



# Incorporating Time Delays in Process Hitting Framework for Dynamical Modeling of Large Biological Regulatory Networks

Iftikhar Ali Sheikh<sup>1</sup>, Jamil Ahmad<sup>1,2\*</sup>, Morgan Magnin<sup>3</sup> and Olivier Roux<sup>3</sup>

<sup>1</sup> Research Centre for Modeling and Simulation, National University of Sciences and Technology, Islamabad, Pakistan,

<sup>2</sup> Department of Computer Science and Information Technology, University of Malakand, Chakdara, Pakistan, <sup>3</sup> Laboratory of Digital Sciences of Nantes (LS2N), UMR CNRS 6004, Ecole Centrale de Nantes, Nantes, France

## OPEN ACCESS

### Edited by:

Luis Mendoza,  
National Autonomous University of  
Mexico, Mexico

### Reviewed by:

Yilun Shang,  
Northumbria University,  
United Kingdom  
Mariana Esther Martinez-Sanchez,  
National Autonomous University of  
Mexico, Mexico

### \*Correspondence:

Jamil Ahmad  
jamil.ahmad@rcms.nust.edu.pk;  
jamil.ahmad@uom.edu.pk

### Specialty section:

This article was submitted to  
Systems Biology,  
a section of the journal  
Frontiers in Physiology

**Received:** 23 November 2018

**Accepted:** 25 January 2019

**Published:** 15 February 2019

### Citation:

Sheikh IA, Ahmad J, Magnin M and  
Roux O (2019) Incorporating Time  
Delays in Process Hitting Framework  
for Dynamical Modeling of Large  
Biological Regulatory Networks.  
*Front. Physiol.* 10:90.  
doi: 10.3389/fphys.2019.00090

Modeling and simulation of molecular systems helps in understanding the behavioral mechanism of biological regulation. Time delays in production and degradation of expressions are important parameters in biological regulation. Constraints on time delays provide insight into the dynamical behavior of a Biological Regulatory Network (BRN). A recently introduced Process Hitting (PH) Framework has been found efficient in static analysis of large BRNs, however, it lacks the inference of time delays and thus determination of their constraints associated with the evolution of the expression levels of biological entities of BRN is not possible. In this paper we propose a *Hybrid Process Hitting* scheme for introducing time delays in Process Hitting Framework for dynamical modeling and analysis of Large Biological Regulatory Networks. It provides valuable insights into the time delays corresponding to the changes in the expression levels of biological entities thus possibly helping in identification of therapeutic targets. The proposed framework is applied to a well-known BRNs of *Bacteriophage λ* and ERBB Receptor-regulated G1/S transition involved in the breast cancer to demonstrate the viability of our approach. Using the proposed approach, we are able to perform goal-oriented reduction of the BRN and also determine the constraints on time delays characterizing the evolution (dynamics) of the reduced BRN.

**Keywords:** process hitting, logical modeling, René Thomas framework, kinetic logic, time delays, hybrid modeling, systems biology, biological regulatory networks

## 1. INTRODUCTION

Biological Regulatory Networks (BRNs) are used to describe almost all biological functions to cover the interactions taking place in various chemical reactions in living organisms. The nature of these interactions is continuous and stochastic in nature and the rate of change in the underlying dynamics of these interactions is variable. A large number of formal approaches have been devised to model the topology of BRNs and to analyse their dynamical behaviors (De Jong, 2002; Sheikh et al., 2016). BRNs are traditionally modeled by differential equations but they are non-linear and thus difficult to solve analytically. Moreover, the available experimental data is mostly qualitative in nature and do not aid in precise determination of quantitative parameters for the differential equations. A powerful tool involving Lie algebra has been reported to solve a large class of non-linear epidemic spreading systems analytically (Shang, 2012, 2015a), however it

has not been applied to the dynamical modeling of BRNs. René Thomas Discrete Modeling technique (Thomas, 1978) proposed a qualitative method for representing the change in expression level of a gene or biological entity with a logical function having discrete values. The information on whether the gene expression would increase or decay is contained in the  $K$ -parameters associated with each entity of the BRN. The values of these parameters determine the dynamical behavior of the system that is represented by a State Transition Graph (STG). The state graph grows exponentially with the increase in the number of entities thus it becomes difficult to analyse medium or large scale BRNs (containing more than 10 or more genes). Moreover, this approach ignores the time taken for the gene to reach the expression level required for production or degradation and therefore, uses an asynchronous evolution of the dynamics of the system. Hybrid Modeling Technique (Ahmad et al., 2006) addressed this shortcoming by associating time delays in BRNs by considering the change in qualitative levels as piece-wise linear functions. This made it possible to perform model checking and obtain constraints in terms of production and/or degradation delays. The hybrid approach works well for BRNs consisting up to 7–8 genes, however, it is unable to compute the time delays for larger networks due to the state space explosion.

A Process Hitting framework for analysis of large BRNs has been proposed (Paulevé et al., 2011; Folschette et al., 2012) which can handle networks comprising of thousands of genes. The very high scalability of this approach is due to the following factors:

- It takes into consideration only the most permissible (generalized) dynamics possible in the interaction graph of a BRN instead of dealing with the whole state space.
- The generalized dynamics is refined with the help of cooperativity between two or more genes, which have a combined influence on any other gene in the network.

The Process Hitting approach is thus able to quickly perform static analysis like determination of stable states, successive reachability and inferring the  $K$ -parameters of the BRN. This technique is based on Stochastic  $\pi$ -Calculus and allows for synthesis of temporal and stochastic parameters, which enables the simulation of dynamical behavior to some extent. However, there is no mechanism for incorporating the time delays in the proposed framework. Lately, time parameters have been introduced into Process Hitting with classes of priorities (Folschette et al., 2015); however, it still does not infer time delays of the interactions.

The Process Hitting is also able to perform goal-oriented reduction of the large BRNs by taking into consideration the minimal traces that are necessary to achieve the desired reachability goal. It applies the cutsets to preserve the minimal traces through Gene Knock-out/in/down technique (Paulevé et al., 2013).

Previously, enhanced modeling of BRNs based on timed automata has been performed in Siebert and Bockmayr (2008) which introduced time delays in the René Thomas Discrete Modelling Framework and it is possible to simulate the dynamics of BRN. However, this approach does not infer the constraints on time delays rather the numerical values have to be adjusted

manually to obtain a certain behavior. Moreover, it is also limited to small BRNs as it introduces intermediate states, which results in an even greater number of possible states in the state transition graph.

Recently, new methodology has been proposed for the inference of BRNs through a time extension of Automata Networks using the time series data and known influences among the genes (Ben Abdallah et al., 2017). This modeling approach requires the observed experimental time series data for deducing the BRN model which satisfies the dynamical behavior depicted by the observed data.

Our approach is to extend the Process Hitting Framework by introducing *Time Delays* in it so that it becomes possible to determine the constraints on the activation and degradation delays associated with the evolution of a gene in the dynamical model of the BRN. In order to achieve it, we represent the biological entities of the BRN with Biological Linear Hybrid Automata (Bio-LHA) (Ahmad et al., 2009; Aslam et al., 2014) and enrich them with time delays while the logical rules for gene activation or degradation are provided by the cooperative hits as determined by Process Hitting. Since PH only takes into consideration the permissible (possible) dynamics of the BRN as given in its Interaction Graph (IG), therefore, we have a reduced STG to be represented by the Hybrid Automata.

The proposed approach, *Hybrid Process Hitting*, has been applied to a toy BRN to fully explain the steps involved and subsequently applied to the biologically well-known network of *Bacteriophage  $\lambda$*  and ERBB Receptor-regulated G1/S transition involved in the breast cancer to successfully determine the constraints on Time Delays in less computational time. Moreover, due to its reduced complexity, the proposed approach would be able to handle BRNs with more number of genes as compared to the Hybrid Modeling approach that takes into account the complete STG while determining the constraints on Time Delays.

The rest of the paper is organized as follows: section 2 discusses the Process Hitting Framework while the basics of René Thomas Discrete Modelling Framework and Hybrid Modeling are described in section 3. The proposed methodology for incorporation of Time Delays in Process Hitting Framework is given in section 4 and the proposed *Hybrid Process Hitting* approach is applied to the BRNs of *Bacteriophage  $\lambda$*  and ERBB-receptor regulated G1/S transition in sections 5, 6 followed by concluding remarks and future work in sections 7, 8, respectively. A glossary of mathematical notations used in the definitions in this paper is provided as supplementary file (**Data Sheet 2**).

## 2. PROCESS HITTING (PH) FRAMEWORK

This section describes the Process Hitting Framework which is used for the static analysis of large scale biological regulatory networks (Paulevé et al., 2011; Folschette et al., 2012) and Goal-oriented model reduction.

### 2.1. Biological Regulatory Networks (BRNs)

BRNs are used to model the interactions between various biological entities (Proteins or RNA). Quite a few complex

processes are involved in the regulations between them, however, these are simplified to only two actions, namely, activation and inhibition (Thomas and d'Ari, 1990). BRNs are often described as an Interaction Graph in which genes or other biological entities are shown as nodes and their activating or inhibiting influences on the other elements are represented by positive or negative edges, respectively. The simple representation of BRN was extended to automatically derive its behavioral model by using model-checking tools (Bernot et al., 2007). The discrete levels are defined for a particular concentration level of a gene after which its influence on other entities in the BRN changes. The *activating* (resp. *inhibiting*) influence of a gene is exerted once it is *above* (resp. *below*) the threshold level.

**Definition 1. (Interaction Graph).** An Interaction Graph  $IG = (N, E)$  of a Biological Regulatory Network is defined where  $N$  is the set of all nodes and  $E$  is the set of all directed edges in which an edge from node  $p$  to  $q$  is  $e = (p \xrightarrow{t,s} q)$  and is labelled by the pair  $(t, s)$  such that  $t$  is a positive integer and represents the qualitative threshold level required for interaction, and  $s \in \{+, -\}$  describes the type of interaction, i.e., “+” for activation and “-” for inhibition.

The interaction graph described above is quite similar with the discrete version of Deffuant model in opinion evolution in a network (Shang, 2015b).

**Definition 2. (Effective Levels).** For a regulation  $p \xrightarrow{t,s} q$  in a BRN and  $s$  being + (resp. -), the effective levels  $LEV_p^+$  are those threshold levels  $\{t; l_p\}$  (resp.  $\{0; t - 1\}$ ) of gene  $p$  at and above (resp. below) which it will activate (resp. inhibit) gene  $q$ , where  $l_p$  is the highest threshold level of gene  $p$ . Conversely, effective levels  $LEV_p^-$  would be  $\{0; t - 1\}$  (resp.  $\{t; l_p\}$ ) for which gene  $q$  would be activated by gene  $p$ .

**Example 1.** In the BRN shown in Figure 1B, the edge  $p \xrightarrow{1,+} q$  depicts the activating influence of gene  $p$  on gene  $q$  at threshold level 1. So effective levels  $LEV_p^+$  are  $\{1; 1\}$  and  $LEV_p^-$  are  $\{0; 0\}$  where gene  $p$  would be inhibiting gene  $q$ . Similarly, for some other interaction  $p \xrightarrow{2,-} q$  with  $l_p = 3$ , the effective levels  $LEV_p^+$  are  $\{0; 1\}$  and  $LEV_p^-$  are  $\{2; 3\}$ .

## 2.2. Automata Networks

Starting from any arbitrary state of the system, Process Hitting takes into account all the possible evolutions (Generalized Dynamics) of the BRN and keeps evolving the network till it reaches a steady state where no more interactions can take place. In Process Hitting Framework, each gene is represented as *automata* and its current level as *local state*. A Process Hitting state is formed by gathering one *local state* of each *automata* present in the BRN. A *local state*  $i$  of a *automata*  $p$  is denoted as  $p_i$ . At any given time for each automaton only one local state is present which represents the current level of activity of that gene. Accordingly, the set of all local states is the global state of the BRN. The local transition is the local possible evolutions inside an automaton. The transition  $p_i \xrightarrow{l} p_j$  is the change of local state  $p_i$  to local state  $p_j$  made possible by the presence of a set  $l$  of local states belonging to other automata which should be active in the current global state. Thus, all local states in  $l$  cooperate to switch the active local state of  $p$  from  $p_i$  to  $p_j$ . It is noted that  $l$

contains at most one local state of another automaton and none from automaton  $p$ ; it can also be empty. Automata Network is formally defined as:

**Definition 3. (Automata Network).** An Automata Network (AN) is a triple  $(\Sigma, L, \mathcal{H})$  where:

- $\Sigma = \{p, q, \dots\}$  is the finite set of automata,
- $L = \prod_{p \in \Sigma} L_p$  is the finite set of global states with  $L_p = \{p_0, \dots, p_{l_p}\}$  the finite set of local states of automata  $p \in \Sigma$  and  $l_p$  a positive integer. Also  $p \neq q \Rightarrow \forall (p_i, q_j) \in L_p \times L_q, p_i \neq q_j$ , and  $LS = \cup_{p \in \Sigma} L_p$  denotes the set of all the local states.
- for each  $p \in \Sigma, \mathcal{H}_p = \{p_i \xrightarrow{l} p_j \in L_p \times \wp(LS \setminus L_p) \times L_p \mid p_i \neq p_j\}$ , is the set of local transitions on automata  $p$ ;  $\mathcal{H} = \cup_{p \in \Sigma} \mathcal{H}_p$  is the set of all local transitions in the network.

