A Reduction of Logical Regulatory Graphs Preserving Essential Dynamical Properties

Aurélien Naldi[1], Elisabeth Remy[2], Denis Thieffry[1,3], and Claudine Chaouiya[1,4]

[1] TAGC Inserm U928 - Université de la Méditerranée, Marseille, France
[2] IML UMR 6206, Marseille, France
[3] CONTRAINTES, INRIA Paris Rocquencourt, France
[4] IGC, Instituto Gulbenkian de Ciência, Oeiras, Portugal

Abstract. To cope with the increasing complexity of regulatory networks, we define a reduction method for multi-valued logical models.

Starting with a detailed model, this method enables the computation of a reduced model by iteratively "hiding" regulatory components. To keep a consistent behaviour, the logical rules associated with the targets of each hidden node are actualised to account for the (indirect) effects of its regulators.

The construction of reduced models ensures the preservation of a number of dynamical properties of the original model. In particular, stable states and more complex attractors are conserved. More generally, we focus on the relationship between the attractor configuration of the original model and that of the reduced model, along with the issue of attractor reachability.

The power of the reduction method is illustrated by its application to a multi-valued model of the segment-polarity network Controlling segmentation in the fly Drosophila melanogaster.

Keywords: Regulatory networks, logical modelling, model reduction, decision diagrams, regulatory circuits, stable states, complex attractors, Drosophila development, segmentation.

1 Introduction

Biological data generation and integration efforts result in the delineation of ever more comprehensive and complex regulatory networks involved in the control of numerous processes. Consequently, current modelling and analysis approaches are reaching their limits in terms of the number and variety of components and interactions that can be efficiently considered. This is true for quantitative frameworks (*e.g.*, differential or stochastic models), as well as for qualitative approaches. Indeed, although logical modelling enables to handle networks comprising relatively large numbers of components (see *e.g.* [1,2]), the size of the state space grows exponentially with the number of regulatory nodes.

P. Degano and R. Gorrieri (Eds.): CMSB 2009, LNBI 5688, pp. 266–280, 2009.

One way to handle this problem consists in developing compositional approaches to compute the dynamical properties of comprehensive networks, relying on the knowledge of the properties of simpler sub-systems or modules. A complementary approach consists in reducing large systems, by focusing on the most relevant components and redefining their interactions in order to preserve relevant dynamical properties (*e.g.* stable states).

Most often, modellers intuitively and manually reduce regulatory networks to address specific questions. Such empirical reductions have several drawbacks: (i) the process is error prone and limited to relatively simple cases; (ii) the maintenance of different versions of a model (complete and reduced) is cumbersome; (iii) storing the sole reduced model leads to the loss of relevant biological information.

These considerations led us to develop a reliable, automated reduction method in the context of a logical modelling framework. In this respect, we lean on the software *GINsim*, which facilitates the definition of comprehensive logical regulatory graphs, as well as the analysis of their dynamical properties [3,4]. Established on firm mathematical bases, our reduction method allows the user to select nodes to be made implicit and to perform dynamical analyses on reduced model versions, which preserve relevant topological and dynamical properties.

The paper is organised as follows. Section 2 recalls the definitions of logical regulatory graphs and of the associated state transition graphs. Next, the reduction method is defined in Section 3. Relationships between the dynamical behaviour of the original model and that of the reduced model are delineated in Section 4. A multi-valued logical model of the segment-polarity network is then used to demonstrate the power of the proposed reduction method in Section 5. The paper ends with conclusions and further prospects.

All models presented in this paper can be opened, edited, simulated, and analysed with *GINsim*, which implements the logical formalism and the reduction method presented here.

2 Logical Modelling of Regulatory Networks

Our modelling approach leans on the generalised logical formalism initially developed by R. Thomas *et al.* [5,6,3]. In this context, a regulatory network and its dynamics are both represented in terms of oriented graphs.

