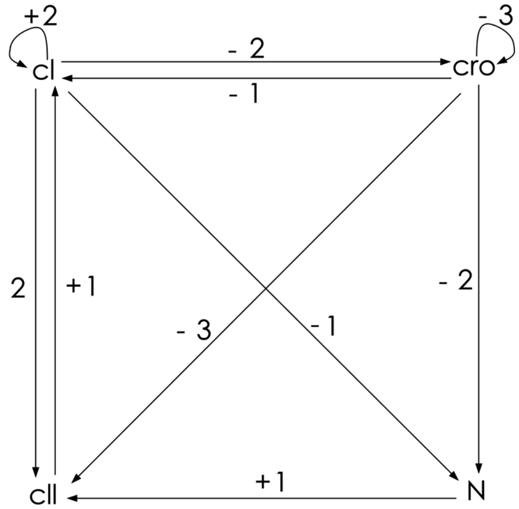
The *lambda phage* first attaches to its host *Escherichia coli* and then after uncoating, it injects its DNA in its hosts. The DNA circularises and integrates into the host DNA and then replicates with the host cell (lysogenic pathway). The phages will remain in the lysogenic cycle if cI proteins predominate, otherwise, if Cro proteins predominate, phages enter in the lytic way leading to the cell lysis.

Figure 4 Stopwatch automaton for the BRN of Figure 2



$\dot{h}_a = 1$, $\dot{h}_b = 1$, $\dot{h}_c = 0$, 011
$\dot{h}_a = 0$, $\dot{h}_b = 0$, $\dot{h}_c = 0$, 111
$h_a == d_a^+$, $h_a \leftarrow 0$
$h_c == d_c^+$, $h_c \leftarrow 0$
$h_a == d_a^-$, $h_a \leftarrow 0$
$\dot{h}_a = 0$, $\dot{h}_b = 1$, $\dot{h}_c = 1$, 010
$h_b == d_b^-$, $h_b \leftarrow 0$
110, $\dot{h}_a = 1$, $\dot{h}_b = 0$, $\dot{h}_c = 1$
$h_b == d_b^+$, $h_b \leftarrow 0$
$\dot{h}_a = 1$, $\dot{h}_b = 0$, $\dot{h}_c = 1$, 001
$h_b == d_b^+$, $h_b \leftarrow 0$
$\dot{h}_a = 0$, $\dot{h}_b = 1$, 101, $\dot{h}_c = 1$
$h_b == d_b^-$, $h_b \leftarrow 0$
$h_a == d_a^+$, $h_a \leftarrow 0$
$h_c == d_c^-$, $h_c \leftarrow 0$
$h_c == d_c^-$, $h_c \leftarrow 0$
$\dot{h}_a = 0$, $\dot{h}_b = 0$, $\dot{h}_c = 0$, 000
$h_a == d_a^-$, $h_a \leftarrow 0$
100, $\dot{h}_a = 1$, $\dot{h}_b = 1$, $\dot{h}_c = 0$

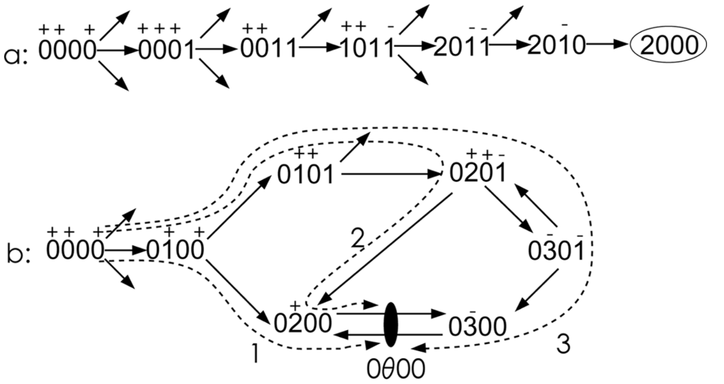
Figure 5 The interaction graph for the regulation of the expression of *lambda phage*, containing only the genes (cl, cro, cll and N) directly involved in the feedback circuit



The biological regulatory network of Figure 5 consists of four genes cl, cro, cll and N represented by variables x, y, z and u respectively, which are mainly involved in the developmental pathways: lytic and lysogenic cycles as shown in Figure 6. The transition model of this example that consists of 48 states and its behavioural model have been explained in the previously mentioned dissertation.

In Thieffry (1993), it has been shown that many paths from the initial state 0000 are possible but the one that leads to the lysogenic state (2000) or to the lytic cycle (around 0θ00) consisting of states 0200 and 0300, as shown in Figure 6, depends on the values of the corresponding delays. We have shown in Table 3 of Section 7, the temporal regions in which the trajectories from state 0000 will always arrive at state 2000 (lysogenic cycle) or at states 0200 and 0300 around 0θ00[1] (lytic cycle).

Figure 6 (a) A part of state graph showing two pathways for lysogenic cycle and (b) lytic cycle. The subscribed sign + (resp. −) stands for expressing the possibility for a concentration to increase (resp. decrease)



Source: Taken from Thieffry (1993, p.138)

5.2 Example of a T-cell activation and anergy system

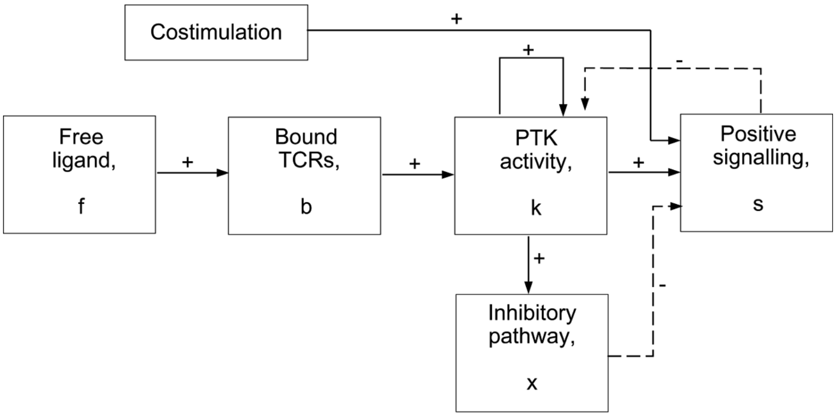
In this sub-section, first, we briefly present the modelling of the activation and anergy system of *T-cell* by the formalism of René Thomas as explained in Kaufman et al. (1999) and then we will present in Section 7, how to automatically obtain, using our BRN modelling approach, the same results as manually obtained in Kaufman et al. (1999). Readers can find in Kam et al. (2001) the statecharts modelling of the *T-cell* activation which mainly addresses the synthesis of different analysis information about *T-cell*.