**Definition 4. (Step).** For an AN  $(\Sigma, L, \mathcal{H})$ , a step  $\theta \subseteq \mathcal{H}$  is a subset of local transitions  $\mathcal{H}$  where, for each automaton  $a \in \Sigma$ , there is at most one local change in  $a$ .

For any local transition  $\theta = p_i \xrightarrow{\ell} p_j, p_i$  is the origin,  $\ell$  is the condition and  $p_j$  is the destination and, respectively, noted as  $\bullet\theta$ ,  $cond(\theta)$  and  $\theta^\bullet$ .

**Definition 5. (Trace).** For an AN  $(\Sigma, L, \mathcal{H})$  and a state  $l \in L$ , a trace  $\pi$  is a sequence of successive steps  $\theta$  which leads to a defined goal state  $l_T$ . The trace  $\pi$  is denoted as  $\theta_1 :: \theta_2 :: \dots$ .

Given an Automata Network  $(\Sigma, L, \mathcal{H})$  and a state  $l \in L$ , the goal state  $l_T \in L$  is reachable from  $l$  if there exists a trace  $\pi$  with  $\bullet\pi \subseteq l$  and  $l_T \in \pi^\bullet$ .

**Example 2.** Figure 1A shows the Automata Network of a toy BRN depicted in Figure 1B where:

- $\Sigma = \{p, q, r\}$ ,
- $L_p = \{p_0, p_1\}, L_q = \{q_0, q_1\}, L_r = \{r_0, r_1\}$  and
- the set of transitions as:  

$$\mathcal{H} = \{q_0 \xrightarrow{\{p_1\}} q_1, q_1 \xrightarrow{\{p_0\}} q_0, r_0 \xrightarrow{\{p_1, q_1\}} r_1, r_1 \xrightarrow{\{p_0\}} r_0, r_1 \xrightarrow{\{q_0\}} r_0\}.$$

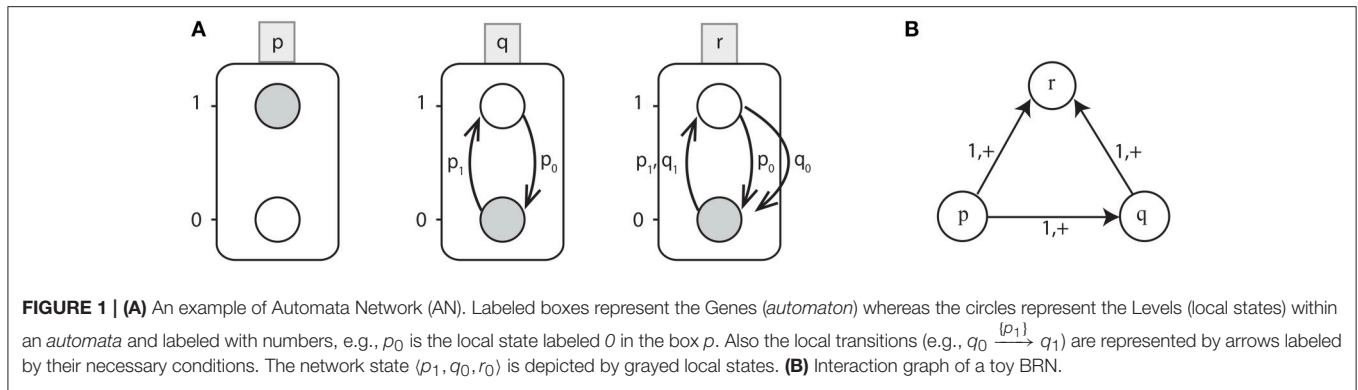
Starting from the state  $l = \langle p_1, q_0, r_0 \rangle$ , only one transition is playable:  $\theta = q_0 \xrightarrow{\{p_1\}} q_1$  which updates the automaton state to  $l \cdot \theta = \langle p_1, q_1, r_0 \rangle$ . From this state, the next possible transition is  $r_0 \xrightarrow{\{p_1, q_1\}} r_1$  which leads to the state  $\langle p_1, q_1, r_1 \rangle$ . Thus, from the state  $l = \langle p_1, q_0, r_0 \rangle$  to  $\langle p_1, q_1, r_1 \rangle$ , the trace  $\pi$  is given by:

$$\pi = \{q_0 \xrightarrow{\{p_1\}} q_1\} :: \{r_0 \xrightarrow{\{p_1, q_1\}} r_1\}.$$

## 2.3. Goal Oriented Reduction and Cutsets

In the Automata Networks representing the BRNs, a goal is typically the *activation* (resp. *inhibition*) of some gene or transcription factor or kinase, etc.. From an initial given state for each entity in BRN, the goal is reachable if a sequence of steps is present which leads to the state which contains the goal (Paulevé, 2018). For example, the goal in a signaling network is usually the activation of the downstream transcription factor starting from the initial inactive state.

A *minimal trace* corresponds to the path or sequence of steps (containing no cycle) starting from the initial state to the defined goal state containing the gene activation / inhibition. In a cycle, the initial global state is visited twice, i.e., the path leads back to its starting point. There can be multiple *traces* which satisfy the



**FIGURE 1 | (A)** An example of Automata Network (AN). Labeled boxes represent the Genes (*automaton*) whereas the circles represent the Levels (local states) within an *automata* and labeled with numbers, e.g.,  $p_0$  is the local state labeled 0 in the box  $p$ . Also the local transitions (e.g.,  $q_0 \xrightarrow{p_1} q_1$ ) are represented by arrows labeled by their necessary conditions. The network state  $(p_1, q_0, r_0)$  is depicted by grayed local states. **(B)** Interaction graph of a toy BRN.

goal reachability in the network. So, the goal oriented reduction aims to preserve all *minimal traces* to the defined goal.

For a global AN  $(\Sigma, L, \mathcal{H})$ , an initial state  $l \in L$  and a reachability goal state  $l_T$ , the goal oriented reduction identifies a subset of local transitions  $\mathcal{H}$  that are sufficient for producing all the minimal traces leading to  $l_T$  from  $l$ .

### 2.3.1. Cutsets for Goal Reachability

*Cutsets* are the sets of local states such that each trace reaching the goal includes a transition involving one of these local states. Also, these sets contain the necessary processes which when disabled would prevent the considered reachability goal. Hence, disabling of all transitions having precondition intersecting with cutset will remove all the traces leading to the goal. The algorithm for calculating the Cutset is described in detail in Paulevé et al. (2013) alongwith the complete methodology. An explanatory image describing the algorithm is given in Figure 2.

The *Cutsets* provide information on potential therapeutic targets if the considered reachability goal represents a diseased condition by preventing all the local states of a cutset to act using other Gene Knock-out/in/down techniques (Paulevé et al., 2013).

## 2.4. Regulation Conditions for Cooperative Hits

The combined effect or influence of multiple genes in a BRN is governed by logical functions (Richard et al., 2006; Bernot et al., 2008) unless there are biological observations which provide evidence that the genes are exerting an influence other than that given by the logical function. In that case the biological observation take precedence over the logical function.

Here we give all the possible cases for two genes  $p$  and  $q$  to cooperate and exert influence on a third gene  $r$  at threshold levels  $t_p$  and  $t_q$ , respectively, and define the rules for logical functions for activation (upregulation) of gene  $r$  when both  $p$  and  $q$  are acting simultaneously. Figure 3 depicts the four cases which cover all possible combinations of two entities which are  $\neg p \wedge \neg q$ ,  $\neg p \wedge q$ ,  $p \wedge \neg q$  and  $p \wedge q$ .

**Definition 6. (Regulation Condition for Individual Hits).** The regulation condition  $Cond_{ACT}$  (resp.  $Cond_{INH}$ ) for the individual hit contained in the set  $\mathcal{H}$  by a gene  $p$  having interaction with gene  $r$  as  $(p \xrightarrow{t_p, s_p} r)$  is defined as:

$$cond_{ACT} = \begin{cases} p_L, & s_p = \{-\} \\ p_H, & s_p = \{+\} \end{cases} \text{ and}$$

$$cond_{INH} = \begin{cases} p_H, & s_p = \{-\} \\ p_L, & s_p = \{+\} \end{cases}$$

Now we consider the combined influence of two genes  $p$  and  $q$  on one single entity  $r$  and define the rule for this cooperative interaction. If both the individual influencers  $p$  and  $q$  are the “inhibitors” of target gene  $r$ , then it will be *activated* when individually both  $p$  and  $q$  are at  $p_L$  and  $q_L$  represented by  $pq_{LL}$ . For all other combinations of  $p$  and  $q$  given by  $pq_{LH}$ ,  $pq_{HL}$ ,  $pq_{HH}$ , the target gene  $r$  would be *inhibited*.

**Definition 7. (Regulation Condition for Cooperative Hits).**

The regulation condition  $Cond_{ACT}$  (resp.  $Cond_{INH}$ ) for the cooperative hit  $\mathcal{H}$  by two genes  $p$  and  $q$  having their individual interactions with gene  $r$  as  $(p \xrightarrow{t_p, s_p} r)$  and  $(q \xrightarrow{t_q, s_q} r)$  respectively is defined as:

$$cond_{ACT} = \begin{cases} pq_{LL}, & s_p = \{-\}, s_q = \{-\} \\ pq_{LH}, & s_p = \{-\}, s_q = \{+\} \\ pq_{HL}, & s_p = \{+\}, s_q = \{-\} \\ pq_{HH}, & s_p = \{+\}, s_q = \{+\} \end{cases}$$

and

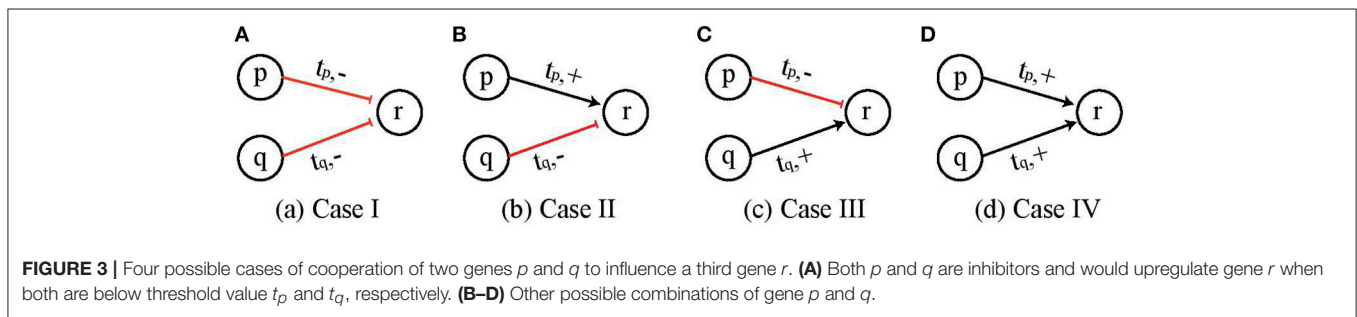
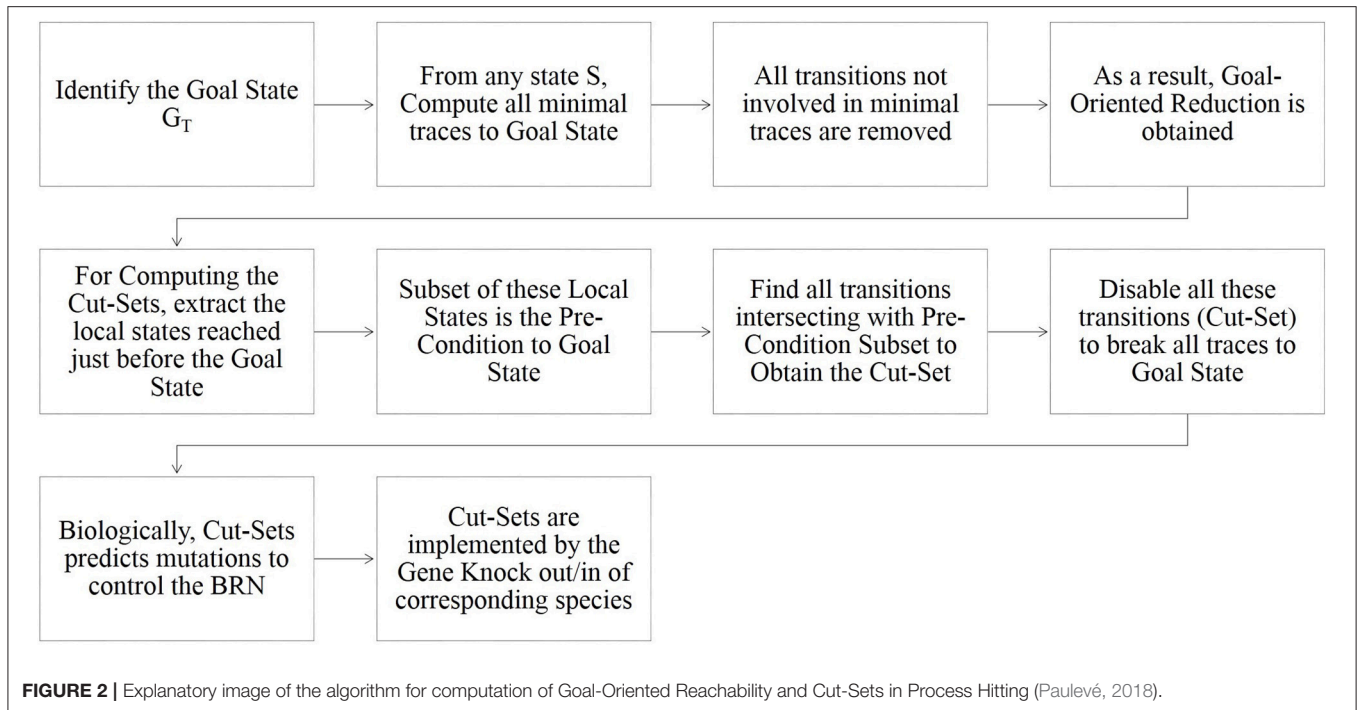
$$cond_{INH} = \begin{cases} pq_{LH}, pq_{HL}, pq_{HH}, & s_p = \{-\}, s_q = \{-\} \\ pq_{LL}, pq_{HL}, pq_{HH}, & s_p = \{-\}, s_q = \{+\} \\ pq_{LL}, pq_{LH}, pq_{HH}, & s_p = \{+\}, s_q = \{-\} \\ pq_{LL}, pq_{LH}, pq_{HL}, & s_p = \{+\}, s_q = \{+\} \end{cases}$$