2.1 Regulatory Graphs

Definition 1. A logical regulatory graph (LRG) *is a directed labelled multigraph* $\mathcal{R} = (\mathcal{G}, \mathcal{M}ax, \Gamma, \Theta, \mathcal{K})$ *where,*

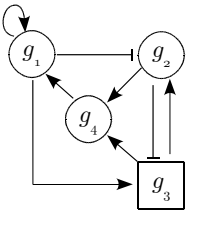
- $\mathcal{G} = \{g_1, \ldots, g_N\}$ *is the set of nodes, representing* regulatory components.
- $\mathcal{M}ax : \mathcal{G} \rightarrow \mathbb{N}^*$ *associates a maximum level* $\mathcal{M}ax(g_i) = \mathcal{M}ax_i$ *to node* g_i. *The* current level *of* g_i*, denoted* x_i*, takes its values in* $\mathcal{D}_i = \{0, \ldots, \mathcal{M}ax_i\}$.
- Γ *is the set of arcs, defined as a finite multiset of ordered pairs of elements of* $\mathcal{G}$ *representing* regulatory interactions. *If* $\mathcal{M}ax_i > 1$, g_i *may have different*

effects onto a component g_j, depending on level x_i. Hence, the arc connecting g_i to g_j may be a multi-arc encompassing different interactions. The multiplicity of the arc (g_i, g_j) (i.e. the number of its constitutive interactions), is denoted $m_{i,j}$ ($1 \leq m_{i,j} \leq \mathcal{M}ax_i$). Loops (even multi-loops) are allowed: an arc (g_i, g_i) denotes an autoregulation of g_i.
For each $g_j \in \mathcal{G}$, $Reg(j)$ denotes the set of its regulators: $g_i \in Reg(j)$ if and only if $(g_i, g_j) \in \Gamma$.

- *Θ is a labelling function, which associates a* threshold *to each element of Γ. More precisely, $\theta_{i,j,k}$ is associated to the k^{th} interaction between g_i and g_j (denoted $(g_i, g_j, \theta_{i,j,k}), k \in \{1, \ldots, m_{i,j}\}$), with $1 \leq \theta_{i,j,1} < \cdots < \theta_{i,j,m_{i,j}} \leq \mathcal{M}ax_i$. This interaction is* active*, when x_i, the level of its source g_i, lays between the threshold of this interaction and that of the next interaction: $\theta_{i,j,k} \leq x_i < \theta_{i,j,k+1}$ (by convention, $\theta_{i,j,m_{i,j}+1} = \mathcal{M}ax_i + 1$).*
- *$\mathcal{K} = (\mathcal{K}_1, \ldots, \mathcal{K}_N)$ defines the* logical rules *attached to the nodes specifying their behaviours: each $\mathcal{K}_i$ is a multi-valued logical function that gives the target value of g_i, depending on the levels of the regulators acting on g_i:*

$$\mathcal{K}_i : \left(\prod_{g_j \in \mathcal{G}} \mathcal{D}_j \right) \mapsto \{0, \ldots, \mathcal{M}ax_i\}.$$

The logical function $\mathcal{K}_i$ can be equivalently defined on the set $\prod_{g_j \in Reg(i)} \mathcal{D}_j$, giving the target value of g_i depending on the current levels of its regulators. Figure 1 illustrates this definition of a logical regulatory graph. In the following, when no confusion is possible, we will use i to denote g_i.



$\mathcal{G} = \{g_1, g_2, g_3, g_4\}$
$\mathcal{M}ax_1 = \mathcal{M}ax_2 = \mathcal{M}ax_4 = 1$
$\mathcal{M}ax_3 = 2$
$\mathcal{D}_1 = \mathcal{D}_2 = \mathcal{D}_4 = \{0, 1\}, \mathcal{D}_3 = \{0, 1, 2\}$
$Reg(2) = \{g_1, g_3\}$
$\theta_{3,2,1} = 2$

Fig. 1. Example of logical regulatory graph. Left: graphical representation of a LRG. Blunt arrows depict inhibitions while normal arrows depict activations (this is only a graphical convention, since the logical functions encode the regulatory effects). The rectangular node g_3 is ternary, whereas the others nodes are Boolean. The thresholds of all interactions are set to 1, except that of (g_3, g_2), which is set to 2. Right: illustration of the notations of Definition 1. Examples of logical functions $\mathcal{K}_i$ are displayed in Figure 2 for the same model.

2.2 State Transition Graphs

We represent the dynamical behaviour of a LRG in terms of a state transition graph, defined as follows.