The initiation and regulation of an immune response to an antigen are dependent on the activation of helper *T lymphocytes* by an appropriate antigen/major histocompatibility complex ligand (**Ag/MHC**). The *T-cell* antigen receptor (**TCR**) by its cognate ligand triggers a series of biochemical events within the cell, which can lead either to cellular activation, i.e., lymphokine production and cell proliferation, or to an induction of a state of unresponsiveness termed anergy.

The model of Figure 7 represents a sequence of events, each requiring a characteristic time to be realised. Binding of free ligand to the **TCRs** activates the receptor-associated tyrosine kinases. Receptor engagement together with protein tyrosine kinase (**PTK**) activation leads to positive signalling. Activated **PTKs** also trigger an inhibitory pathway, which negatively affects the positive response.

Costimulation participates in signal generation. The positive action of the **PTKs** on themselves indicates that once receptor-associated **PTK** activity is established, it remains sustained even in the absence of ligand. This autocatalytic maintenance mechanism is suppressed by interleukin 2 (**IL-2**) linked signalling and proliferation. Except for the initial increase in **PTK** activity, the signal here includes both the early (calcium rise, protein kinase C, and Ras activation) and late transduction events (up-regulation of **IL-2** receptors, **IL-2** production, and cell proliferation). A distinction between different subsets of transduction events, while allowing to account for a wider variety of partial responses, does not change the basic properties of the model.

Figure 7 Schematic interaction diagram. f = free; b = **TCRs** bound to ligand; k = receptor-associated **PTK** activity; x = tyrosine kinase-dependent inhibitory pathway; s = metabolic and mitogenic response. Positive and negative interactions are indicated by a plus and minus sign respectively



Source: Taken from Kaufman et al. (1999, p.3895)

The state of the system is defined in terms of four logical variables that take the logical values 0 or 1. Thus, $b = 1$ means receptor bound to ligand, otherwise $b = 0$; $k = 1$ means receptor-associated **PTKs** activated, otherwise $k = 0$; $x = 1$ means inhibitory pathway activated, otherwise $x = 0$; $s = 1$ means activatory pathway activated, otherwise $s = 0$.

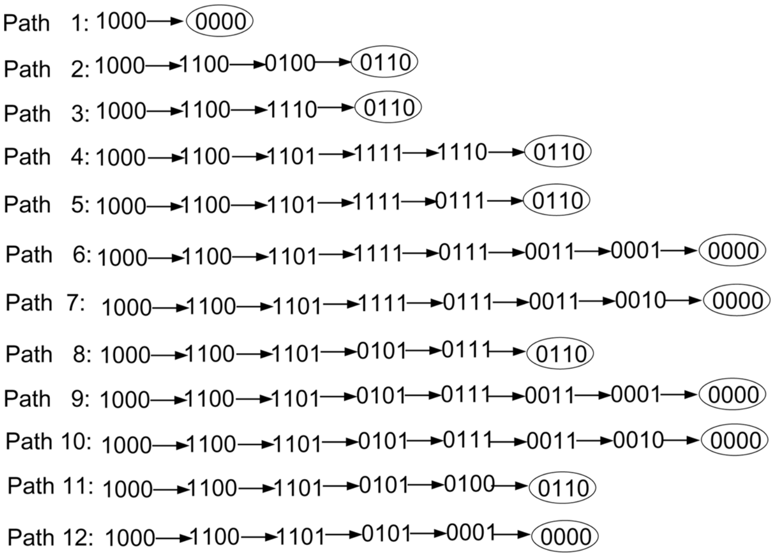
Moreover to each variable b, k, x, and s is associated a boolean function B, K, X, and S which represents the activity on the promoters : if the promotor is activated, then the function is equal to 1. The system is therefore represented in the sequential logical way, at any time, the future state(s) of the system is the function of the present state of the system. For the model of Figure 7 the set of logical equations are:

$$\begin{cases} B = 0 \\ K = b + k.\bar{s} \\ X = k \\ S = b.k.\bar{x} \end{cases} .$$

In the above set of equations, function B describes the natural tendency of a bound **TCR** to dissociate from ligand, irrespective of the presence or absence of free ligand. Function K expresses that receptor-associated tyrosine kinase activity will increase above basal level if the receptors are bound to ligand ($b = 1$). It also involves a maintenance mechanism, through autophosphorylation, that persists in the absence of ligand-bound TCRs. This maintenance mechanism is suppressed by the occurrence of **IL-2**-linked signalling events and proliferation ($s = 1$). Thus, $K = 1$ (and k tends toward or remains at 1) if either or both $b = 1$, $k = 1$; otherwise $K = 0$. Function X states that the receptor-associated tyrosine kinases activate, directly or indirectly, an inhibitory pathway. $X = 1$ if $k = 1$; otherwise $X = 0$. Finally, function S means that positive signalling (i.e., cellular activation) will occur ($S = 1$) if the receptors are bound ($b = 1$) and the receptor-associated kinases are activated ($k = 1$) and if their positive action on cell activation is not inhibited ($x = 0$); otherwise $S = 0$.

Figure 8 shows various pathways from state (1000) where the receptor has just become bound by ligand.

Figure 8 Various pathways leading to full immunocompetence state (0000) and anergic state (0110)


Path 1: 1000 → 0000
Path 2: 1000 → 1100 → 0100 → 0110
Path 3: 1000 → 1100 → 1110 → 0110
Path 4: 1000 → 1100 → 1101 → 1111 → 1110 → 0110
Path 5: 1000 → 1100 → 1101 → 1111 → 0111 → 0110
Path 6: 1000 → 1100 → 1101 → 1111 → 0111 → 0011 → 0001 → 0000
Path 7: 1000 → 1100 → 1101 → 1111 → 0111 → 0011 → 0010 → 0000
Path 8: 1000 → 1100 → 1101 → 0101 → 0111 → 0110
Path 9: 1000 → 1100 → 1101 → 0101 → 0111 → 0011 → 0001 → 0000
Path 10: 1000 → 1100 → 1101 → 0101 → 0111 → 0011 → 0010 → 0000
Path 11: 1000 → 1100 → 1101 → 0101 → 0100 → 0110
Path 12: 1000 → 1100 → 1101 → 0101 → 0001 → 0000

The variable is aligned with the value of the function if the values of a logical function and its corresponding variable disagree. This alignment is executed after a characteristic time delay, unless a counter order is received before the delay has elapsed. For example, the value of x will align on the value $X = 1$ after a time delay d_x^+, provided there has not been a counter order, $X = 0$, before expiration of the delay. Similarly, if $x = 1$ and $X = 0$, then x will shift from $x = 1$ to $x = 0$ in a characteristic time d_x^-. `On` and `off` delays for the other state variables are denoted similarly.