We see that for Case I ( $\neg p \wedge \neg q$ ) shown in Figure 3A, the logical AND function implies that  $q$  would be upregulated when both  $p$  and  $q$  are below threshold  $t_p$  and  $t_q$ , respectively, which is represented by  $pq_{LL}$  and it would be downregulated for all other values, i.e.,  $pq_{LH}$ ,  $pq_{HL}$ , and  $pq_{HH}$ . Once the appropriate regulation condition has been obtained then the corresponding effective levels are determined in a straight forward manner. For  $Cond_{ACT} = pq_{LL}$  the effective levels are found to be  $LEV_p^- = \{0; t_p - 1\}$  and  $LEV_q^- = \{0; t_q - 1\}$ . Similarly, the conditions and effective levels for *activation* (resp. *inhibition*) for Cases II, III and IV are given in Table 1.

### 2.4.1. Multi-Level Genes

Now we consider the interactions involving multi-level genes which form a cooperative sort. For instance, consider gene  $p$  and gene  $q$  to have 3 and 4 threshold levels, respectively (i.e.,  $l_p = 1$





and  $l_q = 2$ ), and they interact with gene  $r$  at threshold level  $t_p = 1$  and  $t_q = 2$ , respectively, as shown in **Figure 4A**. Since the individual genes activate (*resp.* inhibit) the target gene for all the discrete levels above (*resp.* below) the threshold level, therefore, this interaction can be represented by a Boolean function with  $H$  containing all the discrete levels required for activation and  $L$  having levels for inhibition. Then for the case  $\neg p \wedge q$  which is the logical function for upregulation of gene  $r$ , the corresponding  $Cond_{ACT}$  is  $pq_{LH}$  which gives us the effective levels  $LEV_p^- < 1 = \{0; 0\}$  and  $LEV_q^+ \geq 2 = \{2; 2\}$ . **Figure 4B** shows the AN for gene  $r$  on which multi-level genes  $p$  and  $q$  are acting at the same time. The corresponding condition for upregulation for gene  $r$  requires  $p_L = \{p < 1\}$  and  $q_H = \{q \geq 2\}$ . Similarly, the downregulation takes place when either  $p_H = \{p \geq 1\}$  or  $q_L = \{q < 2\}$ .

### 2.5. Abstraction of Switch Conditions for Gene Regulation

We intend to model the Process Hitting in a BRN with the help of concurrent finite timed automata (Alur, 1999) in which the

qualitative levels of each gene would be represented by the states of its corresponding automaton and the gene evolution would be determined by the automaton dynamics. For this we need to abstract the switch conditions under which the automaton would move from one state to another.

Our approach is to directly derive the switch conditions from the effective levels corresponding to each action contained in the set  $\mathcal{H}$ . In order to model the timed automaton representing the dynamics of gene  $r$ , we abstract the switch conditions which are required for it to move from one state to another. These switch conditions are abstracted directly from the Regulation conditions and effective levels.

**Definition 8. (Hit Part).** We abstract the hit part  $\mu$  from the permissible process hits contained in  $\mathcal{H}$  corresponding to up (*resp.* down) regulation of each gene in the following manner:

$$\mu = \begin{cases} p_i, & i \in \{L, H\}, \quad \text{for hit by single gene} \\ p_{xy}, & xy \in \{LL, LH, HL, HH\}, \quad \text{for cooperative hit} \end{cases}$$

where  $L = [0; t_p - 1]$  and  $H = [t_p; l_p]$ .

Corresponding to each hit part  $\mu$ , we obtain the qualitative threshold levels which permit the change in expression level of

**TABLE 1** | The regulation conditions and effective levels for Up (resp. Down) Regulation by two genes  $p$  and  $q$  having a combined influence on another gene in a BRN are governed by the logical function.

Cases	Logical function	Activation		Inhibition	
		Cond <sub>ACT</sub>	Effective levels	Cond <sub>INH</sub>	Effective levels
I	$\neg p \wedge \neg q$	$pq_{LL}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^- = \{0; t_q - 1\}$	$pq_{LH}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^+ = \{t_q; l_q\}$
				$pq_{HL}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^- = \{0; t_q - 1\}$
				$pq_{HH}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^+ = \{t_q; l_q\}$
II	$\neg p \wedge q$	$pq_{LH}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^+ = \{t_q; l_q\}$	$pq_{LL}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^- = \{0; t_q - 1\}$
				$pq_{HL}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^- = \{0; t_q - 1\}$
				$pq_{HH}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^+ = \{t_q; l_q\}$
III	$p \wedge \neg q$	$pq_{HL}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^- = \{0; t_q - 1\}$	$pq_{LL}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^- = \{0; t_q - 1\}$
				$pq_{LH}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^+ = \{t_q; l_q\}$
				$pq_{HH}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^+ = \{t_q; l_q\}$
IV	$p \wedge q$	$pq_{HH}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^+ = \{t_q; l_q\}$	$pq_{LL}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^- = \{0; t_q - 1\}$
				$pq_{LH}$	$LEV_p^- = \{0; t_p - 1\}$ $LEV_q^+ = \{t_q; l_q\}$
				$pq_{HL}$	$LEV_p^+ = \{t_p; l_p\}$ $LEV_q^- = \{0; t_q - 1\}$

All the four possible functions are listed.

the target gene. From these levels we deduce the appropriate switch conditions  $\wedge$  which result in the dynamics of the target gene.

**Definition 9. (Switch Conditions for Gene Regulation).** We define Switch Conditions  $\wedge$  for each process hit contained in the set  $\mathcal{H}$  as below:

- for every hit by a single gene  $\mu = p_i, i \in \{L, H\}$ , the switch condition is:

$$\wedge = \begin{cases} p = \{0, \dots, t_p - 1\}, & \text{for } i = L \\ p = \{t_p, \dots, l_p\}, & \text{for } i = H \end{cases}$$

- for every cooperative hit  $\mu = p_{xy}, xy \in \{LL, LH, HL, HH\}$ , the switch condition is:

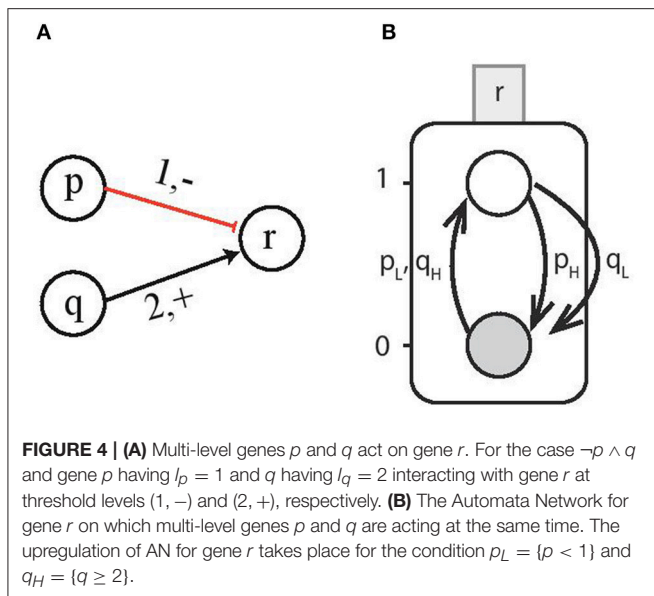
$$\wedge = \begin{cases} p = \{0, \dots, t_p - 1\}, q = \{0, \dots, t_q - 1\}, & \text{for } x = L, y = L \\ p = \{0, \dots, t_p - 1\}, q = \{t_q, \dots, l_q\}, & \text{for } x = L, y = H \\ p = \{t_p, \dots, l_p\}, q = \{0, \dots, t_q - 1\}, & \text{for } x = H, y = L \\ p = \{t_p, \dots, l_p\}, q = \{t_q, \dots, l_q\}, & \text{for } x = H, y = H \end{cases}$$

**Example 3.** For the transition  $r_0 \xrightarrow{p_0, q_1} r_1$  given in Example 3 in which the gene  $r$  is upregulated from  $r_0$  to  $r_1$  in the presence of  $p_0$  and  $q_1$ , the effective levels are obtained as  $LEV_p^- = 0$  and  $LEV_q^+ = 1$ . We see that gene  $r$  can change its state from  $r_0$  to  $r_1$  only when gene  $p$  and  $q$  are at threshold levels 0 and 1, respectively. Therefore, the appropriate switch condition is derived as  $\wedge : \{p = 0, q = 1\}$ . For the multi-level genes shown in Figure 4B, the effective levels for regulation condition  $Cond_{ACT}$  of gene  $r$  were obtained as  $LEV_p^- < 1 = \{0; 0\}$  and  $LEV_q^+ \geq 2 = \{2; 3\}$ . Thus, in this case the switch conditions would be  $\wedge : \{(p = 0, q = 2) \vee (p = 0, q = 3)\}$  which are all the combinations of effective levels of genes  $p$  and  $q$  at which gene  $r$  is upregulated.

### 3. HYBRID MODELING

#### 3.1. René Thomas' Logic Formalism

Initially, René Thomas presented a qualitative formalism based on Boolean logic applicable on Biological Regulatory Networks (Thomas, 1978, 2013) which closely approximated the ODE models. Later, the boolean model was found unable to



represent various interactions taking place in BRNs at varying gene expression concentrations. This led to the presentation of kinetic logic formalism which allows the modeling of discretely abstracted concentration levels other than boolean as well (Thomas and d’Ari, 1990). It has been further enriched with Parametric Time Delays in Hybrid Modeling (Ahmad et al., 2006).

### 3.2. Hybrid Modeling Framework

It is difficult to model the changes in concentration levels of a protein through the discrete modeling frameworks. For instance, the protein expression level which often follows a sigmoidal curve (Figure 5A) is represented by positive integers in a discrete model where the change in level from 0 to 1 takes place instantly (Figure 5B) which is not a true representation of the actual increase in concentration of protein in a cellular environment. Therefore, the changes in concentration expression levels were proposed to be represented by piecewise linear curves for the modeling of the sigmoidal nature of protein expression (Ahmad et al., 2008) as shown in Figure 5C, as opposed to the step functions used in discrete models. Figures 5D–F) depict the modeling in case of degradation of protein.

A clock variable  $h_p$  is associated with each protein to measure the time it takes (delay) to go from a particular expression level to the next one. Initially, the value of each delay  $h_p$  is set to zero, which then increase to  $d_p^+$  (resp.  $d_p^-$ ) that signifies the production (resp. degradation) delay, i.e., the time it takes to increase (resp. decrease) the concentration level of the particular protein by 1 threshold level. Evolution rate for each clock  $h_p$  is set using first order derivative  $dh_p/dt = x$  where  $x \in \{0, 1, -1\}$  which represents no change, increase or decrease, respectively (Ahmad et al., 2008). Since the exact value of the delays associated with the production or degradation of proteins is not known in most cases, therefore, unvalued parametric delays are used instead and are determined through hybrid modeling (Ahmad

et al., 2012). Hybrid modeling approach has been applied for behavioral modeling of several biological networks including MAL-associated network (Ahmad et al., 2012), dengue virus pathogenesis and clearance mechanism (Aslam et al., 2014), tail-resorption in tadpole’s metamorphosis (Khalis et al., 2009), role of O-linked N-acetylglucosamine transferase in oncogenesis and cancer progression in hexosamine biosynthetic pathway (Saeed et al., 2016) and immunity control mechanism in bacteriophage lambda (Richard et al., 2006).

The hybrid modeling method has also been applied in opinion evolution in social networks (Shang, 2018). The Hybrid consensus for averager-copier voter networks with non-rational agents has been modeled and it is shown that there is a distinct influence of voters on the consensus and evolution of the opinion process.

The delays are of two types; Signal Transmission Delay and Signal Processing Delay. While the outcome for both types of delay would be the same, however, the processing delay is likely to cause errors in the consensus behavior whereas it is independent of the transmission delay (Shang, 2017). For study of BRNs we have to deal with the processing delay, hence, keeping track of the transient changes during the delay period is important and essentially required to be taken into account for correctly modeling the evolution of BRN and understanding its dynamical behavior.