Definition 2. *Given a LRG* $\mathcal{R} = (\mathcal{G}, \mathcal{M}ax, \Gamma, \Theta, \mathcal{K})$*, its associated full state transition graph (STG)* $\mathcal{E} = (\mathcal{S}, \mathcal{T})$ *is a directed graph, where:*

- $\mathcal{S} = \Pi_{i \in \mathcal{G}} \mathcal{D}_i$ *is the state space, a state of the system being a vector* $x = (x_i)_{i=1,\ldots,N}$*, with* $x_i \in \mathcal{D}_i$, $\forall i \in \mathcal{G}$,
- $\mathcal{T} \subset \mathcal{S}^2$ *is the set of transitions defined as follows:* $(x, y) \in \mathcal{T}$ *if and only if* $\exists i \in \mathcal{G}$ *such that:*
  $x_i \neq \mathcal{K}_i(x)$,
  $y = x + \Delta_i(x).e^i$*, where* $\Delta_i(x) = \frac{\mathcal{K}_i(x) - x_i}{|\mathcal{K}_i(x) - x_i|}$ *and* e^i *is the canonical vector in* $\mathcal{S}$ *(*$e^i_i = 1$ *and* $e^i_j = 0$, $\forall j \in \mathcal{G}, j \neq i$*).*

Here $\Delta_i(x)$ gives the sign of the update of i (increase or decrease). One can also consider a state transition graph related to an initial (set of) condition(s). It is then a subgraph of the full STG.

When analysing the behaviour of a LRG, we mainly focus on attractors, which represent asymptotic dynamical properties. Given a STG, attractors are its terminal strongly connected components, classified as:

- *stable states*: reduced to a unique terminal node,
- *cyclic attractors*: terminal elementary (oriented) cycles,
- *complex attractors*: other terminal strongly connected components (*i.e.* involving intertwined cycles).

Cyclic and complex attractors will be called *non-trivial* attractors.

In what follows, LRGs are assumed consistent, *i.e.* all interactions are effective and autoregulations functional, meaning that all interactions have a dynamical role and could be recovered from the logical functions $\mathcal{K}$ (see further explanations in the Appendix A).

3 Logical Regulatory Graph Reduction

This section presents the principles underlying the reduction of a regulatory graph and then defines the new model, called *reduced model*. In what follows, we consider a reduction consisting of the *removal* of a single regulatory component (making it implicit). The generalisation to a reduction encompassing a set of nodes is obtained by iterating the corresponding one-node reductions. However, the ordering of a sequence of one-node reductions may have an impact on the resulting reduced model (see Appendix B).

Here, we aim at defining a reduction method, which preserves, as much as possible, the dynamical properties of the original model. The underlying principle is already intuitively applied by modellers when they make regulatory nodes implicit in their networks.

The removal of a node r basically consists in connecting directly its regulators to its targets, which logical functions are thus revised. In the revised logical functions, the effect of r at a given value x_r is conveyed by the values of the regulators leading r to x_r. In other words, we consider the update of the removed component as a *fast process*, which is performed before anything else.

Following this principle, it is impossible to remove an autoregulated component since fixed values of its other regulators may not lead to a unique target value. Thus, the removal of an autoregulated component implies additional decisions, impeding the definition of a systematic procedure. In the following, we will require that autoregulated components should not be removed.

To properly implement an algorithm producing the reduced model, we need further notations to manipulate the logical functions. Given a regulatory graph $\mathcal{R} = (\mathcal{G}, \mathcal{M}ax, \Gamma, \Theta, \mathcal{K})$ and a node $i \in \mathcal{G}$, we denote:

- $x_i^{\{l\}}$ $(l \in \mathcal{D}_i)$ the Boolean variable with value 1 when $x_i = l$, 0 otherwise.
- x_i^S the Boolean variable that is true if $x_i \in S$, false otherwise. Hence x_i^S is defined by,
$$x_i^S \triangleq \bigvee_{l \in S} x_i^{\{l\}},\ S \subseteq \mathcal{D}_i.$$
Note that $x_i^{\emptyset}$ is always false and $x_i^{\mathcal{D}_i}$ always true.
- For all $v \in \mathcal{D}_i$, the logical function $\mathcal{K}_i^v$ that gives the conditions under which the target value of node i is v. This function is defined as follows:
$$\mathcal{K}_i^v = \bigvee_{n=1,\ldots p} \mathcal{C}_i^n, \tag{1}$$
where $\mathcal{C}_i^n$ are conjunctive clauses $\mathcal{C}_i^n = \bigwedge_{j \in Reg(i)} x_j^{S_{j,i,n}}$, where $S_{j,i,n} \subseteq \mathcal{D}_j$. Each clause $\mathcal{C}_i^n$ defines a situation (*i.e.* sets of combinations of incoming interactions acting upon i) for which the target value of i is v.