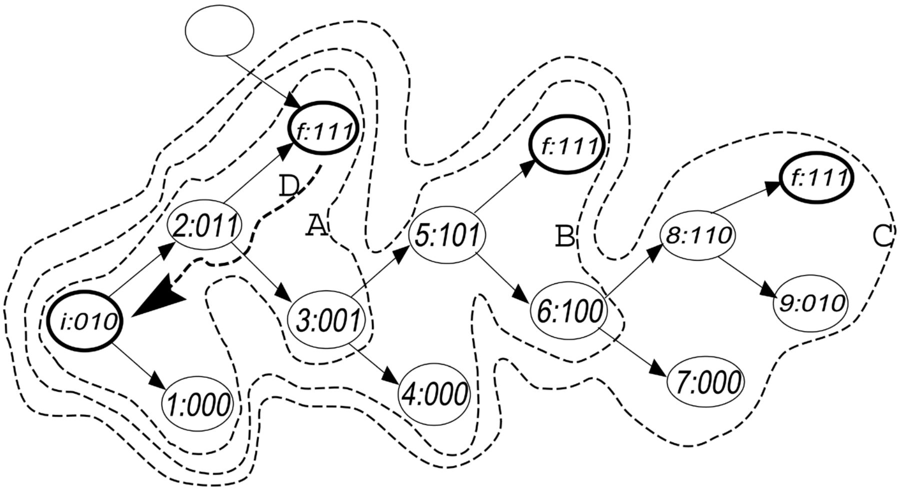
Table 4 of Section 7.3, contains the parameter constraints of delays for the transitions in paths of Figure 8. One can compare, in Section 7.3, our HyTech results of parameter constraints with those manually obtained in Kaufman et al. (1999).

6 Searching paths between two states

The analysis of the hybrid refinement of the BRN is performed by using a linear hybrid model checker HyTech (Henzinger et al., 1997). The delays are defined as parameters whose values are unknown. Our path Algorithm 1 finds all the possible paths between two qualitative states of a regulatory network. We have implemented this algorithm in HyTech (see Appendix A.1) for two purposes:

- to find the exact number of paths between any two states of a BRN
- to automatically synthesise parameter constraints for each transition within a path.

Figure 9 How the algorithm finds the final states from initial state and vice versa. The empty state shows the accessibility of final state through other path



Algorithm 1 is the pseudocode of the HyTech implementation. `pre` and `post` operators, which are the classical predecessor and successor operators of hybrid systems analysis, return respectively the predecessors and successors of a state including the state itself. The difficulty of the algorithm lies in the fact that it converts the breadth-first search (induced by the `post` operator) into the depth-first search of a path. The algorithm consists of two main loops. In the outer loop the algorithm exhaustively searches the `final_state` from the `initial_state` and accumulates the accessed states in a set named `states_accumulated`. When the algorithm finds the `final_state` then it starts the nested loop and begins backward search from `final_state` and takes the intersection of each accessed states with the set `states_accumulated`. If the intersection is not empty then the algorithm gives the intersection as a state of the path which is accumulated in a set `path_states`. Finally the algorithm invokes the procedure `print_path(path_states)` to print the states of a path in proper order.

The dashed lines (`A`), (`B`) and (`C`) of Figure 9 represent the successive sets of accumulated states when the algorithm finds the final state `f` during the outer loop. The inner loop is used for backward search and the dashed arrow (`D`) shows this search for the first path. In Figure 9, the set A is equal to `post(post(010))` and the set of path states is equal to `post(post(010))` $\bigcap$ `pre(pre(111))`.

Algorithm 1 finds the three paths between states (0,1,0) and (1,1,1) in the example of Figure 4.

Algorithm 1 Finds paths between two states

```
Path(initial_state, final_state)
states_accessed:=initial_state; // The first accessed states is the initial state
reached := initial_state; // The first visited state is the initial state
path := initial_state; // Path is set to initial state
states_accumulated := initial_state;
// while exist accessible states of a BRN from initial state
while not_empty(states_accessed) do
    // The set of accessed states is now the successors of the previously accessed states
    states_accessed := post(states_accessed) - states_accessed;
    path := states_accessed - path; // Find new states of a path
    states_accumulated := states_accumulated $\bigcup$ path; // Set that accumulates the states
    states_accumulated := states_accumulated - initial_state; // Remove the initial state from the set
    states_accessed := states_accessed - reached; // Remove all previously visited states from set of accessed states
    reached :=reached $\bigcup$ states_accessed; // The previously visited states will now be the accessed states
    // Check if final state is accessed
    if not_empty(path $\bigcap$ final_state) then
        states_accessed1:= final_state;
        reached1 := final_state;
        path1:=final_state;
        //The nested loop starts here
        while not_empty(states_accessed1) do
            // The set of accessed states is now the predecessors of the previously
            // accessed states in a set of accumulated states
            states_accessed1 := (pre(states_accessed1)-states_accessed1) $\bigcap$ states_accumulated;
            path1:=states_accessed1 - path1;
            path_states := path_states $\bigcup$ path1; // Set that accumulates the states of one discovered path
            states_accessed1 := states_accessed1 - reached1;
            reached1:= reached1 $\bigcup$ states_accessed1;
        end while
        print_path(path_states); // To print the states of a path
    end if
end while
```

7 Parameters synthesis

In this section, we present the HyTech results of parameters synthesis, for the three examples presented in Sections 4 and 5. The HyTech results are presented in Tables 2–4. In these tables, the two/three-columned rows show the transitions and their respective parameter constraints while the one-columned rows show the labelled paths and their equivalent temporal regions in terms of delay parameter constraints.

7.1 HyTech results for the example of Toy gene regulatory network

In this sub-section, we present the HyTech result of the parameters synthesis for the preliminary example of Section 4.

The delay parameters used for the increasing and decreasing of the expression level of a gene v can be synthesised in HyTech to form timing constraints for each transition that takes place in paths between two states. The conjunction of constraints along any sequence of transitions gives the synthesised parameters constraint for the given path.

Now if we desire to find only the path shown by the bold line in Figure 4, then we use the timing constraints which are synthesised by HyTech for the transitions in one path starting from a state where $h_a = h_b = h_c = 0$ (see below). Thus, we draw an equivalence between a path and the region described by the conjunction of its associated constraints. Table 2 shows the constraints synthesised by HyTech for the example of Section 4.