### 3.3. Parametric Bio Linear Hybrid Automaton

We formally define hybrid automaton here by mainly using the notations and definitions given in Alur (1999) and adapted from Ahmad et al. (2006). We use a set of clock variables  $X = \{h_1, \dots, h_n\}$  to represent time which progress synchronously in accordance with  $h_i = 1$  or  $-1$  and can be reset to zero. The clock constraints  $\varphi$  for the set  $\phi(X)$  is defined by the grammar:

$$\varphi ::= true \mid h \leq d \mid h \geq d \mid h = d \mid \varphi_1 \wedge \varphi_2$$

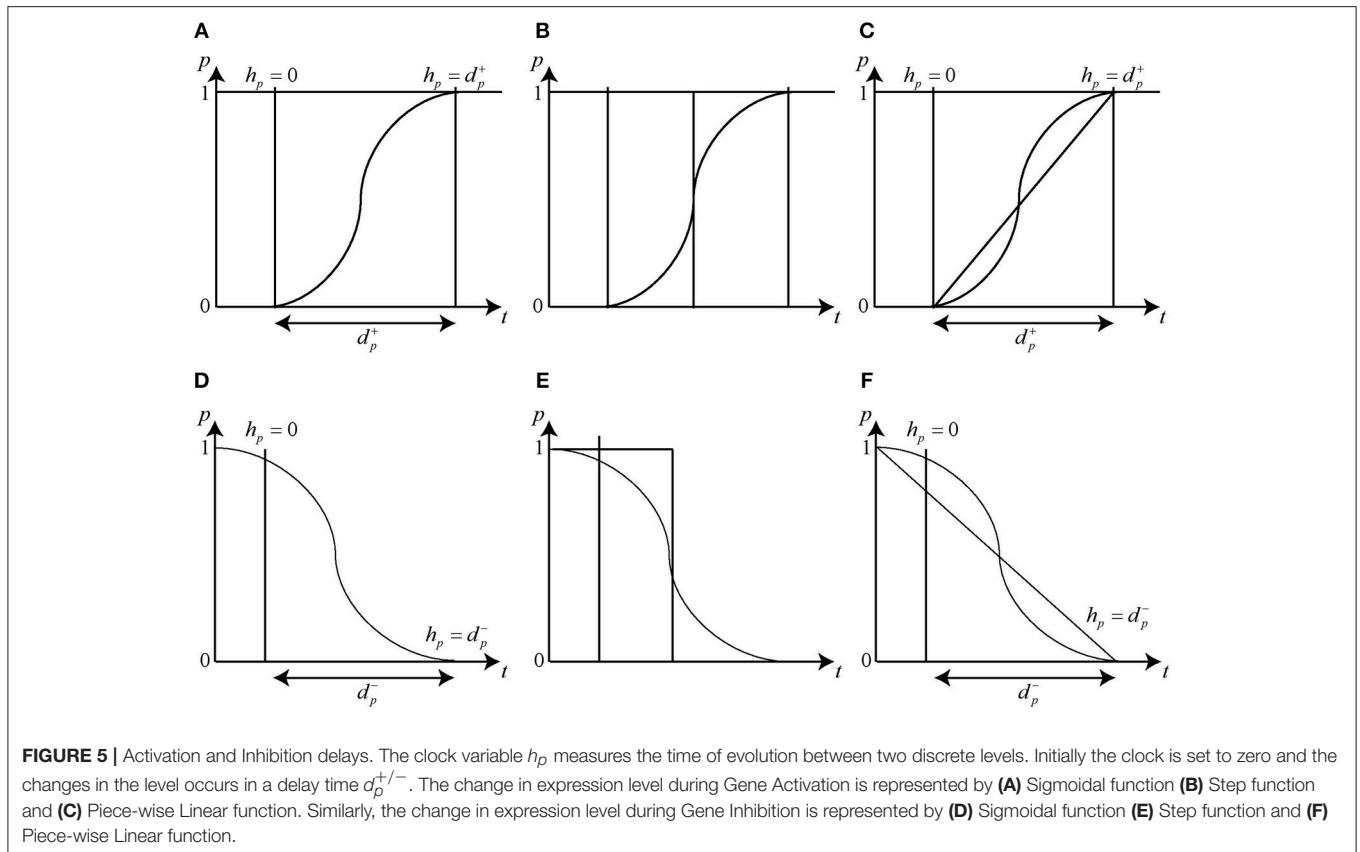
where  $h \in X$  and  $d$  is an un-valued parameter.

**Definition 10. (Parametric Biological Linear Hybrid Automaton (Bio-LHA)).** A parametric Bio linear hybrid automaton  $\mathbb{B}$  is a tuple  $(L, l_0, D, X, E, Inv, Dif)$  where,

- $L$  is a finite set of locations,
- $l_0 \in L$  is the initial location,
- $D$  is a finite set of parameters (delays),
- $X$  is a finite set of real-valued variable (clocks),
- $E \subseteq L \times \varphi \times 2^X \times L$  is a finite set of edges with typical element  $e = (l, g, R, l') \in E$  representing an edge from  $l$  to  $l'$  with guard  $g$  (clock constraint of the form  $h = d$ ) and the reset set  $R \subseteq X$ ,
- $Inv : L \rightarrow \varphi$  assigns an invariant to any location,
- $Dif : L \times X \rightarrow \{-1, 0, 1\}$  maps each pair  $(l, h)$  to an evolution rate.

The Transition System related semantics of the parametric Bio-LHA is given below according to the time domain  $\mathbb{T}$ , where  $\mathbb{T}^* = \mathbb{T} \setminus \{0\}$ .

**Definition 11. (Semantics of Bio-LHA).** Let  $\gamma$  be a valuation for the parameters  $P$  and  $v$  represents the values of



*clocks in a location. The  $(\mathbb{T}, \gamma)$ -semantics of a parametric Bio-LHA  $\mathbb{B} = (L, l_0, P, X, E, \text{inv}, \text{dif})$  is defined as a timed transition system  $B = (\mathbb{S}, s_0, \mathbb{T}, \rightarrow)$  where:*

- (1)  $\mathbb{S} = \{(l, v) \mid l \in L \text{ and } v \models \text{Inv}(l)\}$ ;
- (2)  $s_0$  is the initial state and
- (3) the relation  $\rightarrow \subseteq \mathbb{S} \times \mathbb{T} \times \mathbb{S}$  is defined for  $t \in \mathbb{T}$  as:
  - *discrete transitions:*  $(\ell, v) \xrightarrow{0} (\ell', v')$  iff  $\exists (\ell, g, R, \ell') \in E$  such that  $g(v) = \text{true}$ ,  $v'(h) = 0$  if  $h \in R$  and  $v'(h) = v(h)$  if  $h \notin R$ .
  - *continuous transitions:* For  $t \in \mathbb{T}^*$ ,  $(\ell, v) \xrightarrow{t} (\ell', v')$  iff  $\ell' = \ell$ ,  $v'(h) = v(h) + \text{Dif}(\ell, h) \times t$ , and for every  $t' \in [0, t]$ ,  $(v(h) + \text{Dif}(\ell, h) \times t') \models \text{Inv}(l)$ , where  $\models$  represents satisfaction operator.

### 3.4. Network of Bio-LHA

A BRN is composed of various entities which evolve concurrently depending on the activating (*resp. inhibiting*) influence of other entities as defined in the BRN. We have represented each entity with a Bio-LHA and its dynamics is also defined above. Next we define the network of Hybrid Automaton  $\mathbb{B}_N$  which is built up from composition of individual Bio-LHA  $\mathbb{B}_1, \dots, \mathbb{B}_n$ .

**Definition 12. (Network of Bio-LHA).** Let  $\mathbb{B}_i := (L_i, l_{0,i}, D_i, X_i, E_i, \text{Inv}_i, \text{Dif}_i)$ ,  $i = 1, \dots, n$  be the component Bio-LHA representing the genes of the BRN. The network of Bio-LHA representing  $n$  genes is  $\mathbb{A} = (L, l_0, D, X, E, \text{Inv}, \text{Dif})$  where,

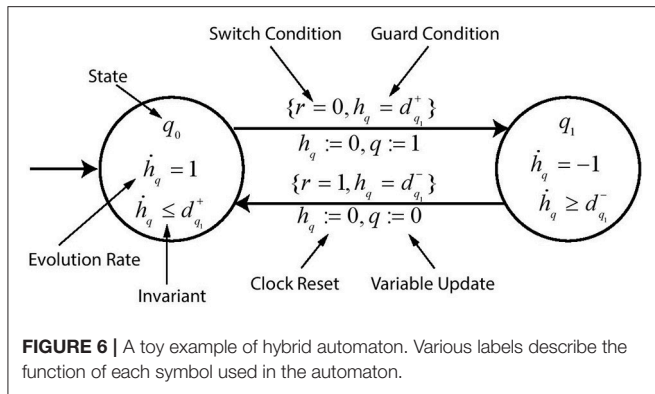
- $L = L_1 \times \dots \times L_n$ , is the set of locations,
- $l_0 = (l_{0,1}, \dots, l_{0,n}) \in L$  is the initial location,

- $D = \bigcup_{i=1}^n D_i$ , is the set of delays,
- $X = \bigcup_{i=1}^n X_i$ , is the set of clocks,
- The transition relation  $E$  is defined by the following asynchronous rule:
  - If  $l_i$  is any component location of  $l \in L$  then  $((\dots l_i, \dots), g_i, R_i, (\dots l'_i, \dots)) \in E$  for  $(l_i, g_i, R_i, l'_i) \in E_i$  and all other component locations of  $l$  do not change. The guard  $g_i$  is the conjunction of switch condition ( $\wedge$ ) and clock constraint ( $\varphi$ ),
- $\text{Inv} : L \rightarrow \bigcup_{i=1}^n \text{Inv}_i(l)$ , is mapping of invariants to locations,
- $\text{Dif} : L \times X \rightarrow \{-1, 0, 1\}$ , is the mapping of clocks to evolution rates.

### 3.5. Representation of Hybrid Automaton

The modeling of Hybrid Automaton representing evolution of gene  $q$  is depicted in **Figure 6** which is inhibited by gene  $r$ . The set of transitions  $\mathcal{H}$  is given by  $\{q_0 \xrightarrow{r_0} q_1\}, \{q_1 \xrightarrow{r_1} q_0\}$ . The automaton is modeled with two states  $q_0$  and  $q_1$  which represents the discrete threshold levels of the gene  $q$ . Each state is labeled with its evolution rate  $d_{q_1}^{+/-}$  and its invariant for which the automaton remains in that state. The edges are labeled with the guard condition on the clock variable as well as the switch condition  $\wedge$  on the gene level  $LEV_q^{+/-}$ . When the guard becomes True, the automaton moves to the next state with the clock being





reset and also updating the variable  $q$  which contains the current value of gene threshold level.

### 3.6. Implementation of Hybrid Automaton With HyTech

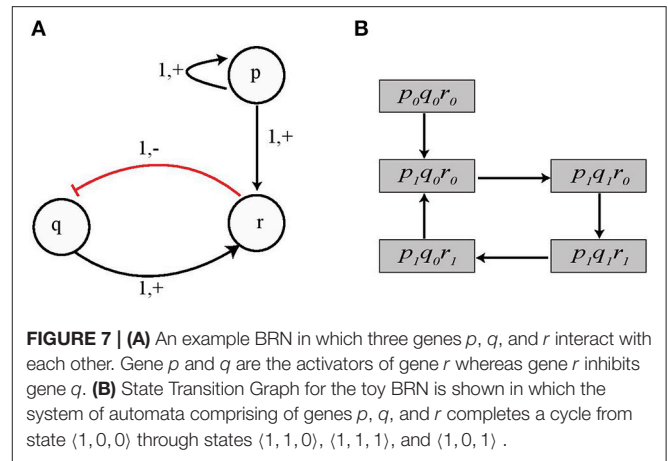
HyTech (Henzinger et al., 1997) is a model checking tool developed for the formal verification of real-time systems. It is suitable for modeling of Linear Hybrid Automata and has a powerful set of analysis commands as well as parametric synthesis capability. Using HyTech for modeling of BRN, each gene is modeled by an automaton. Thus, the current state of the BRN would be given by the current state of all the automata considered together.

**Example 4.** The code for implementation of toy example of hybrid automaton *geneQ* shown in **Figure 6** in HyTech is given below. Keyword *loc* is used to specify the *state* of automaton (e.g.,  $q_0$ ), keyword *while* is followed by the *invariant* condition (e.g.,  $h_q \leq dpq1$ ), keyword *wait* is used to give the rate of increase or decrease during the transition (e.g.,  $\{dhq=1\}$ ), keyword *when* is followed by the *guard* condition (e.g.,  $r = 0 \ \& \ hq = dpq1$ ), keyword *do* is used for updating of the state variable (e.g.,  $\{hq' = 0, q' = 1\}$ ) and keyword *goto* is followed by the next state of the automaton (e.g.,  $q_1$ ).

```

automaton geneQ
synclabs: ;
initially q0 & q=0 & hq=0;
loc q0: while hq<=dpq1 wait {dhq=1}
when r=0 & hq=dpq1 do {hq'=0,q'=1} goto q1;
loc q1: while hq<=dnq1 wait {dhq=-1}
when r=1 & hq=dnq1 do {hq'=0,q'=0} goto q0;
end --geneQ
    
```

We use the parametric delay variable  $hq$  here with all the transitions between different states of automaton which evolve at some rate till the switch condition  $\wedge$  becomes true for respective transitions. In this way we are able to get the range of values for this delay variable for the automaton to move from one state to another. Accordingly, we are able to determine the constraints on the delay variable corresponding to each state of automaton. When the system of automata representing the complete interaction graph of BRN is composed with switch conditions and delay variables for each gene evolution, we get the



constraints on delay variables for each of the composed state of automata. The complete methodology is given in next section.

