Refining Dynamics of Gene Regulatory Networks in a Stochastic π-Calculus Framework

Loïc Paulevé, Morgan Magnin, Olivier Roux

Refining Dynamics of Gene Regulatory Networks in a Stochastic π-Calculus Framework

Loïc Paulevé, Morgan Magnin, Olivier Roux

IRCCyN, UMR CNRS 6597,
École Centrale de Nantes, France
{pauleve,magnin,roux}@irccyn.ec-nantes.fr

Abstract. In this paper, we introduce a framework allowing to model and analyse efficiently Gene Regulatory Networks in their temporal and stochastic aspects. The analysis of stable states and inference of René Thomas' discrete parameters derives from this logical formalism. We offer a compositional approach which comes with a natural translation to the Stochastic π-Calculus. The method we propose consists in successive refinements of generalized dynamics of Gene Regulatory Networks. We apply this method to the control of the differentiation in a Gene Regulatory Network generalizing metazoan segmentation processes.

1 Introduction

Modelling, analysis and numerical or stochastic simulations are a usual means to predict the behaviour of complex living systems such as interacting genes.

Regulations between genes (activation or inhibition) are generally represented by Gene Regulatory Network (GRN) graphs. However, a GRN graph is not enough to describe dynamics. In continuous frameworks such as ordinary differential equations, parameters for differential equations are needed. In logical (or qualitative) frameworks such as boolean or discrete networks, dynamics are driven by René Thomas' parameters [1].

Hybrid modelling brings quantitative aspect — such as temporal or stochastic parameters — to logical modelling. In the field of formal languages, κ language [2] or Stochastic π-Calculus [3,4,5] bring theoretical Computer Science frameworks for biological modelling. In the field of formal verifications of biological systems, frameworks like Time or Stochastic Petri Nets [6,7], Biocham [8], Timed Automata [9] and Linear Hybrid Modelling [10] bring the first bricks for verifying and controlling dynamics of such systems.

Inference of temporal and stochastic parameters is still challenging as the domain of parameters is continuous and its volume generally grows exponentially with the number of genes. Compositional approaches, inherent to process algebras, aspire at reducing this complexity by allowing a local reasoning.

Our aim is temporal parameters synthesis for verifying formal properties on hybrid models.

Our contribution consists in the introduction of both temporal and stochastic parameters into process algebra models of GRNs through a new Stochastic π-Calculus framework: the Process Hitting framework.

Starting from a GRN without any other parameters, its largest dynamics are expressed in Process Hitting and then are refined to match the expected behaviour. Such a refinement is achieved by constructing cooperativity between genes and by adding stable states. As we will show, detecting stable states of a Process Hitting is straightforward, as well as infering the René Thomas' discrete parameters K.

The Stochastic π-Calculus naturally brings time and stochasticity into our Process Hitting framework. We introduce a stochasticity absorption factor to favor either the temporal or stochastic aspect of reactions. The direct translation of Process Hitting to the Stochastic π-Calculus allows simulations of such models by softwares like BioSpi [3] or SPiM [11].

This paper is structured as follows. Section 2 introduces our framework and how it is used to build the generalized dynamics of a GRN. Section 3 presents dynamics refinement techniques and Sect. 4 shows how infering the René Thomas' parameters leading to such dynamics. Section 5 introduces temporal and stochastic parameters in our framework. Finally, Sect. 6 applies this method to a specific GRN involved in biological segmentations.

Notations Given a set S, $\overbrace{S \times \cdots \times S}^{n}$, will be abbreviated as S^n. If S is finite and countable, we note $|S|$ its cardinality. Given a n-tuple C, $C[x/y]$ refers to the n-tuple C within the element y has been substituted by x. Belonging and cartesian product for n-tuples are defined similarly to sets. $[x_i; x_{i+n}]$ refers to the interval $\{x_i, x_{i+1}, \ldots, x_{i+n}\}$. '$\wedge$' stands for the logical *and* connector.

2 Generalized Dynamics for Gene Regulatory Networks

First, we recall the basis of the René Thomas' discrete modelling framework from which we designed our refinement approach. This method is described in subsections 2.2 and 2.3. This leads to a straightforward translation into the π-Calculus which makes it possible to express generalized dynamics of GRNs.

2.1 Gene Regulatory Networks

GRNs are often described by interaction graphs where nodes are genes with activation and inhibition relations respectively represented by positive and negative edges.

In the discrete framework of René Thomas, each gene has at least two qualitative levels. The influence of an activating (resp. inhibiting) gene on its target depends on a threshold value: when the level of the gene is greater or equal than the threshold, the gene holds a positive (resp. negative) effect; when the level of the gene is lower than the threshold, the gene holds a negative (resp. positive) effect.

Definition 1 (Gene Regulatory Network Graph). *A* Gene Regulatory Network graph *is a triple* (Γ, E_+, E_-) *where* Γ *is the finite set of genes and* $a \xrightarrow{t} b \in E_+$ *(resp.* E_-*),* t *positive integer, if and only if the gene* a *above level* t *is an activator (resp. inhibitor) for* b*. We note* a_i *the level* i *of the gene* a*.*

Given a GRN graph (Γ, E_+, E_-), the maximum qualitative level for gene $a \in \Gamma$ is noted a_{l_a} where l_a is the highest threshold involved in its regulations:

$$l_a = max(\{t,\ \exists b \in \Sigma, a \xrightarrow{t} b \in E_+ \cup E_-\})\ . \tag{1}$$

We denote $levels_+(a,b)$ (resp. $levels_-(a,b)$) the set of levels of a where a effectively activates (resp. inhibits) b.

Definition 2 (Effective Levels). *If* $a \xrightarrow{t} b \in E_+$*,* $levels_+(a,b) = [a_t; a_{l_a}]$ *and* $levels_-(a,b) = [a_0; a_{t-1}]$*. If* $a \xrightarrow{t} b \in E_-$*,* $levels_+(a,b) = [a_0; a_{t-1}]$ *and* $levels_-(a,b) = [a_t; a_{l_a}]$*. Else* $levels_+(a,b) = levels_-(a,b) = \emptyset$*.*

2.2 The Process Hitting Framework

We want to describe the action of a gene at a given level on another one. If the gene a at a given level i is an activator for b, it has a positive action on b, meaning the level of b will tend to increase. Conversely if a at a level i' is an inhibitor for b, it has a negative action on b, the level of b will tend to decrease.