Table 2 HyTech results of the delay constraints for the path shown by the bold line in Figure 4 ($\wedge$ means logical AND)

Transitions	*Constraints*
$010 \rightarrow 011$	$d_c^+ \leq d_b^-$
$011 \rightarrow 001$	$d_b^- \leq d_a^+ + d_c^+$
$001 \rightarrow 101$	$d_a^+ + d_c^+ \leq d_b^- + d_c^-$
$101 \rightarrow 111$	$d_a^+ + d_b^+ + d_c^+ \leq d_b^- + d_c^-$
$010 \rightarrow 011 \rightarrow 001 \rightarrow 101 \rightarrow 111$ $\equiv (d_c^+ \leq d_b^-) \wedge (d_b^- \leq d_a^+ + d_c^+) \wedge (d_a^+ + d_c^+ \leq d_b^- + d_c^-) \wedge (d_a^+ + d_b^+ + d_c^+ \leq d_b^- + d_c^-)$	

Table 3 HyTech results of the delay constraints for the paths shown in Figure 6

Transitions	*Constraints*
$0000 \rightarrow 0001$	$d_u^+ \leq d_x^+ \wedge d_u^+ \leq d_y^+$
$0001 \rightarrow 0011$	$d_z^+ + d_u^+ \leq d_x^+ \wedge d_z^+ + d_u^+ \leq d_y^+$
$0011 \rightarrow 1011$	$d_x^+ \leq d_y^+$
$1011 \rightarrow 2011$	$2d_x^+ \leq d_y^+ \wedge d_x^+ \leq d_u^-$
$2011 \rightarrow 2010$	$d_u^+ \leq d_x^+ + d_z^-$
$0000 \rightarrow 0100$	$d_y^+ \leq d_x^+ \wedge d_y^+ \leq d_u^+$
$0100 \rightarrow 0200$	$2d_y^+ \leq d_u^+$
$0100 \rightarrow 0101$	$d_u^+ \geq 2d_y^+$
$0101 \rightarrow 0201$	$2d_y^+ \leq d_z^+ + d_u^+$
$0201 \rightarrow 0200$	$2d_y^+ + d_u^- \leq d_z^+ + d_u^+ \wedge d_u^- \leq d_y^+$
$0201 \rightarrow 0301$	$3d_y^+ \leq d_z^+ + d_u^+ \wedge d_y^+ \leq d_u^-$
$0301 \rightarrow 0300$	$d_u^- \leq d_y^-$
Path a: $0000 \rightarrow 0001 \rightarrow 0011 \rightarrow 1011 \rightarrow 2011 \rightarrow 2010 \rightarrow 2000$ $\equiv (d_u^+ \leq d_x^+ \wedge d_u^+ \leq d_y^+) \wedge (d_z^+ + d_u^+ \leq d_x^+ \wedge d_z^+ + d_u^+ \leq d_y^+) \wedge (d_x^+ \leq d_y^+)$ $\wedge(2d_x^+ \leq d_y^+ \wedge d_x^+ \leq d_u^-) \wedge (d_u^+ \leq d_x^+ + d_z^-)$	
Path b-1: $0000 \rightarrow 0100 \rightarrow 0200$ $\equiv (d_y^+ \leq d_x^+ \wedge d_y^+ \leq d_u^+) \wedge (2d_y^+ \leq d_u^+)$	
Path b-2: $0000 \rightarrow 0100 \rightarrow 0101 \rightarrow 0201 \rightarrow 0200$ $\equiv (d_y^+ \leq d_x^+ \wedge d_y^+ \leq d_u^+) \wedge (d_u^+ \geq 2d_y^+) \wedge (2d_y^+ \leq d_z^+ + d_u^+)\wedge$ $(2d_y^+ + d_u^- \leq d_z^+ + d_u^+ \wedge d_u^- \leq d_y^+)$	
Path b-3: $0000 \rightarrow 0100 \rightarrow 0101 \rightarrow 0201 \rightarrow 0301 \rightarrow 0300$ $\equiv (d_y^+ \leq d_x^+ \wedge d_y^+ \leq d_u^+) \wedge (d_u^+ \geq 2d_y^+) \wedge (3d_y^+ \leq d_z^+ + d_u^+ \wedge d_y^+ \leq d_u^-)$ $\wedge(2d_y^+ \leq d_z^+ + d_u^+) \wedge (d_u^+ \leq d_y^-)$	

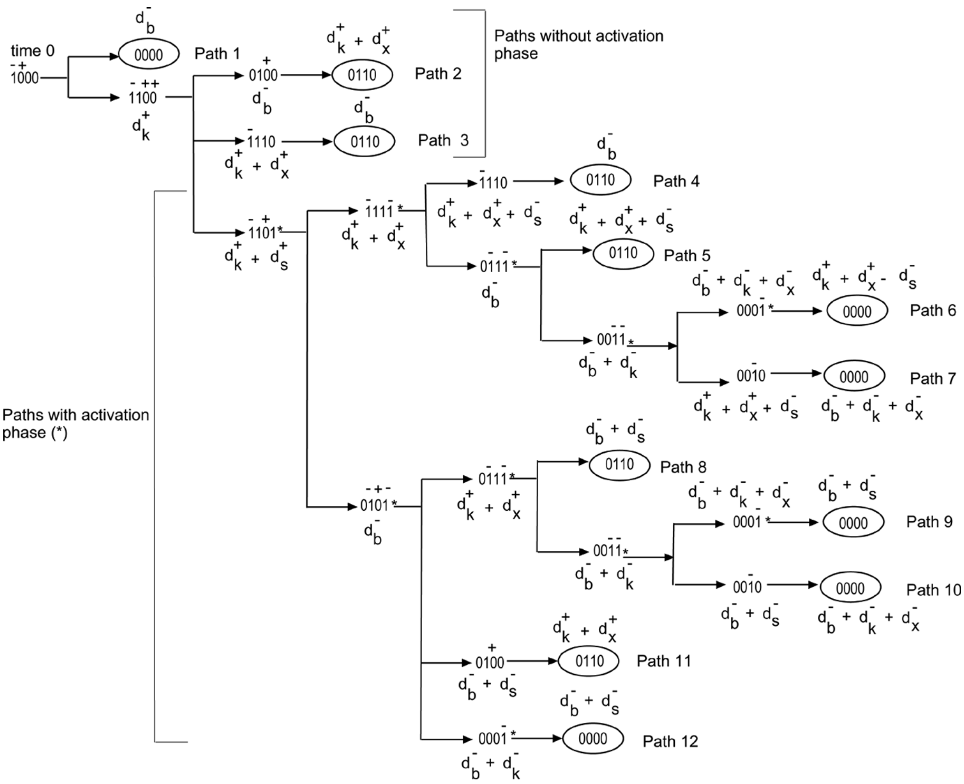
7.2 HyTech results for the example of Lambda phage

In this sub-section, we present the HyTech results of the parameter synthesis for the example of *lambda phage* of Section 5.1. Here, we show paths leading to lytic and lysogenic states and their delay parameter constraints, representing equivalent temporal regions. The equivalent regions give a choice among paths for lysogenesis and lysis (of Figure 5), as labelled in Table 3 by Path a and Paths b (b-1, b-2, b-3) respectively.