## 4. INCORPORATING TIME DELAYS IN PROCESS HITTING FRAMEWORK

The aim of this section is to give the methodology of our proposed *Hybrid Process Hitting* technique in which we have incorporated the Time Delays from Hybrid Modeling described above into the Process Hitting Framework.

### 4.1. Obtaining Process Hits for BRN

Starting with the Interaction Graph of a BRN which specifies all the regulations (*type and threshold level*) of all its genes on each other, all the local transitions in the BRN were captured through application of Process Hitting as a first step. We now consider an example BRN in which two individual genes  $p$  and  $q$  are interacting with gene  $r$  as shown in **Figure 7A**. By considering the permissible actions in the generalized dynamics in this BRN we get the possible local transitions  $\mathcal{H}$  as given below:

$$\mathcal{H} = \{q_0 \xrightarrow{\{r_0\}} q_1, q_1 \xrightarrow{\{r_1\}} q_0, r_0 \xrightarrow{\{p_1, q_1\}} r_1, r_1 \xrightarrow{\{p_0, \{q_0\}} r_0\}.$$

### 4.2. Deducing the Switch Conditions and Effective Levels

Now using **Table 1**, the effective levels of interacting genes  $p$  and  $q$  are determined from the corresponding condition required by the cooperative sort for upregulation of gene  $r$ . For  $pq_{11}$  we get  $LEV_p^+ = \{1\}$  for gene  $p$  and  $LEV_q^+ = \{1\}$  for gene  $q$ . We interpret the values of effective levels in a straight forward manner in which the levels represent the current state of the individual automaton of gene  $p$  and  $q$ . In this case, the automaton *geneP* is at state  $P_1$  and automaton *geneQ* is at state  $Q_1$ . For automaton *geneR* which describes the dynamics of gene  $r$ , the effective levels, therefore, specifies the *guard* condition in terms of exact combination of state variables of other automata that would induce the transition from state  $R_0$  to  $R_1$ .

### 4.3. State Transition Graph

All the automata when considered together would represent the complete model of BRN. Hence, the transitions from one state to another would be governed by the dynamics of the modeled BRN. At any given time, the locations of all the automata give us the current state of the BRN. A change in location of any one automata result in change of state of BRN. For the BRN shown in **Figure 7A** consisting of genes  $p, q$  and  $r$  with expression levels as 1, 0 and  $\downarrow$ , respectively, the current state is denoted as  $\langle 1, 0, 0 \rangle$  with corresponding locations of automata as  $loc p_1, loc q_0$  and  $loc r_0$ . Here, we note that the switch condition for evolution of automaton  $geneQ$  is satisfied (i.e.,  $LEV_r^+ = \{0\}$ ) for it to move from location  $loc q_0$  to  $loc q_1$ , thus the next state of the system would be  $\langle 1, 1, 0 \rangle$ .

In the state  $\langle 1, 0, 0 \rangle$ , it may be noted that the required switch conditions for evolution of automaton  $geneR$  were not satisfied, so this automaton could not evolve. However, we see that in state  $\langle 1, 1, 0 \rangle$  the switch conditions for evolution of  $geneR$  are now satisfied (i.e.,  $p = 1$  and  $q = 1$ ) therefore, the next transition lead the system to state  $\langle 1, 1, 1 \rangle$ . Subsequently, the system evolves to state  $\langle 1, 0, 1 \rangle$  from where it comes back to state  $\langle 1, 0, 0 \rangle$ . The corresponding state transition graph for the dynamics of the system model is shown in **Figure 7B**.

### 4.4. Enriching the Automaton With Time Delays

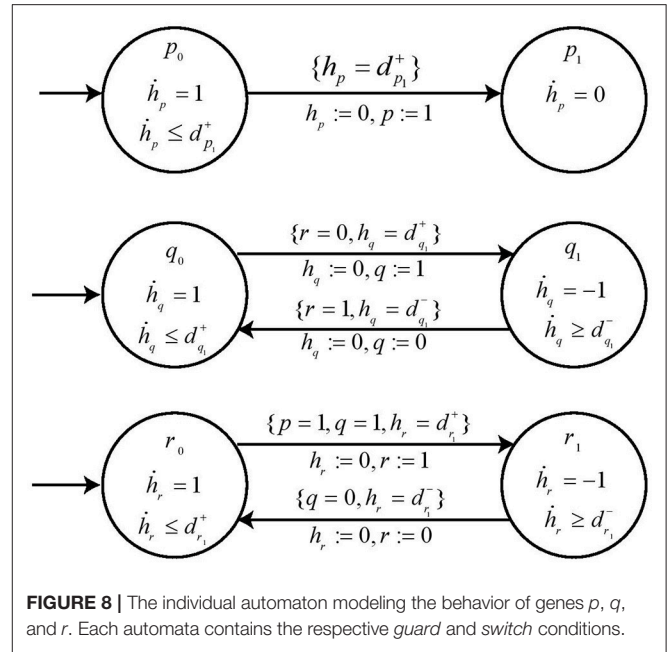
The transition from one state of the system to another takes place when the concentration level increases (*resp. decreases*) from one threshold level to another and it takes a finite amount of time. This time is measured in terms of delay variable  $d_q^{+/-}$  in hybrid modeling of BRN. We note that the automaton moves from one state to another instantaneously once the switch conditions are satisfied. So in order to measure the time delay for this transition we introduce the *parametric* clock variable  $hq$  which increases at a rate  $dhq/dt$  for the time the automaton remains in a particular state and its value is recorded on transition to the next state.

Consider the initial location of automaton  $geneQ$  is  $loc q_0$  and the clock variable  $h_q$  is initially set to 0 which starts to increase with rate  $dhq/dt$ . On satisfaction of switch conditions, automaton  $geneQ$  leaves its current state  $loc q_0$  and moves to next state  $loc q_1$  which represents the transition of Gene  $q$  from expression level 0 to 1. At this time the value of clock variable  $h_q$  for this transition is noted in terms of delay variable as  $d_{q_1}^+$ . Similarly, the delay for moving from level 1 to 0 will be given by  $d_{q_1}^-$ .

### 4.5. Composing System of Automata

We compose the System of Automata by considering all individual automaton together. The transitions in each automaton are linked with the state of other automata through switch conditions and the time taken by each transition is obtained in terms of its delay variable. We recall that the switch conditions are given by the *refined cooperative hits* and represent the permissible dynamics in the BRN. When all the individual automaton are considered together we obtain the constraints on the range of values of respective clocks for the system to move from one state to another.

**Example 5.** In order to describe our methodology for obtaining the time delays, we now model the individual genes



**TABLE 2** | Delay Constraints obtained for the example BRN through modeling of its corresponding system of Automata in HyTech.

Path	$p_0q_0r_0 \rightarrow p_1q_0r_0 \rightarrow p_1q_1r_0 \rightarrow p_1q_1r_1 \rightarrow p_1q_0r_1 \rightarrow p_1q_0r_0$
Delay	$(d_{p_1}^+ \leq d_{q_1}^+ \wedge d_{p_1}^+ \leq d_{r_1}^+) \wedge (d_{q_1}^+ \leq d_{p_1}^+ \wedge d_{q_1}^+ \leq d_{r_1}^+ +  d_{q_1}^- ) \wedge$
Constraints	$(d_{r_1}^+ \leq d_{q_1}^+ +  d_{q_1}^- ) \wedge (d_{q_1}^+ +  d_{q_1}^-  \leq d_{r_1}^+ +  d_{r_1}^- ) \wedge$ $(d_{r_1}^+ +  d_{r_1}^-  \leq 2d_{q_1}^+ +  d_{q_1}^- )$

The parametric delays have been introduced in Process Hitting Framework to give the constraints for the complete path of the State Transition Graph.

in BRN of **Figure 7A** with respective automaton. From  $\mathcal{H}$  we note the *action* (switch condition) for upregulation  $q_0$  to  $q_1$  for automaton  $geneQ$  as  $r_0$  and effective level  $LEV_r^+ = \{0\}$ . Therefore, the *guard* condition for automaton  $geneQ$  is set as  $r = 0$  &  $h_q = d_{q_1}^+$ . Similarly for downregulation, the *guard* condition is  $r = 1$  &  $h_q = d_{q_1}^-$ . In case of  $geneR$ , we have the switch condition for upregulation as  $pq_{11}$  given by the logical function  $p \wedge q$  with effective levels as  $LEV_p^+ = \{1\}$  and  $LEV_q^+ = \{1\}$ . For downregulation, we get the switch conditions and effective levels from **Table 1** for the given logical function. Accordingly, the *guard* conditions in automaton  $geneR$  are set to model this behavior. The individual automaton for  $p, q$  and  $r$  are modeled as shown in **Figure 8**. The corresponding *hytech* file for modeling this BRN is given in **Supplementary File 1**. We assume that  $gene p$  is at level 1 initially and *hytech* file is run to get the constraints on delay parameters corresponding to each state of composed system as shown in **Table 2** below.

### 4.6. Inference of Parametric Delays in Hybrid Process Hitting

The range of delay parameters alongwith the constraints on delay variables in terms of time delays is obtained as output from simulation of System of Automata which modeled the proposed

*Hybrid Process Hitting* approach. As evident from Table 2, for each path or trajectory from one particular state to another of the State Transition Graph, we get the constraints on clock variables in terms of time delays for each gene of BRN. For instance, the delay constraint for the path from state  $p_0q_0r_0$  to state  $p_1q_0r_0$  requires that  $d_{p_1}^+ \leq d_{q_1}^+$  and  $d_{p_1}^+ \leq d_{r_1}^+$ . Similarly, the delay constraints corresponding to the paths between various states are obtained from running the *HyTech* model.

It is highlighted that the delays  $d_{q_1}^+$  and  $d_{q_1}^-$  represents the accumulation and degradation time of gene  $q$ . Hence, the constraints on the delays give us the conditions in terms of time taken for accumulation or degradation of biological entities in the BRN and clearly define the dynamics of the modeled system.

We are thus able to determine the dynamical behavior of a biological regulatory network in which the constraints on time delays are deduced for each state.

## 5. BIOLOGICAL APPLICATION I: BACTERIOPHAGE LAMBDA

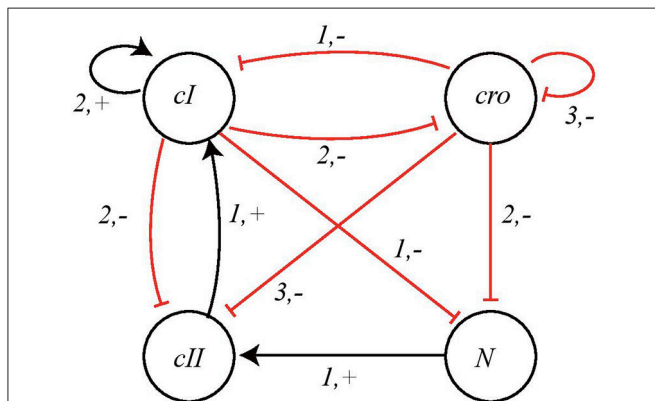
In this section we apply the proposed *Hybrid Process Hitting* approach to the well-known example of *Bacteriophage λ* (Thieffry and Thomas, 1995). This network has been widely studied because of its peculiar switch mechanism because of which this virus chooses between *lysis* and *lysogenization*. These two responses are the result of different dynamics of the BRN and thus leads to separate attractors.

### 5.1. Automata Network

We take into consideration the BRN model proposed in Thieffry and Thomas (1995) and shown in Figure 9. The network has 4 nodes, having expression levels 3, 4, 2, and 2, respectively, so in this case:

$$\begin{aligned} \Sigma &= \{cI, cro, cII, N\}, L_{cI} = \{cI_0, cI_1, cI_2\}, \\ L_{cro} &= \{cro_0, cro_1, cro_2, cro_3\}, L_{cII} = \{cII_0, cII_1\} \text{ and} \\ L_N &= \{N_0, N_1\}. \end{aligned}$$

We determine the *local transitions*  $\mathcal{H}$  for this network from the BRN interaction graph. The associated Automata Network is shown in Figure 10.



**FIGURE 9 |** BRN of Bacteriophage λ immunity response showing interactions between nodes  $cI$ ,  $cro$ ,  $cII$ , and  $N$ .

### 5.2. Obtaining the Switch Conditions

The logical function for the cooperation between  $cI$  and  $cro$  is  $\neg cI \wedge \neg cro$  as both nodes are inhibitors of node  $N$  which implies that the process  $cI - cro_{LL}$  produce upregulation of node  $N$  while all other processes, i.e.,  $cI - cro_{LH}$ ,  $cI - cro_{HL}$ , and  $cI - cro_{HH}$  would result in downregulation of target node  $N$ . Here we note that the nodes  $cI$  and  $cro$  are having multiple expression levels but the threshold level for interactions on node  $N$  are 1 and 2, respectively. This condition then gives us the *effective levels* for the process  $cI - cro_{LL}$  which are  $LEV_{cI}^- = \{0\}$  and  $LEV_{cro}^- = \{0, 1\}$ . Thus, the gene  $N$  would bounce from  $N_0$  to  $N_1$  if its regulators have the expression levels  $(cI_0, cro_0)$  or  $(cI_0, cro_1)$ .