The action is "a at level i making b at level j increase (or decrease) to level k". We say a_i *hits* b_j to make it *bounce* to b_k and note this action $a_i \to b_j \barwedge b_k$. A Process Hitting is then a set of such actions between genes (or processes) at different levels.

Definition 3 (Action). *An* action *is noted* $a_i \to b_j \barwedge b_k$ *where* a_i *is the level of a process* a *and* $b_j \neq b_k$ *two levels of a given process* b*.* $a_i \to b_j$ *is the* hit *part, and* $b_j \barwedge b_k$ *the* bounce *part. When* $a_i = b_j$*, we use the term of* self-action *and call* a_i *a* self-hitting process level*.*

In this paper, processes are supposed to be at a unique level at any instant. Hence, we do not allow hits between different levels of a same process. The set of these living process levels gives the state of the Process Hitting.

Definition 4 (Process Hitting). *A* Process Hitting $\mathcal{PH}$ *is a triple* $(\Sigma, L, \mathcal{H})$*:*

- $\Sigma = \{a, b, \dots\}$ *is the finite countable set of processes,*
- $L = \prod_{a \in \Sigma} L_a$ *is the set of states for* $\mathcal{PH}$*, with* $L_a = \{a_0 \dots a_{l_a}\}$ *the finite and countable set of levels for process* $a \in \Sigma$ *and* l_a *a positive integer,* $a \neq b \Rightarrow a_i \neq b_j\ \forall (a_i, b_j) \in L_a \times L_b$*,*
- $\mathcal{H} = \{a_i \to b_j \barwedge b_k, \dots, (a,b) \in \Sigma^2,\ (a_i, b_j, b_k) \in L_a \times L_b \times L_b, b_j \neq b_k, a = b \Rightarrow a_i = b_j\}$*, is the finite set of actions.*

At a given state $s \in L$, an action $a_i \to b_j \barwedge b_k$ is applicable if both process levels a_i and b_j are present in s. When this action is played, the process level b_k replaces b_j.

Definition 5 (Next States). *Let $(\Sigma, L, \mathcal{H})$ be a Process Hitting and $s \in L$ be one of its states. The set of the next possible states for s are computed as follows:*

$$next(s) = \{s[b_k/b_j],\ \exists (a_i, b_j) \in s^2, \exists b_k \in L_b, a_i \to b_j \Rsh b_k \in \mathcal{H}\}\ .$$

Definition 6 (Stable state). *Let $\mathcal{PH} = (\Sigma, L, \mathcal{H})$ be a Process Hitting and $s \in L$ be a state, s is a* stable state *for $\mathcal{PH}$ if and only if $next(s) = \emptyset$.*

2.3 Graphical Representations of a Process Hitting

We set up two complementary graphical representations of a Process Hitting. The first one exhibits the actions between process levels, the second one points out the absence of hits between them. We finally define the State Graph of a Process Hitting. Figure 1 shows an instance for each of these three representations.

Given a Process Hitting $(\Sigma, L, \mathcal{H})$, its *Hypergraph* represents each action $a_i \to b_j \Rsh b_k \in \mathcal{H}$ by a directed hyperedge from a_i to b_k passing by b_j. The hit part (a_i to b_j) is drawn as a plain edge and the bounce part (b_j to b_k) as a dotted edge.

Definition 7 (Process Hitting Hypergraph). *The* Hypergraph *of a Process Hitting $(\Sigma, L, \mathcal{H})$ is a couple (P, A) where $P = \bigcup_{a \in \Sigma} L_a$ are the vertices and $A \subseteq P^3$ the directed hyperedges given by $A = \{(a_i, b_j, b_k), a_i \to b_j \Rsh b_k \in \mathcal{H}\}$.*

In the following we introduce a complementary representation we call *Hitless Graph*. It will allow us to obtain extra results such as the stable states of a Process Hitting (Sect. 3.2). The Hitless Graph of a Process Hitting $(\Sigma, L, \mathcal{H})$ relates two process levels of different processes if and only if they hit neither each other nor themselves. Vertices of a Hitless Graph may be split into $n \leq |\Sigma|$ partitions having no element inside related to each other: a partition is, for any process, a subset of its levels without self-actions. Such a graph is called n-partite.

Definition 8 (n-Partite Graph). *A graph $G = (V, E)$ is* n-partite *if and only if $V = \bigcup_{k=1}^{n} V_k,\ V_k \neq \emptyset$, $\forall 1 \leq k, k' \leq n, V_k \cap V_{k'} = \emptyset$ and $(a_i, b_j) \in E \Rightarrow$ $\exists 1 \leq k \neq k' \leq n,\ a_i \in V_k \wedge b_j \in V_{k'}$.*

Definition 9 (Hitless Graph). *Given a Process Hitting $\mathcal{PH} = (\Sigma, L, \mathcal{H})$, its Hitless Graph $\overline{\mathcal{PH}} = (V, E)$ is defined as a non-directed graph where the vertices V and edges E are computed as follows:*

$$V = \bigcup_{a \in \Sigma} \{a_i \in L_a, \forall a_{i'} \in L_a \nexists\, a_i \to a_i \Rsh a_{i'} \in \mathcal{H}\}$$

$$E = \{(a_i, b_j) \in V^2, \forall b_{j'} \in L_b \nexists\, a_i \to b_j \Rsh b_{j'} \in \mathcal{H} \wedge \forall a_{i'} \in L_a \nexists\, b_j \to a_i \Rsh a_{i'} \in \mathcal{H}\}\ .$$

Property 1. By construction of V and E, $\overline{\mathcal{PH}}$ is a n-partite graph, $n \leq |\Sigma|$, where each partition is a subset of levels for one and only one process and to each process corresponds at most one partition.

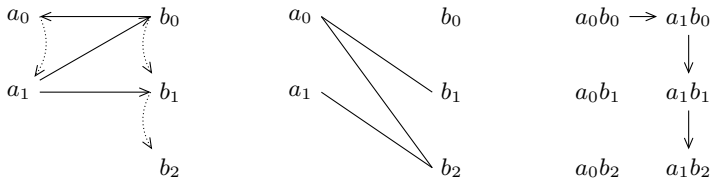
We also define the n-cliques of a graph which are subsets of n vertices such that each element is related to each other.