In Equation (1), each clause $\mathcal{C}_i^n$ defines a subset of $\mathcal{S}$, $D = \Pi_{j \in \mathcal{G}} S_{j,i,n}$ (with $S_{j,i,n} = \mathcal{D}_j$, $\forall j \notin Reg(i)$), such that for all $x \in D$, $\mathcal{K}_i(x) = v$. Hence, Equation (1) defines a set of cubes in the state space $\mathcal{S}$, where the target value of i is v.

Definition 3. *Given a LRG $\mathcal{R} = (\mathcal{G}, \mathcal{M}ax, \Gamma, \Theta, \mathcal{K})$, the reduced LRG $\mathcal{R}^r = (\mathcal{G}^r, \mathcal{M}ax^r, \Gamma^r, \Theta^r, \mathcal{K}^r)$ obtained by removing a non-autoregulated component $r \in \mathcal{G}$ is defined as follows:*

- $\mathcal{G}^r = \mathcal{G} \setminus \{r\}$.
- $\mathcal{M}ax^r : \mathcal{G}^r \to \mathbb{N}^*$, *s.t.* $\forall i \in \mathcal{G}^r\ \mathcal{M}ax^r(i) = \mathcal{M}ax_i$.
- *For all $i \in \mathcal{G}^r$, and for all $v \in \mathcal{D}_i$, the logical function $\mathcal{K}_i^{rv}$ is defined as follows. Consider $\mathcal{K}_i^v = \bigvee_{n=1,\ldots p} \mathcal{C}_i^n$, the disjunctive form of $\mathcal{K}_i^v$, as defined previously. For all $n = 1, \ldots p$ (i.e. for each clause $\mathcal{C}_i^n$), let define $\mathcal{F}_i^{rn}$ as:*
$$\mathcal{F}_i^{rn} = \left(\bigvee_{w \in S_{r,i,n}} \mathcal{K}_r^w \right) \wedge \left(\bigwedge_{j \in Reg(i) \setminus \{r\}} x_j^{S_{j,i,n}} \right)$$
Then $\mathcal{K}_i^{rv} = \bigvee_{n=1,\ldots p} \mathcal{F}_i^{rn}$.
- *Γ^r and Θ^r are deduced from $\mathcal{K}^r$; for all $i \in \mathcal{G}^r$, $j \in \mathcal{G}^r$,*
$$m_{i,j}^r = \sum_{v \in [1, \mathcal{M}ax_i]} \mathbb{1}_{i,j,v},$$

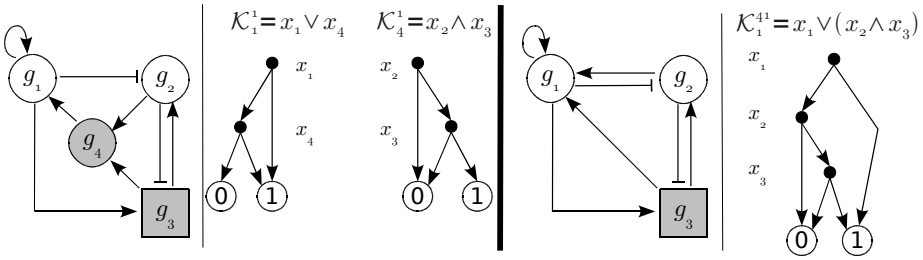


Fig. 2. Reduction in terms of MDDs. Left: the same LRG as in Figure 1, where g_4 (greyed-out) is selected for removal. Logical functions for g_1 and g_4 are shown on the right, along with their MDD representations. Right: the reduced LRG after removal of g_4, along with the resulting logical function for g_1. In the MDDs, internal nodes are labelled with the associated variable (x_i), whereas leaves represent the value of the logical functions. Children of internal nodes are ordered from left to right: the leftmost (resp. rightmost) child is the root of the sub-diagram corresponding to the case $x_i = 0$ (resp. $x_i = \mathcal{M}ax_i$).