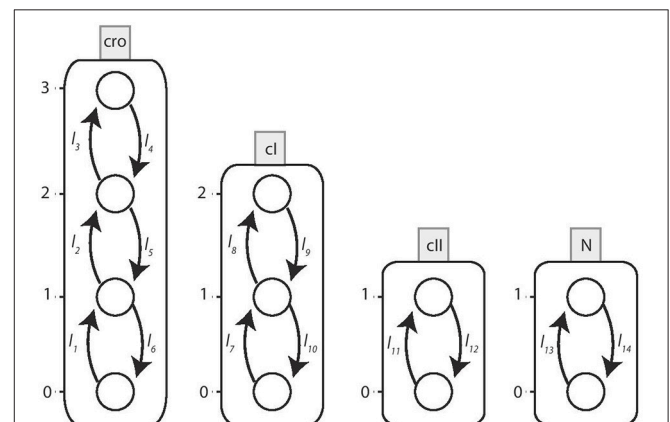
Similarly, we get the effective levels and corresponding expression levels for downregulation of node  $N$  from the other processes of cooperative sort  $cI - cro$  as follows:

$$\begin{aligned} cI - cro_{LH} : LEV_{cI}^- &= \{0\}, LEV_{cro}^+ = \{2, 3\} \\ &(cI_0, cro_2), (cI_0, cro_3) \\ cI - cro_{HL} : LEV_{cI}^+ &= \{1, 2\}, LEV_{cro}^- = \{0, 1\} \\ &(cI_1, cro_0), (cI_1, cro_1), (cI_2, cro_0), (cI_2, cro_1) \\ cI - cro_{HH} : LEV_{cI}^+ &= \{1, 2\}, LEV_{cro}^+ = \{2, 3\} \\ &(cI_1, cro_2), (cI_1, cro_3), (cI_2, cro_2), (cI_2, cro_3) \end{aligned}$$

### 5.3. Automaton Specification

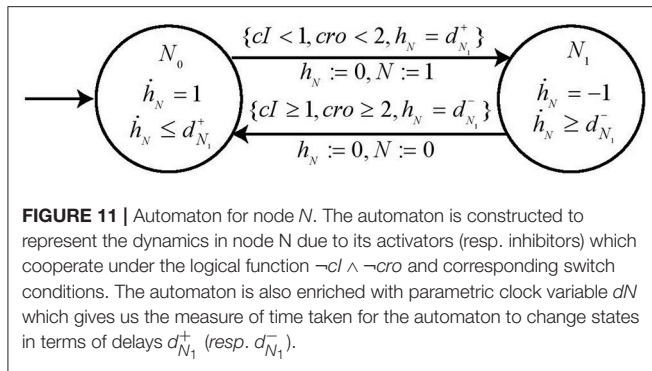
These expression levels give us the required switch conditions for completely specifying the automaton for node  $N$ . It contains two states, namely, *loc*  $N_0$  and *loc*  $N_1$  since node  $N$  has two processes and are represented by state variable  $N = 0$  and  $N = 1$ . The clock variable is  $h_N$  and the parametric delay variables are  $d_{N_1}^+$  (*resp.*  $d_{N_1}^-$ ) for the time taken for up (*resp.* down) regulation for node  $N$ . By following our proposed methodology we deduce the *Invariant* and *guard* conditions for each state of the automaton to get the *Hybrid Process Hitting* model for node  $N$  as shown in Figure 11.

Following the same methodology, the automaton for other nodes  $cI$ ,  $cro$  and  $cII$  are constructed with their switch conditions



**FIGURE 10 |** The Automata Network model of Bacteriophage λ immunity response. The corresponding local transitions  $\mathcal{H}$  are:

$$\begin{aligned} I_1 = I_2 = I_3 &= \{cI_{\leq 1}\}, & I_4 &= \{cI_2, cro_3\}, I_5 = I_6 = \{cI_2\} \\ I_7 = I_8 &= \{cII_1, cro_0\}, & I_9 = I_{10} &= \{cII_0\}, \{cro_{\geq 1}\} \\ I_{11} &= \{N_1, cI_{\leq 1}, cro_{\leq 2}\}, & I_{12} &= \{N_0\}, \{cI_2\}, \{cro_3\} \\ I_{13} &= \{cI_{<1}, cro_{<2}\}, & I_{14} &= \{cI_{\geq 1}\}, \{cro_{\geq 2}\}. \end{aligned}$$



obtained from the logical functions  $\neg cI \wedge \neg cro \wedge cII$ ,  $\neg cI \wedge \neg cro$  and  $\neg cI \wedge \neg cro \wedge N$ , respectively. These logical functions are the result of refined cooperative hits obtained from Process Hitting. The cooperativity among three nodes as required for nodes  $cI$  and  $cII$  is constructed in two steps as explained earlier in section 2 above. The specifications of these automata are completed by introducing the respective parametric clock and delay variables. The automata for these other entities,  $cI$ ,  $cII$  and  $cro$  are shown in Figure 12.

## 5.4. Modeling of Clock Variables for Multi-Level Genes

In the Bacteriophage  $\lambda$  BRN we have two multi-level genes, i.e.,  $cI$  and  $cro$  which have 3 and 4 levels, respectively. For modeling these multi-level genes in *HyTech*, we introduce new clock variable  $h_n$ . Consider the Automaton of gene  $cI$  which starts from initial location  $cI_0$  and moves to next location  $cI_1$  when the appropriate switch conditions ( $cII = 1, cro < 1$ ) are met. Here if the switch conditions remain the same, the automaton would move to location  $cI_2$ , however if either of the switch condition changes then the automaton would move towards location  $cI_0$ . We note that the jump from  $cI_1$  to  $cI_2$  represents the accumulation of  $cI$  whereas moving from  $cI_1$  to  $cI_0$  shows the degradation of entity  $cI$ . Hence, we require two clock variables  $h_{cI}$  and  $h_{ncI}$  to keep track of the accumulation and degradation delays, respectively.

## 5.5. Composition of System of Automata

Once all the individual automaton are constructed, we now compose the system of automata by executing the *HyTech* file. Since we are interested in the dynamical characteristics of the transitions that take place, therefore, we use the *reach* command in the *Analysis* section of *HyTech*. So starting from any arbitrary initial state we are able to find the next states that the system of automata can reach alongwith the corresponding constraints on the time taken for this change of state in terms of *time delays*.

## 5.6. Results

We compose the system of automata for *Bacteriophage*  $\lambda$  and get the results on the constraints of time delays corresponding to each state that the system can reach as given in **Supplementary File 2**. It is highlighted that we get 42 states instead of the total 48 possible states which shows that the state

transition graph has been reduced through application of *Process Hitting*. Since we used the *reach forward* command, therefore, the *HyTech* has computed all trajectories alongwith the delay constraints for the system of automata which it can possibly traverse with the defined *switch conditions* for each automaton.

We know that the *phage*  $\lambda$  BRN has two attractors, i.e., *lysogenic* corresponding to state  $\langle cI, cro, cII, N \rangle = \langle 2, 0, 0, 0 \rangle$  and *lysis* alternating between states  $\langle cI, cro, cII, N \rangle = \langle 0, 2, 0, 0 \rangle$  and  $\langle 0, 3, 0, 0 \rangle$  (Thieffry and Thomas, 1995). From the results, we note the constraints on time delays for the paths leading to above mentioned states. For *lysogenic* condition  $\langle 2, 0, 0, 0 \rangle$ , gene  $cI$  reaches its highest threshold level of 2 and represses the expression of all other genes. The time delays for the path leading to this state are found to be:

$$d_{cl_2}^+ \leq |d_{cl_1}^-| \wedge d_{cl_1}^+ + d_{cl_2}^+ \leq d_{cro_1}^+ \wedge d_{cl_2}^+ \leq d_{cII_1}^+ \wedge d_{cl_1}^+ \leq d_{cII_1}^+$$

In *lysis*, the delays for the path leading to state  $\langle 0, 2, 0, 0 \rangle$  are:

$$d_{cro_2}^+ \leq |d_{cro_1}^-| \wedge d_{cro_2}^+ \leq d_{cII_1}^+ \wedge d_{cro_1}^+ \leq d_{cII_1}^+ \wedge d_{cro_1}^+ + d_{cro_2}^+ \leq d_{N_1}^+$$

and for the path leading to state  $\langle 0, 3, 0, 0 \rangle$  the delays are:

$$d_{cro_2}^+ \leq |d_{cro_1}^-| \wedge d_{cro_3}^+ \leq |d_{cro_2}^-| \wedge d_{cro_2}^+ \leq d_{cII_1}^+ \wedge d_{cro_1}^+ \leq d_{cII_1}^+ \wedge d_{cro_1}^+ + d_{cro_2}^+ + d_{cro_3}^+ \leq d_{N_1}^+ \wedge d_{cro_3}^+ \leq d_{cII_1}^+ \wedge 2d_{cro_1}^+ + 2d_{cro_2}^+ + 2d_{cro_3}^+ \leq d_{cI_1}^+ + 2d_{cII_1}^+ \wedge d_{cro_1}^+ + d_{cro_2}^+ + d_{cro_3}^+ \leq 2d_{cII_1}^+$$

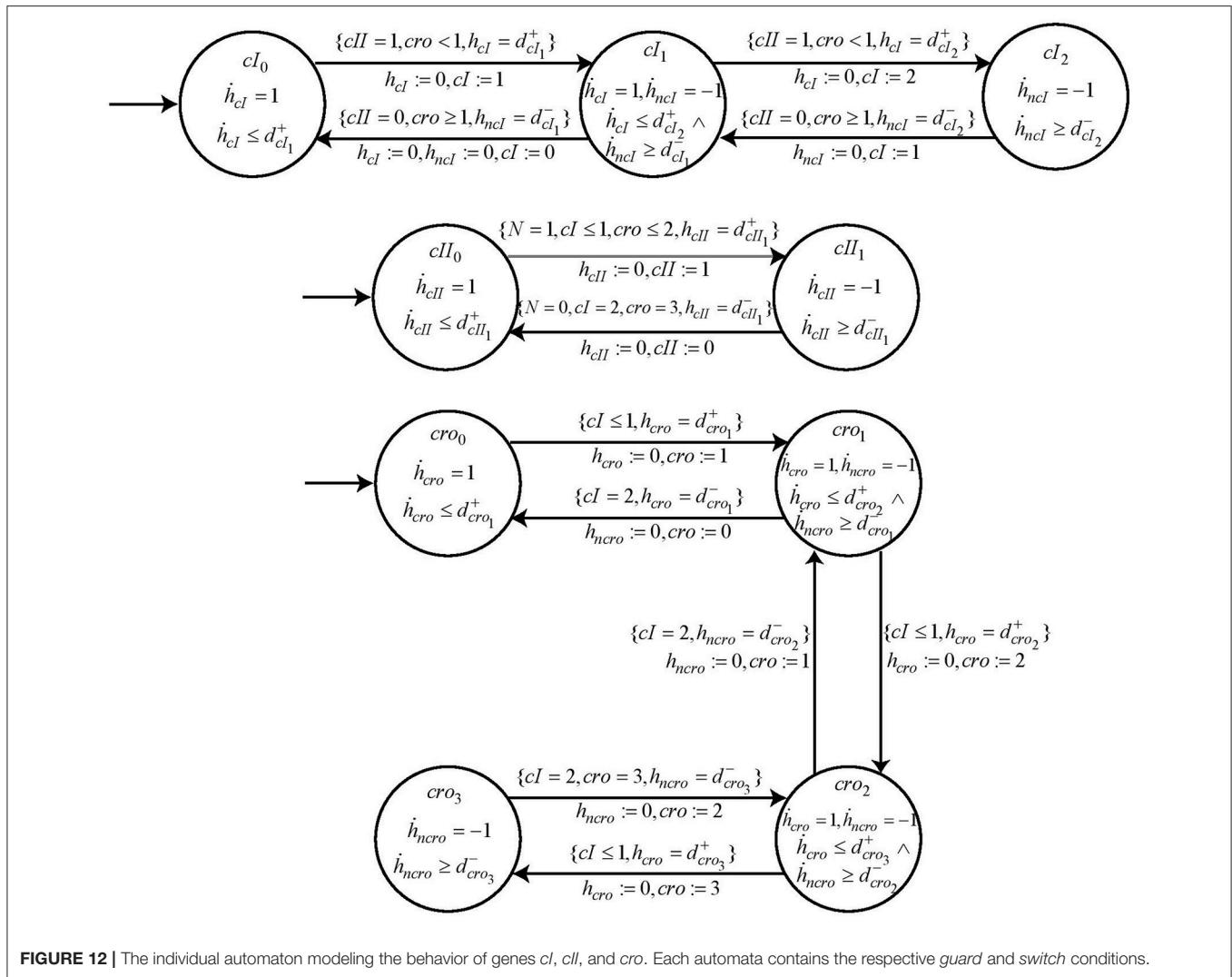
By analysing the delay constraints for each state we get the relationship between the rates of accumulation or decay of a particular entity in the BRN. These results on delay constraints are found to be slightly different from Ahmad et al. (2008) due to the difference in the way the delays are modeled. In Ahmad et al. (2008), the time delays between various levels of a multi-level gene were modeled with the same delay variable whereas we have specified it to be different for each change in level. For instance, our modeling methodology considers  $d_{cro_1}^+$  as the time delay for  $cro$  to change from level 0 to 1 while  $d_{cro_2}^+$  gives the time delay from level 1 to 2. But in Ahmad et al. (2008), both these delays were modeled with the delay variable  $d_{cro}^+$ . Our specification of time delays is considered more suitable for modeling of dynamic behavior of genes in BRNs since the time for change amongst various levels of a biological entity are usually different from each other.