Definition 10 (n-Clique). *Given a graph $G = (V, E)$, $C \subseteq V$ is a $|C|$-clique of G if and only if $\forall (a_i, b_j) \in C^2, \{a_i, b_j\} \in E$.*

Property 2. n-cliques of a n-partite graph have one and only one vertex in each partition.

Finally, the *State Graph* of a Process Hitting represents the possible transitions between each couple of its states.

Definition 11 (State Graph). *Given a Process Hitting $(\Sigma, L, \mathcal{H})$, its* State Graph *is a directed graph $\mathcal{S} = (L, E \subseteq L^2)$ with $(s, s') \in E \Leftrightarrow s' \in next(s)$.*



(a) Hypergraph (b) Hitless Graph (c) State Graph

Fig. 1. Graphical representations for the Process Hitting $\mathcal{PH} = (\{a, b\}, \{a_0, a_1\} \times \{b_0, b_1, b_2\}, \mathcal{H})$ with $\mathcal{H} = \{b_0 \to a_0 \not\vdash a_1, a_1 \to b_0 \not\vdash b_1, a_1 \to b_1 \not\vdash b_2\}$.

2.4 From Process Hitting to the π-Calculus

A main advantage of our approach is its natural translation to the π-Calculus. In this subsection we propose a method to translate any Process Hitting into a π-Calculus expression.

We briefly present the fragment of the π-Calculus which is sufficient for translating a Process Hitting. The full syntax for π-Calculus and examples can be found in [5,12]. π-Calculus expressions compose two kinds of objects: independently defined processes and channels shared by some processes. A process P has the capability to output (resp. input) on a channel γ and then become P', noted $!\gamma.P'$ (resp. $?\gamma.P'$). Output and input are synchronized operations, i.e. an outputting process is blocked until another process inputs on the same channel. A process can also execute an internal action (τ), nil operation ($\mathbf{0}$) or one amongst several ($P' + P''$).

Let $\mathcal{PH} = (\Sigma, L, \mathcal{H})$ be a Process Hitting. For each process level a_i of $\mathcal{PH}$, a π-Calculus process A_i is defined as follows. For each action $a_i \to b_j \not\vdash b_k \in \mathcal{H}$ where $a \neq b$, a new channel γ_α is created. The process A_i has the ability to output on this channel and the process B_j has the ability to input on this channel so

as to become B_k (2). For each self-action $a_i \to a_i \Rsh a_j \in \mathcal{H}$, the process A_i has the ability to become A_j after performing an internal action τ_α (3).

$$A_i ::= \sum_{\substack{\alpha=a_i \to b_j \Rsh b_k \in \mathcal{H} \\ a \neq b}} !\gamma_\alpha.A_i + \sum_{\substack{\alpha=b_j \to a_i \Rsh a_k \in \mathcal{H} \\ a \neq b}} ?\gamma_\alpha.A_k \quad (2)$$

$$+ \sum_{\alpha=a_i \to a_i \Rsh a_k \in \mathcal{H}} \tau_\alpha.A_k \quad (3)$$

2.5 Generalized Dynamics for Gene Regulatory Networks

Our method to analyse GRNs takes benefit from the use of refinement techniques. Starting from the largest set of possible dynamics for the GRN, we gradually take into account only the specified behaviours and exclude the other ones, thus leading to a restrictive process.

We call this largest set of dynamics the *generalized dynamics* for the GRN graph. It is described by the following rules: the level of a gene increases (resp. decreases) if and only if at least one of its activators (resp. inhibitors) is present. The absence of activators is equivalent to the presence of one inhibitor.

Let $\mathcal{G} = (\Gamma, E_+, E_-)$ be a GRN graph. For all $(a, b) \in \Gamma^2$, we build the set of actions $\mathcal{H}_a^b$ from a to b reflecting the rules above:

- If $a \xrightarrow{t} b \in E_+$, all process levels of a below the threshold t hit all process levels of b but b_0 to make them decrease to the level below. Moreover, all process levels of a above the threshold t hit all process levels of b but b_{l_b} to make them increase to the level above:

$$\begin{aligned}\mathcal{H}_a^b = &\{a_i \to b_j \Rsh b_{j-1},\ 0 \leq i < t, 1 \leq j \leq l_b\} \\ &\cup \{a_{i'} \to b_{j'} \Rsh b_{j'+1},\ t \leq i' \leq l_a, 0 \leq j' < l_b\}\ .\end{aligned}$$

- If $a \xrightarrow{t} b \in E_-$, the actions are defined similarly to the previous case except for the bounce directions which are reversed:

$$\begin{aligned}\mathcal{H}_a^b = &\{a_i \to b_j \Rsh b_{j+1},\ 0 \leq i < t, 0 \leq j' < l_b\} \\ &\cup \{a_{i'} \to b_{j'} \Rsh b_{j'-1},\ t \leq i \leq l_a, 1 \leq j \leq l_b\}\ .\end{aligned}$$

- If $b = a$ and $\nexists c \in \Gamma,\ c \xrightarrow{t} b \in E_- \cup E_+$, gene b lives in absence of activators: all process levels of b but b_0 hit themselves to decrease to the level below.

$$\mathcal{H}_b^b = \{b_i \to b_i \Rsh b_{i-1},\ 1 \leq i \leq l_b\}\ .$$

- Obviously, if $a \xrightarrow{t} b \notin E_- \cup E_+$ for any t and the previous case does not hold, we define $\mathcal{H}_a^b = \emptyset$.

The Process Hitting for the generalized dynamics of $\mathcal{G}$ is given by

$$\mathcal{PH} \ = \ (\Gamma,\ \prod_{a \in \Gamma} \{a_0, \ldots, a_{l_a}\},\ \bigcup_{(a,b) \in \Gamma^2} \mathcal{H}_a^b)\ .$$

3 Refining Dynamics of Gene Regulatory Networks

We present two methods which aim at narrowing the set of dynamics of a Process Hitting for a GRN: the first one is based on cooperativity between genes, the other one deals with the knowledge of the stable states.