where $\mathbb{1}_{i,j,v} = 1$ if it exists $x \in \mathcal{S}$ such that $x_i = v-1$ and $\mathcal{K}_j^r(x) \neq \mathcal{K}_j^r(x+e^i)$. Then $(i,j) \in \Gamma^r$ if $m_{i,j}^r > 0$ (and the multiplicity of (i,j) in Γ^r is given by $m_{i,j}^r$). Finally, the ordered set of values v such that $\mathbb{1}_{i,j,v} = 1$ defines the thresholds $\theta_{i,j,k}^r$ ($k = 1, \ldots, m_{i,j}^r$).

The logical function $\mathcal{K}_i^{rv}$ is deduced from the logical function $\mathcal{K}_i^v$ by replacing, in each clause, literals x_r^S by the formulae giving the conditions under which the target value of r is in S (remark that this definition may not give $\mathcal{K}_i^{rv}$ in a proper disjunctive form). Note that if $\mathcal{C}_i^n$ does not depend on r (*i.e.* $r \notin Reg(i)$) then $S_{r,i,n} = \mathcal{D}_r$ and $\mathcal{F}_i^{rn} = \mathcal{C}_i^n$ for all n, therefore $\mathcal{K}_i^{rv} = \mathcal{K}_i^v$.

The set of arcs verifies:

$$\Gamma^r \subseteq \{(i,j) \in \mathcal{G}^r \times \mathcal{G}^r, \text{ s.t. } (i,r),\ (r,j) \in \Gamma \text{ or } (i,j) \in \Gamma\}.$$

In practice, the construction of the new logical function is performed using Reduced Ordered Multivalued Decision Diagrams (ROMDDs or MDDs for short). Decision diagrams are rooted directed acyclic graphs, widely used to represent logical functions (see *e.g.* [7,8]). In these diagrams, internal nodes are labelled with *decision variables* and have one child per value, while leaves represent the values of the function. Decision variables are ordered: each internal node has a rank and the sub-diagrams rooted by the children of a node of rank i do not contain internal nodes of rank $j \leq i$. In [9] we used MDDs to represent the logical functions $\mathcal{K}_i$. In this context, decision variables are the levels of the components of the model. For the sake of simplicity, we consider that the ordering of the MDD variables is the same as that of the LRG components. Given the MDD representation of $\mathcal{K}_i$ and a state x, a unique path from the root of the MDD to one of its leaves is defined. Along this path, the child chosen for each non-terminal node is labelled with the value of the corresponding variable in state

x. The terminal node reached through this path gives the value of $\mathcal{K}_i(x)$. Each clause of $\mathcal{K}_i^v$ corresponds to a path leading to a leaf valued v.

To compute the MDD representing $\mathcal{K}_i^r$, we define the recursive algorithm given in Appendix C and illustrated in Figure 2.

4 Dynamics of the Reduced Model

In this section, the dynamical behaviour of a reduced LRG (as specified in Definition 3) is compared to that of the original LRG. In particular, we show that the reduction preserves existing attractors and does not add any spurious path.

Let $\mathcal{E} = (\mathcal{S}, \mathcal{T})$ be the full state transition graph of $\mathcal{R} = (\mathcal{G}, \mathcal{M}ax, \Gamma, \Theta, \mathcal{K})$ and $r \in \mathcal{G}$ a node not autoregulated. Let $\mathcal{E}^r = (\mathcal{S}^r, \mathcal{T}^r)$ be the full STG of $\mathcal{R}^r = (\mathcal{G}^r, \mathcal{M}ax^r, \Gamma^r, \Theta^r, \mathcal{K}^r)$, the LRG obtained after the removal of r from $\mathcal{G}$.

Consider the projection $\pi_r : \mathcal{S} \to \mathcal{S}^r$ such that, $\forall i \in \mathcal{G}^r$, $\forall x \in \mathcal{S}$, $(\pi_r(x))_i = x_i$, and the equivalence relation on $\mathcal{S}$: $\forall x, y \in \mathcal{S}$, $x \sim_r y$ iff $\pi_r(x) = \pi_r(y)$.