7.3 HyTech results for the T-cell activation and anergy system

In this sub-section, we present the HyTech results of the parameter synthesis for the example of *T-cell* activation and anergy system of Section 5.2, along with the results given in Kaufman et al. (1999) for the same system, as shown here in Figure 10. We notice that HyTech gives the same results of delay constraints as manually calculated in Kaufman et al. (1999) and therefore this shows the usefulness of BRN modelling along with the HyTech tool for the temporal analysis of regulatory networks.

Figure 10 State graph showing the total time to reach a given state, relative to time 0 in terms of time delays



Source: Taken from Kaufman et al. (1999, p.3897)

Various timing dependent signalling properties presented in Kaufman et al. (1999, pp.3896–3898) are given below.

- *Fast ligand dissociation.* Path 1 is followed if ligand dissociation precedes kinase activation ($\mathbf{d_b^-} < \mathbf{d_k^+}$) and corresponds to what is observed for 'null' or inactive ligands. Path 2 is followed if ligand dissociation is slower than kinase activation but faster than significant activation of the stimulatory and inhibitory pathways: $\mathbf{d_k^+} < \mathbf{d_b^-} < \mathrm{Min}\,(\mathbf{d_k^+} + \mathbf{d_s^+}, \mathbf{d_k^+} + \mathbf{d_x^+})$.

- *Absence of costimulation.* If $\mathbf{d}_{\mathbf{x}}^{+} < \mathbf{d}_{\mathbf{s}}^{+}$ and $\mathbf{d}_{\mathbf{k}}^{+} + \mathbf{d}_{\mathbf{x}}^{+} < \mathbf{d}_{\mathbf{b}}^{-}$, path 3 will be followed. Here, the receptor-associated kinases are activated but inhibition precedes significant signal transmission, so that there is no positive signalling. After ligand dissociation, the system ends up again in the unresponsive state and restimulation in optimal conditions does not lead to cellular activation.
- *Positive signalling.* The general conditions for positive signalling are: $\mathbf{d}_{\mathbf{s}}^{+} < \mathbf{d}_{\mathbf{x}}^{+}$ and $\mathbf{d}_{\mathbf{k}}^{+} + \mathbf{d}_{\mathbf{s}}^{+} < \mathbf{d}_{\mathbf{b}}^{-}$. Positive signalling should be faster than significant inhibition, and the ligand residence time must exceed the time required for activation of the kinases and signal transmission. These timing conditions correspond to paths 4–12 and account for signalling upon stimulation with an activatory ligand in the presence of costimulation. Along paths 8–12 ($\mathbf{d}_{\mathbf{b}}^{-} < \mathbf{d}_{\mathbf{k}}^{+} + \mathbf{d}_{\mathbf{x}}^{+}$), the time length of the activation phase essentially is determined by the ligand residence time on the receptors, whereas along paths 4–7 ($\mathbf{d}_{\mathbf{k}}^{+} + \mathbf{d}_{\mathbf{x}}^{+} < \mathbf{d}_{\mathbf{b}}^{-}$), it is mainly determined by the time lag between positive signalling and inhibition.

After activation the system will end up in one of two stable steady states, corresponding either to recovery of responsiveness (0000) or to anergy (0110). Independently of the precise pathway that is followed, the necessary and sufficient conditions to reach each of these two final states can be determined by using the tools of combinatorial logic. These additional conditions follow.

- For positive signalling with recovery of responsiveness:

  $$(\mathbf{d}_{\mathbf{s}}^{-} > \mathbf{d}_{\mathbf{k}}^{-}) \quad \text{AND} \quad (\mathbf{d}_{\mathbf{b}}^{-} < \mathbf{d}_{\mathbf{k}}^{+} + \mathbf{d}_{\mathbf{x}}^{+} + \mathbf{d}_{\mathbf{s}}^{-} - \mathbf{d}_{\mathbf{k}}^{-})$$

  which means that the activatory phase will be followed by recovery of immunocompetence both if the ligand does not bind too strongly and positive signalling does not decay before inactivation of the kinases.

  This situation requires that the ligand residence time be in an optimal range and is related to a memory-type response.

- For positive signalling followed by anergy:

  $$(\mathbf{d}_{\mathbf{s}}^{-} < \mathbf{d}_{\mathbf{k}}^{-}) \quad \text{OR} \quad (\mathbf{d}_{\mathbf{k}}^{+} + \mathbf{d}_{\mathbf{x}}^{+} + \mathbf{d}_{\mathbf{s}}^{-} - \mathbf{d}_{\mathbf{k}}^{-} < \mathbf{d}_{\mathbf{b}}^{-})$$

  which means that the activatory phase will be followed by anergy if either both the positive signal decays rapidly or ligand dissociation is very slow. This situation corresponds to 'activation-induced anergy'.

  □ End of (Kaufman et al., 1999) quotation.

One can notice that the constraints, as shown in bold faced, in the HyTech Table 4 and in Kaufman et al. (1999) are exactly the same for all paths.

8 Cycles in BRN

The state graph of a BRN is not a simple tree like graph: it frequently contains cycles. It has been stated and then shown that cycles play a crucial role in the dynamics:

Table 4 HyTech results for the delay constraints of the transitions of paths shown in Figure 8

Transitions	*Paths*	*Constraints*
1000 → 0000	1	$\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+$
1000 → 1100	2–12	$\mathbf{d}_\mathbf{k}^+ \leq \mathbf{d}_\mathbf{b}^-$
1100 → 0100	2	$\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \wedge \mathbf{d}_\mathbf{b}^- \leq d_k^+ + d_x^+$
1100 → 1110	3	$\mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{s}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{b}^-$
1100 → 1101	4–12	$\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-$
1101 → 1111	4–7	$\mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{b}^-$
1101 → 0101	8–12	$\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+$
1111 → 1110	4	$d_k^+ + d_x^+ + d_s^- \leq d_b^-$
1111 → 0111	5–7	$d_b^- \leq d_k^+ + d_x^+ + d_s^-$
0111 → 0110	5	$d_k^+ + d_x^+ + d_s^- \leq d_b^- + d_k^-$
0111 → 0011	6–7	$\mathbf{d}_\mathbf{b}^- + \mathbf{d}_\mathbf{k}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ + \mathbf{d}_\mathbf{s}^-$
0011 → 0001	6	$d_b^- + d_k^- + d_x^- \leq d_k^+ + d_x^+ + d_s^-$
0011 → 0010	7	$d_k^+ + d_x^+ + d_s^- \leq d_b^- + d_k^- + d_x^-$
0101 → 0111	8–10	$d_k^+ + d_x^+ \leq d_b^- + d_k^- \wedge d_k^+ + d_x^+ \leq d_b^- + d_s^-$
0101 → 0100	11	$d_s^- \leq d_k^- \wedge d_b^- + d_s^- \wedge d_k^+ + d_x^+$
0101 → 0001	12	$d_k^- \leq d_s^- \wedge d_b^- + d_k^- \leq d_k^+ + d_x^+$
0111 → 0110	8	$\mathbf{d}_\mathbf{s}^- \leq \mathbf{d}_\mathbf{k}^-$
0111 → 0011	9–10	$\mathbf{d}_\mathbf{k}^- \leq \mathbf{d}_\mathbf{s}^-$
0011 → 0001	9	$d_k^- + d_x^- \leq d_s^-$
0011 → 0010	10	$d_s^- \leq d_k^- + d_x^-$