3.1 Cooperative Hits

Given two genes c and f regulating a gene a, the action of c on a may depend on the level of f: there exists a cooperativity between c and f on a. In discrete frameworks, the cooperativity is often described by a boolean function between genes levels [13]. We show how to build cooperativity within Process Hitting.

Let $(\Sigma, L, \mathcal{H})$ be a Process Hitting and $\sigma \subset \Sigma$ be a set of *cooperating processes* on a given process level a_k to make it bounce to $a_{k'}$. We call $S = \prod_{z\in\sigma} L_z$ the set of all states for the cooperating processes. We define $\top \subset S$ the subset of states where the cooperativity is effective.

We set σ as the *cooperative process* having process levels $L_\sigma = \{\sigma_\varsigma, \forall \varsigma \in S\}$. When a process $z \in \sigma$ is at level z_i it hits process levels σ_ς where $z_i \notin \varsigma$ to make it bounce to $\sigma_{\varsigma[z_i/z_j]}, z_j \in \varsigma$. We denote $\mathcal{H}_\sigma$ the set of such actions (4). In this way, the level of process σ will reflect the current state of its representatives.

The cooperativity between σ processes is added into the Process Hitting by replacing σ processes hits $\mathcal{H}_{coop}$ to c_k (5) by hits $\mathcal{H}'_{coop}$ from levels of the cooperative process σ selected in $\top$ (6).

$$\mathcal{H}_\sigma = \{z_i \to \sigma_\varsigma \,\vec{\vdash}\, \sigma_{\varsigma[z_i/z_j]},\ \forall z \in \sigma, \forall (z_i, z_j) \in L_z^2, \forall \sigma_\varsigma \in L_\sigma, z_j \in \varsigma\} \quad (4)$$

$$\mathcal{H}_{coop} = \{z_i \to a_k \,\vec{\vdash}\, a_{k'} \in \mathcal{H},\ \forall z \in \sigma\} \quad (5)$$

$$\mathcal{H}'_{coop} = \{\sigma_\varsigma \to a_k \,\vec{\vdash}\, a_{k'},\ \forall \varsigma \in \top\}\ . \quad (6)$$

The resulting Process Hitting is $(\Sigma \cup \{\sigma\}, L \times L_\sigma, (\mathcal{H} \setminus \mathcal{H}_{coop}) \cup \mathcal{H}_\sigma \cup \mathcal{H}'_{coop})$.

Example 1. Let $(\{f, c, a\}, \{f_0, f_1\} \times \{c_0, c_1\} \times \{a_0, a_1\}, \mathcal{H})$ be a Process Hitting where $\{f_1 \to a_0 \,\vec{\vdash}\, a_1, c_0 \to a_0 \,\vec{\vdash}\, a_1\} \subset \mathcal{H}$. The creation of a cooperativity between f_1 and c_0 on a_0 ($\sigma = \{f, c\}$, $\top = \{f_1 c_0\}$) is illustrated by Fig. 2.

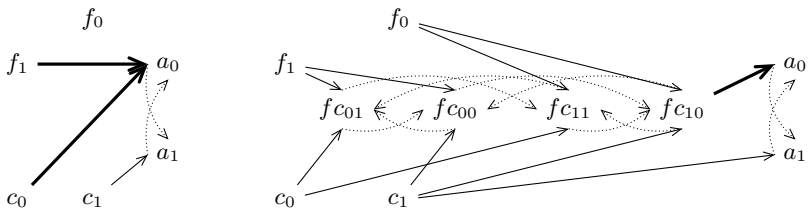


Fig. 2. Construction of a cooperative hit between f_1 and c_0 on a_0 (thick lines): $\sigma = \{f, c\}$, $\top = \{f_1 c_0\}$

3.2 Stable State Pattern

Given a Process Hitting $(\Sigma, L, \mathcal{H})$, we prove the $|\Sigma|$-cliques of its Hitless Graph are exactly its stable states. Thus, stable states may be created by removing from the Process Hitting the very hits that make such such patterns appear.

Theorem 1. *Let $\mathcal{PH} = (\Sigma, L, \mathcal{H})$ be a Process Hitting and $\overline{\mathcal{PH}}$ its Hitless Graph. A state $s \in L$ is stable if and only if s is a $|\Sigma|$-clique for $\overline{\mathcal{PH}}$.*

Proof. By definition, $next(s) = \emptyset$ if and only if there is no hit between any couple of process levels in s. This is equivalent to have s a clique of $\overline{\mathcal{PH}}$.

Figure 3 shows an instance of Process Hitting having one stable state.

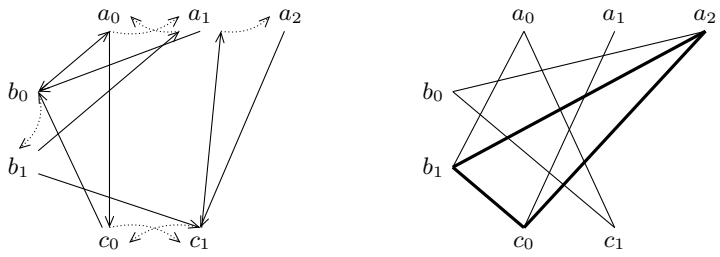
We outline an algorithm for finding the n-cliques of a Hitless Graph $\overline{\mathcal{PH}} = (V, E)$ where $n = |\Sigma|$.

Thanks to Prop. 1, we split V into n partitions corresponding to each process: $V = \cup_{a \in \Sigma} V_a,\ V_a \subseteq L_a$. If one of this partition is empty, there can not be n-cliques as it requires to have at least one vertex in each partition. We will assume $V_a \neq \emptyset,\ \forall a \in \Sigma$.

For each partition $a \in \Sigma$ and each vertex $a_i \in V_a$, we define $E^b_{a_i} = \{b_j \in V_b,\ (a_i, b_j) \in E\}$ for each other partition $b \in \Sigma, b \neq a$, the set of vertices in V_b related to a_i. If there exists $b \in \Sigma$ such that $E^b_{a_i} = \emptyset$, the vertex a_i is removed from candidates as it can not belong to a n-clique. Finally, we set $E^a_{a_i} = \{a_i\}$.