## 6. BIOLOGICAL APPLICATION II: ERBB RECEPTOR-REGULATED G1/S TRANSITION BRN

Here we consider a comparatively large BRN of ERBB Receptor-regulated G1/S transition involved in the breast cancer and has regulations between 20 genes (Sahin et al., 2009). The BRN for this network is shown in Figure 13. The gene  $pRB$  is activated by this activation cascade network which subsequently controls the G1/S transition involved in cell divisions. The input of this network is the gene  $EGF$  which when expressed will result in potential activation of gene  $pRB$ .

In multicellular organisms, growth factors play a significant role in order to maintain proper growth, development and homeostasis. In regard to receptor tyrosine kinases (RTKs), the





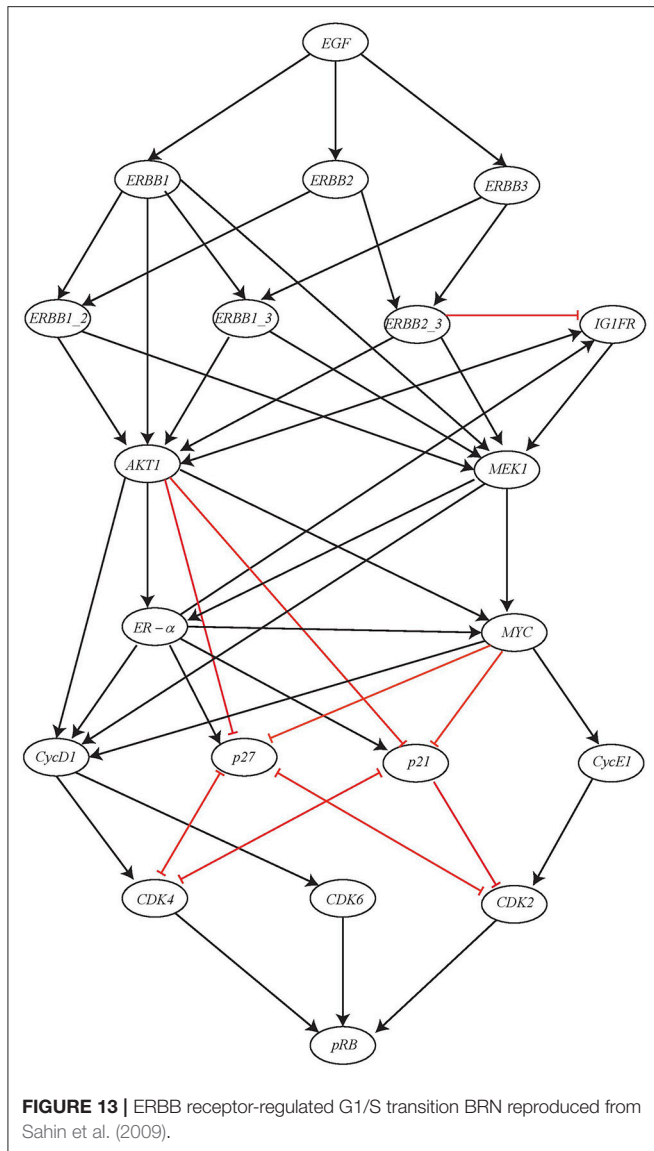
**FIGURE 12 |** The individual automaton modeling the behavior of genes *cl*, *cll*, and *cro*. Each automata contains the respective *guard* and *switch* conditions.

epidermal growth factor i.e., EGF family of RTKs also known as ERBB receptors is considered as hallmark of the human cancers along with process of growth, development and physiology. The ERBB family of protein functions through homodimers and heterodimers and thus, regulates the G1/S transition during cell cycle progression in eukaryotic organisms by modulating the activity of the Cyclin D, Cyclin E/CDK complex, the c-MYC oncogene, and the p27 kinase inhibitor.

In order to maintain a continuous simulation of EGF, the initial state of *ER* -  $\alpha$  node was set to 1. Considered to be the inhibitors of CDKs, p21 and p27 were assigned the initial state of 0. This is because the expression level of both of these inhibitors was found to be high during G0/G1 arrested cells and once cells progress through S phase, their level gets decreased because of their proteasomal degradation. Similarly, the initial state of cyclin D1 was set to 0. Now, using the above mentioned initial states, all possible state transitions were evaluated until a unique stable state was obtained that had fixed levels of all the proteins as shown in **Table 3**.

We apply the proposed Hybrid Process Hitting approach to this network to obtain the Hybrid Automaton for determining the time delays involved in its dynamical evolution. First we note the switch conditions and by using them we model the Hybrid Automaton. Once this Hybrid Automaton model was run in *HyTech* to compute the time delays, it was observed that the software went *Out of Memory*. This clearly showed the limitation of *HyTech* as it was unable to handle 20 Hybrid Automata at one time. The problem was overcome by utilizing the powerful reduction feature of PINT software (Paulevé, 2017) which is able to provide *Goal-oriented* model reduction through its *Cutsets* feature.

Typically a goal is the activation / inhibition of some gene in the modeled Automata Networks for BRNs. From a given initial state, the goal is reachable if a sequence of steps exists which leads to a the state in which goal is present. The sequence of states leading to the goal are known as *trace*. So *Cutsets* are sets of those local states which are included in traces leading to the goal state. Thus, the goal oriented reduction would be to apply *Cutsets*



**FIGURE 13 |** ERBB receptor-regulated G1/S transition BRN reproduced from Sahin et al. (2009).

**TABLE 3 |** Stable states for ERBB receptor-regulated G1/S transition obtained by PINT software (Paulevé, 2017).

AKT1	CDK2	CDK4	CDK6	CycD1
0	0	0	0	0
1	1	1	1	1
1	1	1	1	1

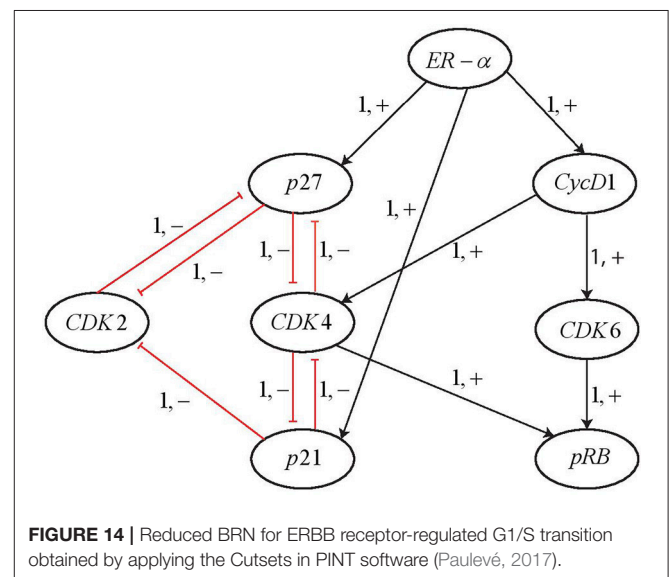
CycE1	EGF	ERBB1	ERBB1_2	ERBB1_3
0	0	0	0	0
1	0	0	0	0
1	1	1	1	1

ERBB2	ERBB2_3	ERBB3	ERalpha	IGF1R
0	0	0	0	0
0	0	0	1	1
1	1	1	1	0

MEK1	MYC	p21	p27	pRB
0	0	0	0	0
1	1	0	0	1
1	1	0	0	1



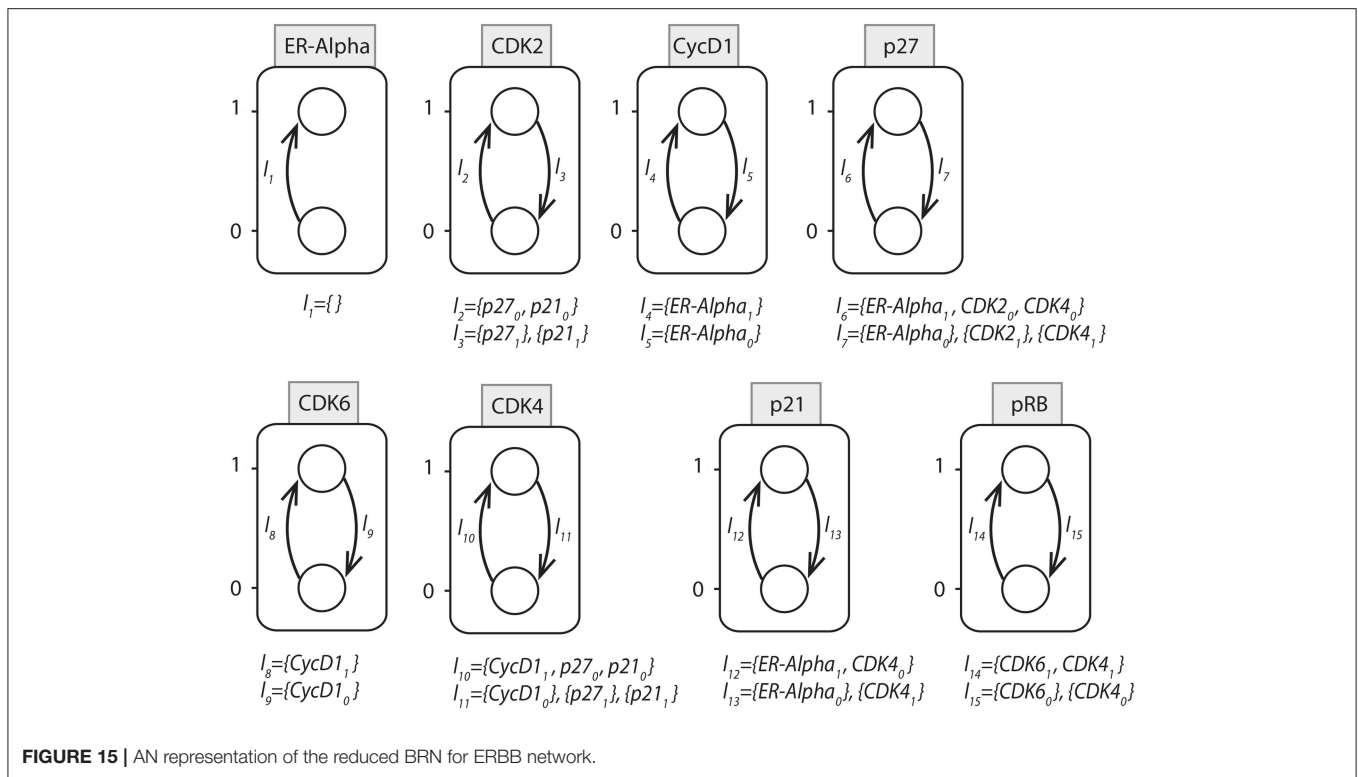
**FIGURE 14 |** Reduced BRN for ERBB receptor-regulated G1/S transition obtained by applying the Cutsets in PINT software (Paulevé, 2017).

so as to remove all *traces* leading to the goal. These *Cutsets* are implemented by the knock-out of the intermediate genes present in the network.

In the ERBB network, Normal epithelial cell lines exhibit high expression levels of ERBB1 along with ERBB2 but, have low levels of ERBB3. Unlike ERBB1, ERBB2 does not show any ligand binding activity and ERBB3 tends to have defective tyrosine kinase domain. Due to which both ERBB2 and ERBB3 are unable to activate the key signaling intermediates MEK1 and AKT1. Hence, we began our activation of network through *ER - α* and eliminated all the ERBB members including, ERBB1, ERBB1-2, ERBB1-3, ERBB2, ERBB2-3, ERBB3 by applying cut-sets in PINT tool using the following command:

```
y = y.having(ERalpha=1).reduce_for_goal("AKT1=1,CDK2=1,CDK4=1,CDK6=1,
CycD1=1,CycE1=1,EGF=1,ERalpha=1,ERBB1=1,ERBB1_2=1,ERBB1_3=1,ERBB2=1,
ERBB2_3=1,ERBB3=1,IGF1R=0,MEK1=1,MYC=1,p21=0,p27=0,pRB=1")
```

By applying the Cutsets to knock out the genes suggested by PINT, the ERBB network was reduced to 8 entities as shown in **Figure 14**. It can be observed that all the entities having feedback loops have been retained whereas the ones with straight forward regulations were removed. The AN representation of the reduced network is shown in **Figure 15**. The reduced network is then modeled as Hybrid Automata in *HyTech* and time delays are computed for the transitions between various states. The *HyTech* code and the time delays corresponding to various states of



reduced ERBB network are given in the **Supplementary Files 3, 4**, respectively.