Path 1: 1000 → 0000 $\equiv \mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+$

Path 2: 1000 → 1100 → 0100 → 0110
$\equiv (\mathbf{d}_\mathbf{k}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \wedge \mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+)$

Path 3: 1000 → 1100 → 1110 → 0110
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{s}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{b}^-)$

Path 4: 1000 → 1100 → 1101 → 1111 → 1110 → 0110
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (d_k^+ + d_x^+ + d_s^- \leq d_b^-)$

Path 5: 1000 → 1100 → 1101 → 1111 → 0111 → 0110
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (d_b^- \leq d_k^+ + d_x^+ + d_s^-)$
$\wedge(\mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ + \mathbf{d}_\mathbf{s}^- \leq \mathbf{d}_\mathbf{b}^- + \mathbf{d}_\mathbf{k}^-)$

Path 6: 1000 → 1100 → 1101 → 1111 → 0111 → 0011 → 0001 → 0000
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{b}^-)$
$\wedge(d_b^- \leq d_k^+ + d_x^+ + d_s^-) \wedge (\mathbf{d}_\mathbf{b}^- + \mathbf{d}_\mathbf{k}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ + \mathbf{d}_\mathbf{s}^-) \wedge (d_b^- + d_k^- + d_x^- \leq d_k^+ + d_x^+ + d_s^-)$

Path 7: 1000 → 1100 → 1101 → 1111 → 0111 → 0011 → 0010 → 0000
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (d_b^- \leq d_k^+ + d_x^+ + d_s^-)$
$\wedge(\mathbf{d}_\mathbf{b}^- + \mathbf{d}_\mathbf{k}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+ + \mathbf{d}_\mathbf{s}^-) \wedge (d_k^+ + d_x^+ + d_s^- \leq d_b^- + d_k^- + d_x^-)$

Path 8: 1000 → 1100 → 1101 → 0101 → 0111 → 0110
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+) \wedge (d_k^+ + d_x^+ \leq d_b^- + d_k^-)$
$\wedge(d_k^+ + d_x^+ \leq d_b^- + d_s^-) \wedge (\mathbf{d}_\mathbf{s}^- \leq \mathbf{d}_\mathbf{k}^-)$

Path 9: 1000 → 1100 → 1101 → 0101 → 0111 → 0011 → 0001 → 0000
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+)$
$\wedge(d_k^+ + d_x^+ \leq d_b^- + d_k^- \wedge d_k^+ + d_x^+ \leq d_b^- + d_s^-) \wedge (\mathbf{d}_\mathbf{k}^- \leq \mathbf{d}_\mathbf{s}^-) \wedge (d_k^- + d_x^- \leq d_s^-)$

Path 10: 1000 → 1100 → 1101 → 0101 → 0111 → 0011 → 0010 → 0000
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+)$
$\wedge(d_k^+ + d_x^+ \leq d_b^- + d_k^- \wedge d_k^+ + d_x^+ \leq d_b^- + d_s^-) \wedge (\mathbf{d}_\mathbf{k}^- \leq \mathbf{d}_\mathbf{s}^-) \wedge (d_s^- \leq d_k^- + d_x^-)$

Path 11: 1000 → 1100 → 1101 → 0101 → 0100 → 0110
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+)$
$\wedge(d_s^- \leq d_k^- \wedge d_b^- + d_s^- \wedge d_k^+ + d_x^+)$

Path 12: 1000 → 1100 → 1101 → 0101 → 0001 → 0000
$\equiv (d_k^+ \leq d_b^-) \wedge (\mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{x}^+ \wedge \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{s}^+ \leq \mathbf{d}_\mathbf{b}^-) \wedge (\mathbf{d}_\mathbf{b}^- \leq \mathbf{d}_\mathbf{k}^+ + \mathbf{d}_\mathbf{x}^+)$
$\wedge(d_k^- \leq d_s^- \wedge d_b^- + d_k^- \leq d_k^+ + d_x^+)$

a positive circuit (resp. negative circuit) in the interaction graph is necessary to observe multistationarity (resp. homeostasis) (Thomas and Kaufman, 2001; Thomas et al., 1995; Cinquin and Demongeot, 2002; Soulé, 2003). For example, homeostasis is expressed in the state graph by oscillatory behaviours (sustained or not), which are often abstracted by cycles. Then it becomes important to analyse the entrance into such cycles. The delay constraints in Section 7 that can select a certain pathway can be synthesised in similar way to enter a cycle (Ahmad et al., 2006), as shown in Figure 3 by bold arrows. But introducing delays in BRN modelling can make the cycle to be stable or unstable, depending on values of these delays (Bernot et al., 2007; Thomas and D'Ari, 1990). To remain stable in the cycle, some initial conditions in terms of constraints of delay parameters and clocks have to be synthesised such that some timed trajectories from these initial conditions remain viable in a cycle. This is supposed to introduce the notion of invariance kernel Schneider (2004) that requires further modelling of BRN.

In real situations, however, the initial state is usually not in the cycle. It is therefore important to analyse which initial states can lead the system into the cycle, to determine the constraints on the time delays which will effectively allow the system to enter the cycle and then to verify that these constraints are compatible with the conditions for remaining in the cycle.

Fixing the values for delays is also an issue in the context of time delays. If nothing is said about the time delays the whole graph remains indeed open, but if one assigns values to the delays (e.g., on the basis of biological data), only one well-defined transition path will remain. Biological execution of such models are in fact subject to some variations due to slight differences between delay parameters chosen by different cells even in the homogeneous population (Thomas and D'Ari, 1990). Further analysis with probabilistic models of these variations on delay parameters would be able to separate different behaviours.

9 Conclusion

We propose in this paper a refinement of the concept of delays for BRN modelling. The introduction of delays allows one to distinguish paths from one state to another one. This refinement reintroduces time in the abstraction of R. Thomas, and this way is different from the refinement of Batt et al. (2005) and Adélaïde and Sutre (2004) which split the state space by partitioning the domains of the state space. The present work describes how the introduction of time can be helpful for modelling such networks, allowing the modeller to verify temporal properties. It is now important to confront this modelling with even more complex real systems where the underlying processes are not yet already known. Our experience in modelling in a multidisciplinary context will help to initiate biological modelling with delays.