Once this pruning is performed, we enumerate potential n-cliques. To reduce this enumeration, we choose the partition a sharing the least number of edges. For each vertex $a_i \in V_a$ we test for all $s \in \prod_{b \in \Sigma} E^b_{a_i}$ if s is a clique of $\overline{\mathcal{PH}}$.

For instance in Fig. 3(b), a_1 is removed from the Hitless Graph ($E^b_{a_1} = \emptyset$), the partition associated to c is chosen (involved into only 4 edges), two states are tested: $a_0 b_0 c_1$ and $a_2 b_1 c_0$ and the latter reveals to be the only 3-clique.



(a) Hypergraph (b) Hitless Graph

Fig. 3. A Process Hitting represented by its Hypergraph (a) and its Hitless Graph (b). The Hitless Graph contains only one 3-clique between a_2, b_1 and c_0 (thick lines): this is the only stable state of this system.

4 From Process Hitting to René Thomas' Parameters

A René Thomas' discrete parameter gives the attractor levels for a gene when its regulators are in a given configuration. Many frameworks and tools dedicated to the study of GRNs take the full set of René Thomas' parameters as essential input. In this section, we give a formal method to infer René Thomas' parameters for a GRN modelled in the Process Hitting framework.

Let (Γ, E_+, E_-) be a GRN graph. A René Thomas' parameter $K_{a,A,B}$, $a \in \Gamma, A \cup B \subseteq \Gamma,\ A \cap B = \emptyset$, gives the interval of attracting levels for a when genes in A are activating a and genes in B are inhibiting a. In this configuration, if the level of a is in $K_{a,A,B}$, then it will never change; otherwise the level of a will tend to a level in $K_{a,A,B}$.

Let $(\Sigma, L, \mathcal{H})$ be a Process Hitting where processes are standing either for genes or for cooperative processes, i.e. $\Sigma = \Gamma \cup \{\sigma^1, \ldots, \sigma^u\}$ with $\forall 1 \leq v \leq u,\ \sigma^v \subset \Gamma$. Let $K_{a,A,B}$ be the René Thomas' parameter to infer. For each process $b \in A \cup B$, we define its context $C^b_{a,A,B}$ as the subset of levels L_b imposed by the René Thomas' parameter: if $b \in A$ (resp. B) only positive (resp. negative) effective levels (Def. 2) are allowed. For each process $b \in \Gamma$ not regulating a (i.e. $b \notin A \cup B$), its context $C^b_{a,A,B}$ is simply L_b (7). The context $C^\sigma_{a,A,B}$ for cooperative processes $\sigma \in \{\sigma^1, \ldots, \sigma^u\}$ is the set of states of its representatives in their context (8).

$$\forall b \in \Gamma,\ C^b_{a,A,B} = \begin{cases} levels_+(b,a) & \text{if } b \in A, \\ levels_-(b,a) & \text{if } b \in B, \\ L_b & \text{else.} \end{cases} \tag{7}$$

$$\forall \sigma \in \{\sigma^1, \ldots, \sigma^u\},\ C^\sigma_{a,A,B} = \{\sigma_\varsigma,\ \forall \varsigma \in \prod_{b \in \sigma} C^b_{a,A,B}\}\ . \tag{8}$$

We denote $\mathcal{H}_{a,A,B}$ the subset of the set of actions $\mathcal{H}$ on a that may be performed by any process in its context (9). A process level of a is reachable if it belongs to the context of a or is the result of any action in $\mathcal{H}_{a,A,B}$. The set of such process levels is noted $L^?_{a,A,B}$ (10). The set of reachable process levels of a not hit by any processes in their context is noted $L^*_{a,A,B}$ (11). Thus, as long as the processes are in their context, if a is at any level of $L^*_{a,A,B}$, its level will not change. We call $L^*_{a,A,B}$ the set of *focal levels* for a.

$$\mathcal{H}_{a,A,B} = \{b_i \to a_j \Rsh a_k \in \mathcal{H},\ b_i \in C^b_{a,A,B} \wedge a_j \in C^a_{a,A,B}\} \tag{9}$$

$$L^?_{a,A,B} = C^a_{a,A,B} \cup \{a_k,\ \forall b_i \to a_j \Rsh a_k \in \mathcal{H}_{a,A,B}\} \tag{10}$$

$$L^*_{a,A,B} = L^?_{a,A,B} \setminus \{a_j,\ \forall b_i \to a_j \Rsh a_k \in \mathcal{H}_{a,A,B}\}\ . \tag{11}$$

Finally, we check that the focal levels are attractors, i.e. all actions $\mathcal{H}_{a,A,B}$ make a bouncing in the direction of the focal levels. If such a condition is satisfied, the focal levels correspond to the value of the requested René Thomas' parameter. We point up that all these operations are linear with the number of actions in the Process Hitting.

Condition 1 (Focal levels are attractors).
$\forall b_i \to a_j \Rsh a_k \in \mathcal{H}_{a,A,B},\ \forall a_f \in L^*_{a,A,B},\ |f-k| < |f-j|$.

Property 3. If $L^*_{a,A,B}$ satisfies Cond. 1, $L^*_{a,A,B}$ is an interval.

Proof. If $L^*_{a,A,B} = \{a_f, \ldots, a_{f'}\}$ is not an interval, there exists $b_i \to a_j \Rsh a_k \in \mathcal{H}_{a,A,B}$ such that $f < j < f'$. If Cond. 1 applies, we have $|f-k| < |f-j| \Rightarrow k < j \Rightarrow |f'-k| > |f'-j|$ which contradicts Cond. 1.

Theorem 2. *If $L^*_{a,A,B} \neq \emptyset$ and Cond. 1 holds, then $K_{a,A,B} = L^*_{a,A,B}$.*

Proof. By construction of $L^*_{a,A,B}$ and application of Cond. 1 and Prop. 3, it immediately appears that if $L^*_{a,A,B} \neq \emptyset$, it is the set of attracting levels for a.

Consequently, there might exist configurations without any correspondence with René Thomas' parameters. First, $L^*_{a,A,B} = \emptyset$ means the gene a is unstable in the fixed context, i.e. its level is changing forever. Second, Cond. 1 is violated when there exists opposite focal levels — i.e. the fate of a is not deterministic — or focal levels are not attractive also unveiling unstable behaviours.