It is noted that  $ER\alpha$  is required for the activation of *Cyclin D1*. At the pivotal point of G1/S transition, cells are dedicated to enter the S phase and results in DNA replication. This process is primarily regulated by *CycD1/CDK4/CDK6* complex that phosphorylates and thus, activates the retinoblastoma tumor suppressor protein *pRB*. For the state  $(1, 0, 1, 0, 1, 1, 0, 1)$  in which the protein *pRB* is expressed from the initial state  $(1, 0, 0, 0, 0, 0, 0, 0)$ , the constraints on the time delays are obtained from the *HyTech* results as shown in **Table 4**.

## 7. CONCLUSION AND FUTURE WORK

In this paper we have proposed a new approach, *Hybrid Process Hitting*, through which the time delay information has been incorporated in the Process Hitting framework for analysis of large Biological Regulatory Networks. It has been accomplished by integrating the principles of hybrid modeling with those of Process Hitting. The proposed approach utilizes the advantage of *Process Hitting* in terms of reduction of large state graph which represents the evolution of BRN. This is achieved by considering only the permissible or possible dynamics in the state graph evolution and also by considering the combined influence in case of interactions involving multiple genes.

The feature of Cutsets is utilized for the large BRNs to knock-out some genes to achieve goal-oriented reduction through PINT software tool (Paulevé, 2017). Once the reduced BRN is obtained then the time delay variables for each gene are introduced in it

**TABLE 4 |** Delay Constraints obtained for the ERBB Receptor-regulated G1/S transition BRN through modeling of its corresponding system of Automata in HyTech.

Path	$ER\alpha_1 CDK2_0 CycD1_0 p27_0 CDK6_0 CDK4_0 p21_0 pRB_0 \rightarrow ER\alpha_1 CDK2_0 CycD1_1 p27_0 CDK6_1 CDK4_1 p21_0 pRB_1$
Delay	$\sigma_{pRB_1}^+ \geq \sigma_{CDK4_1}^+ \wedge \sigma_{pRB_1}^+ \geq \sigma_{CDK6_1}^+ \wedge \sigma_{pRB_1}^+ \geq  \sigma_{ER\alpha_1}^-  \wedge$
Constraints	$\sigma_{pRB_1}^+ \leq \sigma_{CDK2_1}^+ \wedge \sigma_{pRB_1}^+ \geq  \sigma_{CycD1_1}^-  \wedge \sigma_{pRB_1}^+ \leq \sigma_{p27_1}^+ \wedge$ $\sigma_{pRB_1}^+ +  \sigma_{CDK6_1}^-  \leq \sigma_{CDK6_1}^+ \wedge \sigma_{pRB_1}^+ +  \sigma_{CDK4_1}^-  \leq \sigma_{CDK4_1}^+ \wedge$ $\sigma_{pRB_1}^+ \leq \sigma_{p21_1}^+$

The constraints on the parametric delays for the path from the State  $(1, 0, 0, 0, 0, 0, 0, 0)$  to the State  $(1, 0, 1, 0, 1, 1, 0, 1)$ , in which the protein *pRB* is expressed, are given.

which keeps track of the time taken during each step of BRN evolution. Hence, it becomes possible to obtain the constraints in terms of time delays for each gene corresponding to every evolution step of State Graph.

The time delays found through the application of Hybrid Process Hitting on the evolution of large biological networks gives us very useful information especially at the bifurcation points in the State Graph. While we can see the different paths in the State Graph through static analysis also, however, it cannot be determined which path would evolve before the others. Here, the time delay information becomes highly useful. Once we have this information, we can clearly identify the therapeutic targets to control the progress of State Graph towards desirable goals, i.e., homeostasis conditions.

It is highlighted that the proposed approach takes into consideration the concurrent evolution of all components of BRN and its modeling is realized through Hybrid Automaton. It implies that all the genes are free to evolve simultaneously as allowed by the switching conditions specified by *Process Hitting*. In this way, true dynamical behavior of BRN evolution from some given initial state to all the possible final (stable) states is obtained.

We have given the complete methodology for this approach with a running example on a toy BRN and it has also been applied to the simpler BRN of *Bacteriophage  $\lambda$*  and large BRN of *ERBB receptor-regulated G1/S transition* to show its tractability to large biological networks for computing the time delays associated with the transitional behavior of Network evolution.

Normal regulation of Insulin secretion related pathways pose an important aspect of pancreatic beta cells functionality. Several different types of substrates including metabolites, cytokines and neurotransmitters are involved in regulating insulin secretion through different pathways which also converge with each other at different points in this large interactome.

We plan to apply the proposed *Hybrid Process Hitting* methodology to Insulin secretion pathways considering all the known stimuli at the same time which would give us a more holistic picture of their regulation in maintaining pancreatic beta cells functional integrity. Overall, our aim is to model the collective dynamics of this insulin secretion interactome

in pancreatic beta cells. This could increase our current comprehension in treating Diabetes Mellitus for which the first line of treatment is the use of oral anti-diabetic drugs which mainly target these pathways.

## AUTHOR CONTRIBUTIONS

IS and JA conceived and developed the methodology, conducted the experiments and writing of manuscript. All the authors took part in discussions, analysis and layout of results, and reviewed the manuscript.

## ACKNOWLEDGMENTS

We would like to thank Dr Rehan Zafar Paracha, Dr. Muhammad Tariq Saeed, and Dr. Amnah Siddiq for their valuable suggestions without which this paper would not have been possible.

## SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fphys.2019.00090/full#supplementary-material>

## REFERENCES

- Ahmad, J., Bernot, G., Comet, J.-P., Lime, D., and Roux, O. (2006). Hybrid modelling and dynamical analysis of gene regulatory networks with delays. *ComplexUs* 3, 231–251. doi: 10.1159/000110010
- Ahmad, J., Bourdon, J., Eveillard, D., Fromentin, J., Roux, O., and Sinoquet, C. (2009). Temporal constraints of a gene regulatory network: refining a qualitative simulation. *Biosystems* 98, 149–159. doi: 10.1016/j.biosystems.2009.05.002
- Ahmad, J., Niazi, U., Mansoor, S., Siddique, U., and Bibby, J. (2012). Formal modeling and analysis of the mal-associated biological regulatory network: insight into cerebral malaria. *PLoS ONE* 7:e33532. doi: 10.1371/journal.pone.0033532
- Ahmad, J., Roux, O., Bernot, G., Comet, J.-P., and Richard, A. (2008). Analysing formal models of genetic regulatory networks with delays. *Int. J. Bioinform. Res. Appl.* 4, 240–262. doi: 10.1504/IJBRA.2008.019573
- Alur, R. (1999). “Timed automata,” in *Computer Aided Verification*, eds N. Halbwachs and D. Peled (Berlin; Heidelberg: Springer), 8–22.
- Aslam, B., Ahmad, J., Ali, A., Paracha, R. Z., Tareen, S. H. K., Niazi, U., et al. (2014). On the modelling and analysis of the regulatory network of dengue virus pathogenesis and clearance. *Comput. Biol. Chem.* 53, 277–291. doi: 10.1016/j.compbiolchem.2014.10.003
- Ben Abdallah, E., Ribeiro, T., Magnin, M., Roux, O., and Inoue, K. (2017). Modeling delayed dynamics in biological regulatory networks from time series data. *Algorithms* 10:8. doi: 10.3390/a10010008
- Bernot, G., Cassez, F., Comet, J.-P., Delaplace, F., Müller, C., and Roux, O. (2007). Semantics of biological regulatory networks. *Electron. Notes Theor. Comput. Sci.* 180, 3–14. doi: 10.1016/j.entcs.2004.01.038
- Bernot, G., Comet, J.-P., and Khalis, Z. (2008). “Gene regulatory networks with multiplexes,” in *Proc of European Simulation and Modelling Conference, ESM’2008* (Le Havre), 423–432.
- de Jong, H. (2002). Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.* 9, 67–103. doi: 10.1089/10665270252833208
- Folschette, M., Paulevé, L., Inoue, K., Magnin, M., and Roux, O. (2012). “Concretizing the process hitting into biological regulatory networks,” in *Computational Methods in Systems Biology. CMSB 2012. Lecture Notes in Computer Science*, Vol. 7605, eds D. Gilbert and M. Heiner (Berlin; Heidelberg: Springer), 166–186.
- Folschette, M., Paulevé, L., Magnin, M., and Roux, O. (2015). Sufficient conditions for reachability in automata networks with priorities. *Theor. Comput. Sci.* 608, 66–83. doi: 10.1016/j.tcs.2015.08.040
- Henzinger, T. A., Ho, P.-H., and Wong-Toi, H. (1997). Hytech: a model checker for hybrid systems. *Int. J. Softw. Tools Technol. Transfer* 1, 110–122. doi: 10.1007/s100090050008
- Khalis, Z., Comet, J.-P., Richard, A., and Bernot, G. (2009). The smbionet method for discovering models of gene regulatory networks. *Genes Genomes Genomics* 3, 15–22. Available online at: <http://www.i3s.unice.fr/~comet/publications/RevueIntl/articleGGG2009-final.pdf>
- Paulevé, L. (2017). “Pint: a static analyzer for transient dynamics of qualitative networks with IPython interface,” in *CMSB 2017-15th Conference on Computational Methods for Systems Biology* (Darmstadt: Springer International Publishing), 309–316. doi: 10.1007/978-3-319-67471-1\_20
- Pauleve, L. (2018). Reduction of qualitative models of biological networks for transient dynamics analysis. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 15, 1167–1179. doi: 10.1109/TCBB.2017.2749225
- Paulevé, L., Andrieux, G., and Koepl, H. (2013). “Under-approximating cut sets for reachability in large scale automata networks,” in *Computer Aided Verification, Lecture Notes in Computer Science*, vol. 8044 eds N. Sharygina and H. Veith. (Berlin; Heidelberg: Springer), 69–84.
- Paulevé, L., Magnin, M., and Roux, O. (2011). “Refining dynamics of gene regulatory networks in a stochastic  $\pi$ -calculus framework,” in *Transactions on Computational Systems Biology XIII. Lecture Notes in Computer Science*, Vol 6575, eds C. Priami, R. J. Back, I. Petre, and E. de Vink (Berlin; Heidelberg: Springer), 171–191.
- Richard, A., Comet, J. P., and Bernot, G. (2006). “Formal methods for modeling biological regulatory networks,” in *Modern Formal Methods and Applications*, ed H. A. Gabbar (Dordrecht: Springer), 83–122.



- Saeed, M. T., Ahmad, J., Kanwal, S., Holowatyj, A. N., Sheikh, I. A., Zafar Paracha R., et al. (2016). Formal modeling and analysis of the hexosamine biosynthetic pathway: role of o-linked n-acetylglucosamine transferase in oncogenesis and cancer progression. *PeerJ* 4:e2348. doi: 10.7717/peerj.2348
- Sahin, Ö., Fröhlich, H., Löbke, C., Korf, U., Burmester, S., Majety, M., et al. (2009). Modeling ERBB receptor-regulated g1/s transition to find novel targets for *de novo* trastuzumab resistance. *BMC Syst. Biol.* 3:1. doi: 10.1186/1752-0509-3-1
- Shang, Y. (2012). A lie algebra approach to susceptible-infected-susceptible epidemics. *Electron. J. Differ. Equat.* 2012, 1–7. Available online at: <https://ejde.math.txstate.edu/Volumes/2012/233/abstr.html>
- Shang, Y. (2015a). Analytical solution for an in-host viral infection model with time-inhomogeneous rates. *Acta Phys. Pol. B* 46, 1567–1577. doi: 10.5506/APhysPolB.46.1567
- Shang, Y. (2015b). Deffuant model of opinion formation in one-dimensional multiplex networks. *J. Phys. A Math. Theor.* 48:395101. doi: 10.1088/1751-8113/48/39/395101
- Shang, Y. (2017). On the delayed scaled consensus problems. *Appl. Sci.* 7:713. doi: 10.3390/app7070713
- Shang, Y. (2018). Hybrid consensus for averager–copier–voter networks with non-rational agents. *Chaos Solit. Fract.* 110, 244–251. doi: 10.1016/j.chaos.2018.03.037
- Sheikh, I. A., Ahmad, J., and Saeed, M. T. (2016). “Modelling and simulation of biological regulatory networks by stochastic petri nets,” in *Proceedings of the World Congress on Engineering and Computer Science* (San Francisco, CA).
- Siebert, H., and Bockmayr, A. (2008). Temporal constraints in the logical analysis of regulatory networks. *Theor. Comput. Sci.* 391, 258–275. doi: 10.1016/j.tcs.2007.11.010
- Thieffry, D., and Thomas, R. (1995). Dynamical behaviour of biological regulatory networks—II. immunity control in bacteriophage lambda. *Bull. Math. Biol.* 57, 277–297.
- Thomas, R. (1978). Logical analysis of systems comprising feedback loops. *J. Theoret. Biol.* 73, 631–656. doi: 10.1016/0022-5193(78)90127-3
- Thomas, R. (2013). “Kinetic logic: a Boolean approach to the analysis of complex regulatory systems,” in *Proceedings of the EMBO Course “Formal Analysis of Genetic Regulation”*, (Brussels: Springer Science & Business Media).
- Thomas, R., and d’Ari, R. (1990). *Biological Feedback*. Boca Raton, FL: CRC Press.

**Conflict of Interest Statement:** The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Sheikh, Ahmad, Magnin and Roux. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.