In addition, we plane to deepen the following points:

- *parameter synthesis*: we have to check when a path is equivalent to an empty region and what this really means
- *cycles*: we have to look for invariance kernels in the state graph of BRN.

Acknowledgements

We would like to acknowledge the many helpful suggestions of the anonymous reviewers on previous versions of this paper.

References

Adélaïde, M. and Sutre, G. (2004) 'Parametric analysis and abstraction of genetic regulatory networks', *Proc. 2nd Workshop on Concurrent Models in Molecular Biology (BioCONCUR'04)*, London, UK, August, ser. Electronic Notes in Theor. Comp. Sci., Elsevier, London, UK, (To appear).

Ahmad, J., Bernot, G., Comet, J-P., Lime, D. and Roux, O. (2006) 'Hybrid modeling and dynamical analysis of gene regulatory networks with delays', *ComPlexUs*, Vol. 3, No. 4, pp.231–251.

Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P-H., Nicollin, X., Olivero, A., Sifakis, J. and Yovine, S. (1995) 'The algorithmic analysis of hybrid systems', *Theoretical Computer Science*, Vol. 138, No. 1, pp.3–34, [Online], Available: citeseer.ist.psu.edu/alur95algorithmic.html

Alur, R., Courcoubetis, C., Henzinger, T.A. and Ho, P-H. (1992) 'Hybrid automata: an algorithmic approach to the specification and verification of hybrid systems', *Hybrid Systems*, pp.209–229, [Online], Available: citeseer.ist.psu.edu/alur92hybrid.html

Batt, G., Ropers, D., de Jong, H., Geiselmann, J., Page, M. and Schneider, D. (2005) 'Qualitative analysis and verification of hybrid models of genetic regulatory networks: nutritional stress response in Escherichia coli', in Morari, M. and Thiele, L. (Eds.): *Eighth International Workshop on Hybrid Systems: Computation and Control, HSCC 2005*, Ser. Lecture Notes in Computer Science, Vol. 3414, Springer, pp.134–150.

Bernot, G., Cassez, F., Comet, J-P., Delaplace, F., Müller, C., Roux, O. and Roux, O. (2007) 'Semantics of biological regulatory networks', *Proc. First Workshop on Concurrent Models in Molecular Biology (BioConcur 2003)*, Marseille, France, ser. Electronic Notes in Theor. Comp. Sci., Elsevier, Vol. 180, No. 3, pp.3–14.

Bernot, G., Comet, J-P., Richard, A. and Guespin, J. (2004) 'Application of formal methods to biological regulatory networks: extending Thomas' asynchronous logical approach with temporal logic', *Journal of Theoretical Biology*, Vol. 229, No. 3, pp.339–347.

Cassez, F. and Larsen, K.G. (2000) 'The impressive power of stopwatches', *CONCUR'00: Proceedings of the 11th International Conference on Concurrency Theory*, Springer-Verlag, London, UK, pp.138–152.

Cinquin, O. and Demongeot, J. (2002) 'Positive and negative feedback: striking a balance between necessary antagonists, *Journal of Theoretical Biology*, Vol. 216, No. 2, pp.229–241.

de Jong, H. (2002) 'Modeling and simulation of genetic regulatory systems: a literature review', *J. Comput. Biol.*, Vol. 9, No. 1, pp.67–103.

de Jong, H., Geiselmann, J., Batt, G., Hernandez, C. and Page, M. (2004) 'Qualitative simulation of the initiation of sporulation in Bacillus subtilis', *Bulletin of Mathematical Biology*, Vol. 66, No. 2, pp.261–299.

de Jong, H., Geiselmann, J., Hernandez, C. and Page, M. (2003) 'Genetic network analyzer: qualitative simulation of genetic regulatory networks', *Bioinformatics*, Vol. 19, No. 3, pp.336–344.

Devloo, V., Hansen, P. and Labbé, M. (2003) 'Identification of all steady states in large networks by logical analysis', *Bull. Math. Biol.*, Vol. 65, No. 6, pp.1025–1051.

Henzinger, T-A., Ho, P-H. and Wong-Toi, H. (1997) 'HYTECH: a model checker for hybrid systems', *International Journal on Software Tools for Technology Transfer*, Vol. 1, Nos. 1–2, pp.110–122, [Online], Available: citeseer.ist.psu.edu/henzinger97hytech.html

Hill, C., Sawhill, B., Kauffman, S.A. and Glass, L. (1998) 'Transition to chaos in models of genetic networks', in *Statistical Mechanics of Biocomplexity: Proceedings of the XV Sitges Conference Held at Sitges*, Barcelona, Spain, ser. Lecture Notes in Physics, Springer, Berlin/Heidelberg, Vol. 527, pp.261–274.

Kam, N., Cohen, I.R. and Harel, D. (2001) 'The immune system as a reactive system: modeling T cell activation with statecharts', *Proceedings of the IEEE 2001 Symposia on Human Centric Computing Languages and Environments (HCC'01)*, 05–07 September, p.15.

Kauffman, S. (2003) 'Understanding genetic regulatory networks', *International Journal of Astrobiology*, Vol. 2, April, pp.131–139.

Kaufman, M., Andris, F. and Leo, O. (1999) 'A logical analysis of T cell activation and anergy', *Proc. Natl. Acad. Sci. U.S.A.*, pp.3894–3899.

Schneider, G. (2004) 'Computing invariance kernels of polygonal hybrid systems', *Nordic Journal of Computing*, Vol. 11, No. 2, pp.194–210.

Snoussi, E. (1989) 'Qualitative dynamics of a piecewise-linear differential equations: a discrete mapping approach', DSS, Vol. 4, pp.189–207.

Snoussi, E. and Thomas, R. (1993) 'Logical identification of all steady states : the concept of feedback loop characteristic states', *Bull. Math. Biol.*, Vol. 55, No. 5, pp.973–991.

Soulé, C. (2003) 'Graphic requirements for multistationarity', *Complexus*, Vol. 1, pp.123–133.

Thieffry, D. (1993) *Modélisation des régulations Génétiques: une méthode logique, son automatisation et quelques applications*, PhD Dissertation, Theoretical Biology Unit, Université Libre de Bruxelles, Belgium, 23 January.

Thomas, A.P., Henzinger, A., Kopke, P.W. and Varaiya, P. (1998) *What's Decidable about Hybrid Automata*, EECS Department, University of California, Berkeley, Tech. Rep., [Online], Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/1998/3418.html

Thomas, R. (1973) 'Boolean formalization of genetic control circuits', *Journal of Theoretical Biology*, Vol. 425, pp.563–585.

Thomas, R. (1978) 'Logical analysis of systems comprising feedback loops', *J. Theor. Biol.*, Vol. 73, No. 4, pp.631–656.