One of the main reasons for non-determinism of Process Hitting is the absence of cooperativity between hits to a same target which may then independently be bounced to both higher and lower levels. We leave as an open question the problem to know whether such unstable and/or non-deterministic dynamics are biologically relevant.

5 Temporal and Stochastic Parameters

Further dynamics refinements may be achieved by taking into account the temporal and stochastic dimensions of biological reactions. On the one hand, we may consider the probability of a reaction to occur at a given state. By introducing stochastic parameters into discrete models, we aim at computing the probability of observing an expected behaviour. On the other hand, because they are faster, some reactions always apply before others. By introducing temporal parameters into discrete models, we aim at reducing their dynamics to match such behaviours.

5.1 From Process Hitting to the Stochastic π-Calculus

The Stochastic π-Calculus [14] adds the capability to attach *use rates* to channels and internal actions of the π-Calculus. This gives a natural introduction for temporal and stochastic aspects in our Process Hitting framework.

A use rate controls both the duration and the probability of a reaction (communication on a channel or internal action). It is associated to a probability distribution for firing reaction along the time. The usual probability distribution is the exponential one, allowing efficient simulations through a Gillespie-like algorithm [11,15]. This is the one we consider for the rest of this paper.

The probability along time t of firing a reaction with use rate r is given by $F(t) = 1 - e^{-rt}$. The average duration of this reaction is r^{-1} with a variance of r^{-2}. When x reactions are possible having use rates of $r_1, \ldots, r_x$ respectively, the probability that the y^{th} reaction is fired is given by $\frac{r_y}{r_1+\cdots+r_x}$.

The translation of Process Hitting $(\Sigma, L, \mathcal{H})$ into the Stochastic π-Calculus is the same as the one presented in Sect. 2.4. Additionally, to each channel γ_α, or internal action τ_α, a use rate r_α is attached.

5.2 Stochasticity Absorption

Use rates are both temporal and stochastic parameters. Nonetheless, these two aspects are closely tied: the lower a use rate is, the higher the variance around its mean duration is. We introduce a *stochasticity absorption factor* to control this variance to favour either the stochastic or the temporal behaviour of an action.

We propose to replace the exponential distribution of a reaction with a rate r by the distribution of the sum of sa random variables each having an exponential distribution of parameter $r.sa$. The resulting probability distribution is also known as the *Erlang distribution*. The average duration is unchanged: $(r.sa)^{-1}sa = r^{-1}$, but the variance is divided by sa: $(r.sa)^{-2}sa = r^{-2}sa^{-1}$. sa stands for the stochasticity absorption factor. Based on the previously presented translation from the Process Hitting to the Stochastic π-Calculus, we supply a simple method to achieve this stochasticity absorption factor which do not require to adapt simulation algorithms based on the memoryless property of the exponential law [16].

Basically, to each channel γ_α, or internal action τ_α, a use rate r_α and a stochasticity absorption factor sa_α is attached. To each component α of the sum defined by the process A_i (2),(3), a counter c_α is attached, initially, $c_\alpha = 1$. This counter is given as a parameter for the process A_i. As long as this counter is not equal to sa_α, A_i is restarted and the counter is incremented by one. When the counter reaches the stochasticity absorption factor value, the next process replaces A_i, having all its counter reset to 1. Let $(\Sigma, L, \mathcal{H})$ be a Process Hitting, for each process level a_i of $\mathcal{PH}$, a π-Calculus process A_i is defined as follows.

$$A_i(\tilde{c}) ::= \sum_{\substack{\alpha=a_i \rightarrow b_j \,\Rsh\, b_k \in \mathcal{H} \\ a \neq b}} !\gamma_\alpha.A_i(\tilde{c})$$

$$+ \sum_{\substack{\alpha=b_j \rightarrow a_i \,\Rsh\, a_k \in \mathcal{H} \\ a \neq b}} [c_\alpha < sa_\alpha]?\gamma_\alpha.A_i(\tilde{c}[c_\alpha + 1]) + [c_\alpha = sa_\alpha]?\gamma_\alpha.A_k(\tilde{1})$$

$$+ \sum_{\substack{\alpha=a_i \rightarrow a_i \,\Rsh\, a_k \in \mathcal{H}}} [c_\alpha < sa_\alpha]\tau_\alpha.A_i(\tilde{c}[c_\alpha + 1]) + [c_\alpha = sa_\alpha]\tau_\alpha.A_k(\tilde{1})$$

where $\hat{c} = c_1, \ldots, c_n$ with $n = |\{b_j \rightarrow a_i \,\Rsh\, a_k \in \mathcal{H}\}|$. $\hat{c}[c_\alpha + 1] = c_1, \ldots, c_\alpha + 1, \ldots, c_n$. $A_k(\tilde{1})$ is an abbreviation for the recursive call to A_k with all parameters set to 1. $[cond]\pi.P$ stands for an action π enabled only when $cond$ is satisfied.

6 Application: Metazoan Segmentation

In this section, we illustrate our method and its benefits on a case study in which our aim is to control the final state of the corresponding GRN. The GRN we chose has been established *in silico* by François et al. [17] but in a differential equations framework. It aims at generalizing a common motif present in biological segmentation networks such as the *Drosophila*.

The GRN (Fig. 4(a)) is composed of three genes. A wavefront gene f activates the gap-gene a whose products are responsible for stripes. Gene f also activates a gene c whose products repress the gene a. The auto-inhibition of c generalizes a chain of repressors on a. The apparition of stripes has to be regular. We attach to each gene two qualitative levels (missing or present) — for instance c_0 (absence) and c_1 (presence) are levels of c. When f switches off, c goes to level c_0 and a has two fates, ending either at level a_0 or a_1. We are interested in the control of the final level of a.

We first compute the Process Hitting for generalized dynamics of the GRN (Sect. 2), Fig 4(b) shows its hypergraph. The specification of dynamics implies two cooperative hits in the Process Hitting: first, c_0 needs products of f to bounce to level c_1; second, expression of a only increases if both f activates it (i.e. f is at level f_1) and c does not inhibit it (i.e. c is at level c_0). Consequently, we create a cooperative process fc reflecting the state of f and c (Sect. 3.1) and replace the independent hits from c_0 and f_1 to c_0 and a_0 by hits from fc_{10}. The resulting Process Hitting is represented in Fig. 5(a).

By looking at the Hitless Graph of the Process Hitting (Fig. 5(b)), only one stable state is present: $f_0c_0fc_{00}a_0$. The stability of the state $f_0c_0fc_{00}a_1$ is controlled by the absence of inhibition by f_0 on a_1. By removing the action $f_0 \to a_1 \not\Uparrow a_0$ from the Process Hitting, we make the state $f_0c_0fc_{00}a_1$ stable. The full set of corresponding René Thomas' parameters for the genes a and c is inferred by applying the method depicted in Sect. 4. We get:

$$\begin{array}{lll} K_{a,\emptyset,\{a,c,f\}} = 0 & K_{a,\{a,c\},\{f\}} = 1 & K_{c,\emptyset,\{c,f\}} = 0 \\ K_{a,\{a\},\{c,f\}} = 0 & K_{a,\{a,f\},\{c\}} = 0 & K_{c,\{c\},\{f\}} = 0 \\ K_{a,\{c\},\{a,f\}} = 0 & K_{a,\{c,f\},\{a\}} = 1 & K_{c,\{f\},\{c\}} = 0 \\ K_{a,\{f\},\{a,c\}} = 0 & K_{a,\{a,c,f\},\emptyset} = 1 & K_{c,\{c,f\},\emptyset} = 1 \ . \end{array}$$

We are interested in controlling the final level of a — either a_0 or a_1 — when f goes down to f_0. Looking at the Process Hitting hypergraph on Fig. 5(a) and considering f is at level f_0, we deduce that the more c_1 is present, the more a_1 may be hit by c_1 to end at level a_0; similarly, the more fc_{10} is present, the more a_0 may be hit by it to end at level a_1. We tune actions only triggered by f_0: we reduce the presence of c_1 by increasing the rate of the action $f_0 \to c_1 \not\Uparrow c_0$ and extend the presence of fc_{10} by reducing the rate of $f_0 \to fc_{10} \not\Uparrow fc_{00}$. This leads to an increase of the probability for a to end at level a_1.

Finally, to obtain regular stripes, we set a high stochasticity absorption factor to actions responsible of the level changes for c and a when f_1 is present. Figure 6 plots the evolution of the genes a,c and f during a simulation under SPiM [18]

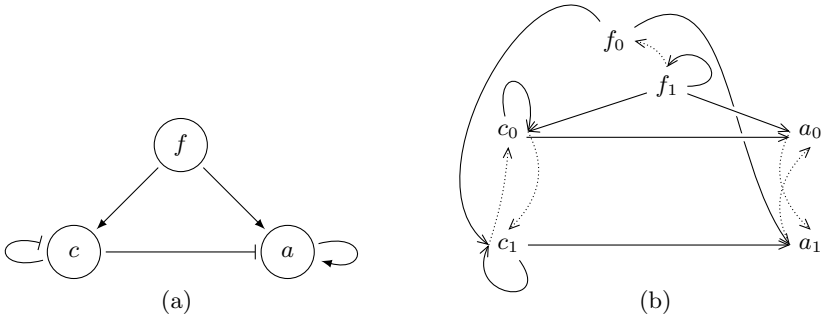


Fig. 4. The starting Gene Regulatory Network graph (a), arrow-ended edges represent the positive regulations, and bar-ended edges the negative ones. All regulation thresholds are 1. The Process Hitting (b) for its generalized dynamics. Cooperativity between f_1 and c_0 on a_0 and c_0 will be applied in the same way as in example. 1.

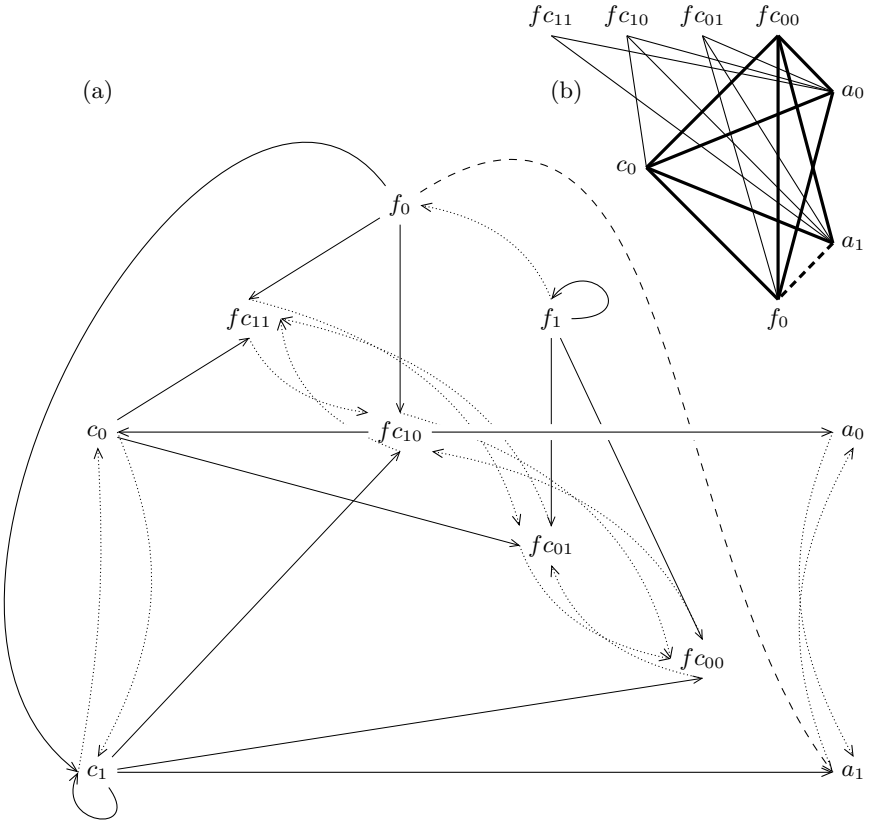


Fig. 5. The final Process Hitting (a) resulting from the refinement of the generalized dynamics depicted on Fig. 4(b). Cooperativity between f_1 and c_0 on a_0 and c_0 has been built in the same way as in example 1. Absence of hit from f_0 to a_1 (dashed lines) controls the presence of the relation between f_0 and a_1 in the Hitless Graph (b). If such a relation exists, two 4-cliques are presents: $c_0 f_0 fc_{00} a_0$ and $c_0 f_0 fc_{00} a_1$ (thick lines).

of the Process Hitting illustrated by Fig. 5(a) with initial state $f_1 c_0 f c_{10} a_0$ and a fast rate for the action $f_0 \to c_1 \Rsh c_0$ compared to the rate of $c_1 \to a_1 \Rsh a_0$. We observe that f_0 hits c_1 before c_1 had time to hit a_1: the final state is then $f_0 c_0 f c_{00} a_1$.