Thomas, R. (1979) 'Kinetic logic: a Boolean approach to the analysis of complex regulatory systems', *Lecture Notes in Biomathematics*, Vol. 29, p.507.

Thomas, R. (1991) 'Regulatory networks seen as asynchronous automata: a logical description', *J. Theor. Biol.*, Vol. 153, pp.1–23.

Thomas, R. and D'Ari, R. (1990) *Biological Feedback*, CRC Press, Boca Raton, FL.

Thomas, R. and Kaufman, M. (2001) 'Multistationarity, the basis of cell differentiation and memory', *Chaos*, Vol. 11, pp.180–195.

Thomas, R., Thieffry, D. and Kaufman, M. (1995) 'Dynamical behaviour of biological regulatory networks-I', *Bull. Math. Biol.*, Vol. 57, No. 2, pp.9247–276.

Note

[1] In Figure 6, this cycle oscillates around a particular point named $0\theta00$. Theta generally corresponds to a threshold above which (resp. below which) the concentration of `Cro` tends to decrease (resp. increase).

Appendix A

In this appendix, we show the HyTech file for the example of Section 4. The HyTech file consists of two parts: the hybrid automaton and the analysis commands.

A.1 Hybrid automaton

The following HyTech codes implement the stopwatch automaton of Figure 4. Here, the variables ha, hb and hc, represent the clocks associated to genes a, b and c respectively. These clocks evolve towards delay parameters dpa, dna, dpb, dnb, dpc and dnc during the activation and inhibition periods of their respective genes.

```
    --/ HyTech file: example.hy --/ BRN:
    (a)---(+)--->(b)---(+)--->(c)---(+)--->(a)
     --/ Variable declarations
    var
     ha,hb,hc : analog;
     state : discrete;
     dpa,dna,dpb,dnb,dpc,dnc : parameter;
     --/ State transition system
    automaton auto synclabs : ;
    initially conf_010;
     --/ for the configuration [0, 0, 0]
     loc conf_000: while True wait {dha=0,dhb=0, dhc=0}
    --/ for the configuration [0, 0, 1]
     loc conf_001: while  ha<=dpa & hc<=dnc wait {dha=1, dhb=0, dhc=1}
     when ha=dpa do {ha'=0,state'=5} goto conf_101;
     when hc=dnc do {hc'=0,state'=0} goto conf_000;
--/ for the configuration [0, 1, 0]
 loc conf_010: while  hb<=dnb & hc<=dpc  wait {dha=0, dhb=1, dhc=1}
     when hb=dnb do {hb'=0,state'=0} goto conf_000;
     when hc=dpc do {hc'=0,state'=3} goto conf_011;
--/ for the configuration [0, 1, 1]
 loc conf_011: while  ha<=dpa & hb<=dnb  wait {dha=1, dhb=1, dhc=0}
     when ha=dpa do {ha'=0,state'=7} goto conf_111;
     when hb=dnb do {hb'=0,state'=1} goto conf_001;
--/ for the configuration [1, 0, 0]
 loc conf_100: while  ha<=dna & hb<=dpb wait {dha=1, dhb=1, dhc=0}
     when ha=dna do {ha'=0,state'=0} goto conf_000;
     when hb=dpb do {hb'=0,state'=6} goto conf_110;
--/ for the configuration [1, 0, 1]
 loc conf_101: while hb<=dpb & hc<=dnc wait {dha=0, dhb=1, dhc=1}
     when hb=dpb do {hb'=0,state'=7} goto conf_111;
     when hc=dnc do {hc'=0,state'=4} goto conf_100;
--/ for the configuration [1, 1, 0]
 loc conf_110: while  ha<=dna & hc<=dpc wait {dha=1, dhb=0, dhc=1}
     when ha=dna do {ha'=0,state'=2} goto conf_010;
     when hc=dpc do {hc'=0,state'=7} goto conf_111;
--/ for the configuration [1, 1, 1]
 loc conf_111: while True wait {dha=0, dhb=0, dhc=0}
end
```

A.2 Analysis commands

The following HyTech codes implement Algorithm 1 of Section 6.

```
-- The operators "&" and "˜" represent the conjunction and negation
--operations respectively. --/Region declarations
 var
 ini_reg, fin_reg, first_state, final_state, path1, path2, path3,
 states_accumulated, path_states, cons, reached, reached2, reached3,
access, X, Y, Z, bogus: region;
 --/ To find the path between two states first modify the
--/  variables first_state, final_state, ir and fr
 first_state:=state=2;
 final_state:=state=7;
--initial region
 ini_reg:= loc[auto] = conf_010 & ha=0 & hb=0 & hc=0;
 -- final region
 fin_reg:= loc[auto] = conf_111 &  final_state;
--/access all reachable states
access:=post(post(post(post(post(post(post(post(post(ini_reg)))))))));
prints "====================================================";
prints "Accessible states from the initial state along with the
parameter constraints"; print hide non_parameters in access endhide;
prints "====================================================";
--/Path(first_state, final_state)
 reached:= first_state;
 path1:= first_state;
 states_accumulated:=path1;
 --/ The outer loop starts here
X:= iterate X from first_state using{
 X:=  post(X & access) & ˜X; --/access the successors of X
 path1:=X & ˜path1;
 states_accumulated:=(states_accumulated | path1);
 states_accumulated:=(states_accumulated & ˜first_state) ;
 X:=weakdiff(X,reached);
 reached:=reached | X;
  if empty(path1 & final_state)
  then bogus:=first_state;
 else
  reached2:=final_state;
  path2:=final_state;
  path_states:=path2;
  --/ The first inner loop starts here
  Y:= iterate Y from  final_state using{

  --/access the predecessors of Y in states_accumulated
 Y:= pre(Y & access) & ˜Y & states_accumulated;
 path2:=Y & ˜path2;
 path_states:=(path_states | path2)   ;
 Y:=weakdiff(Y, reached2);
 reached2:=reached2 | Y;
}; --/ The first inner loop ends here
--print_path(path_states)
reached3:= first_state;
path3 := first_state;
Z:=path3;
prints "=======================================";

prints "Path=";
--/ First state of the path
print hide non_parameters in  ini_reg & access endhide;
--/ The second inner loop starts here
Z:= iterate Z from first_state using{
    --/access the successors of Z in path_states
     Z:= post(Z & access) & ~Z & path_states;
     path3:=Z & ~ path3;
     path3:=path3 & ~ final_state;
     path3:=path3 & ~ first_state;
     path3:=hull(path3);
     print hide non_parameters in  path3 & access endhide;
     Z:=weakdiff(Z, reached3);
     reached3:=reached3 | Z;
    };--/ The second inner loop ends here
    --/ Final state of the path
    print hide non_parameters in  fin_reg & access endhide;
   endif;
  }; --/ The outer loop ends here
```