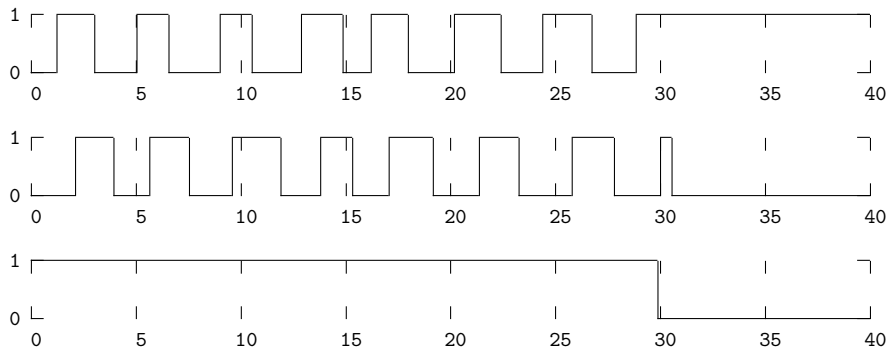


Fig. 6. Simulation of the Process Hitting for segmentation: evolution of the expressions of the gap-gene a (top), the autonomous clock c (middle) and the wavefront f.

7 Conclusion

We introduced the Process Hitting framework for modelling qualitative dynamics of GRNs with temporal and stochastic features. Temporal and stochastic parameters determine probabilities, durations and temporal variance of reactions in the model. We exhibited a direct translation from Process Hitting to the Stochastic π-Calculus. Detection of stable states and inference of René Thomas' parameters for dynamics derive from this framework. The methods we offered work by successive refinements of generalized dynamics for GRNs, by specifying both the cooperativity between genes and the expected stable states. We illustrated this method by inferring temporal parameters for the dynamics of a GRN generalizing metazoan segmentation processes (with the aim of controlling its final state).

Our results strongly rely on the compositionality of the framework and the presence of Process Hitting structure patterns. Thanks to these patterns, it become possible to lead a local analysis, which has the major advantage to prevent us from exploring the full state and parameter space.

In future works, we aim at identifying more Process Hitting patterns leading to the emergence of particular behaviours (e.g. oscillations) and especially hybrid patterns coupling both discrete structure and continuous temporal and stochastic parameters. Finally, techniques have to be developed to automate temporal and stochastic parameters inference for Stochastic π-calculus models of GRNs.

Supplementary Material

The Process Hitting compiler to SPiM and presented models are available at the following URL: http://www.irccyn.ec-nantes.fr/~pauleve/tcsb09-suppl.tar.gz.

References

1. Richard, A., Comet, J.P., Bernot, G.: Formal Methods for Modeling Biological Regulatory Networks. In: Modern Formal Methods and Applications. (2006) 83–122
2. Danos, V., Feret, J., Fontana, W., Harmer, R., Krivine, J.: Rule-Based Modelling of Cellular Signalling. In: CONCUR 2007 Concurrency Theory. (2007) 17–41
3. Priami, C., Regev, A., Shapiro, E., Silverman, W.: Application of a stochastic name-passing calculus to representation and simulation of molecular processes. Inf. Process. Lett. **80**(1) (2001) 25–31
4. Kuttler, C., Niehren, J.: Gene regulation in the pi calculus: Simulating cooperativity at the lambda switch. Transactions on Computational Systems Biology **4230**(VII) (November 2006) 24–55
5. Blossey, R., Cardelli, L., Phillips, A.: A compositional approach to the stochastic dynamics of gene networks. Transactions in Computational Systems Biology **3939** (Jan 2006) 99–122
6. Popova-Zeugmann, L., Heiner, M., Koch, I.: Time petri nets for modelling and analysis of biochemical networks. Fundamenta Informaticae **67**(1) (2005) 149–162
7. Heiner, M., Gilbert, D., Donaldson, R.: Petri Nets for Systems and Synthetic Biology. In: Formal Methods for Computational Systems Biology. (2008) 215–264
8. Rizk, A., Batt, G., Fages, F., Soliman, S.: On a Continuous Degree of Satisfaction of Temporal Logic Formulae with Applications to Systems Biology. In: Computational Methods in Systems Biology. (2008) 251–268
9. Siebert, H., Bockmayr, A.: Incorporating Time Delays into the Logical Analysis of Gene Regulatory Networks. In: Computational Methods in Systems Biology. (2006) 169–183
10. Ahmad, J., Bernot, G., Comet, J.P., Lime, D., Roux, O.: Hybrid modelling and dynamical analysis of gene regulatory networks with delays. Complexus **3**(4) (2006) 231–251
11. Phillips, A., Cardelli, L.: Efficient, correct simulation of biological processes in the stochastic pi-calculus. In: Computational Methods in Systems Biology. Volume 4695 of LNCS., Springer (Sep 2007) 184–199
12. Milner, R.: A calculus of mobile processes, parts. I and II. Information and Computation **100** (1992) 1–77
13. Bernot, G., Comet, J.P., Khalis, Z.: Gene regulatory networks with multiplexes. In: European Simulation and Modelling Conference Proceedings. (Oct 2008) 423–432
14. Priami, C.: Stochastic pi-Calculus. The Computer Journal **38**(7) (1995) 578–589
15. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. The Journal of Physical Chemistry **81**(25) (1977) 2340–2361
16. Priami, C.: Stochastic π-calculus with general distributions. In: Proc. of the 4th Workshop on Process Algebras and Performance Modelling, CLUT. (1996) 41–57
17. Francois, P., Hakim, V., Siggia, E.D.: Deriving structure from evolution: metazoan segmentation. Mol Syst Biol **3** (Dec 2007)
18. Phillips, A.: SPiM. http://research.microsoft.com/~aphillip/spim.