We denote $[x]_{\sim r}$ the equivalence class: $[x]_{\sim r} = \{y \in \mathcal{S} \text{ s.t. } y \sim_r x\}$. The class $[x]_{\sim r}$ contains all states of $\mathcal{S}$ that differ only by their r^{th} component, *i.e.* the $(\mathcal{M}ax_r + 1)$ states $\{x^i \in \mathcal{S},\ i = 0, \ldots, \mathcal{M}ax_r\}$, such that $x^i \sim_r x$ and $x_r^i = i$. Because r is not autoregulated, $\forall x^i \in [x]_{\sim r}$, $\mathcal{K}_r(x^i) = \mathcal{K}_r(x)$. This implies that:

- $(x^i, x^{i+1}) \in \mathcal{T}$, for all $i < \mathcal{K}_r(x)$,
- $(x^i, x^{i-1}) \in \mathcal{T}$, for all $i > \mathcal{K}_r(x)$,
- $(x^{\mathcal{K}_r(x)}, x^i) \notin \mathcal{T}$, for all i.

Hence, for all $x \in \mathcal{S}$, there exists a path in $\mathcal{S}$ from x to $x^{\mathcal{K}_r(x)}$, which is the representative state of $[x]_{\sim r}$.

Definition 4. *$x \in \mathcal{S}$ is the* representative state *of an equivalence class for $\sim_r$ iff $x_r = \mathcal{K}_r(x)$.*

We can then define the *retrieval* function $s_r : \mathcal{S}^r \to \mathcal{S}$ such that, $\forall z \in \mathcal{S}^r$,

$$(s_r(z))_i = z_i, \text{ for all } i \in \mathcal{G} \setminus \{r\},$$

$$(s_r(z))_r = \mathcal{K}_r(x), \text{ with } x \text{ such that } \pi_r(x) = z.$$

In other words, $s_r(z)$ is the representative state of the equivalence class projected on z (see Figure 3). Relying on this, we can introduce an alternative definition of the logical functions in the reduced LRG: $\forall i \in \mathcal{G}^r$, $\mathcal{K}_i^r : \mathcal{S}^r \mapsto \mathcal{D}_i$ is defined as $\mathcal{K}_i^r(z) = \mathcal{K}_i(s_r(z))$. Note that if $(r, i) \notin \Gamma$ (*i.e.* r is not a regulator of i), $\mathcal{K}_i^r(\pi_r(x)) = \mathcal{K}_i(x)$.

Remark 1. It follows from their definitions that functions π_r and s_r verify:

1. $\pi_r \circ s_r$ is the identity function.
2. For any $x \in \mathcal{S}$, $(s_r \circ \pi_r(x)) \sim_r x$.
3. If $x \in \mathcal{S}$ is a representative state, then, $s_r \circ \pi_r(x) = x$.
4. For any $z \in \mathcal{S}^r$, $\mathcal{K}^r(z) = \pi_r(\mathcal{K}(s_r(z)))$; indeed, $\forall x \in \mathcal{S}$, $\forall i \in \mathcal{G}^r$, $\mathcal{K}_i^r(\pi_r(x)) = \mathcal{K}_i(s_r \circ \pi_r(x))$.

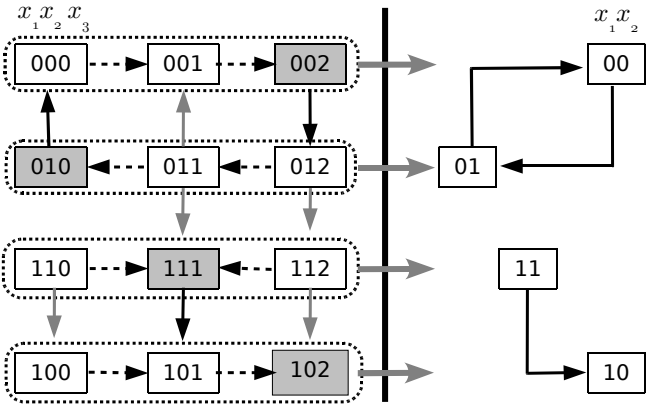


Fig. 3. Dynamical behaviour of the reduced model given in Figure 2, before and after removal of the ternary node g_3. Left: State transition graph (STG), partitioned into four equivalence classes for g_3. Each equivalence class contains 3 states; its representative state is greyed out and internal transitions are dashed. Right: STG of the reduced model, each state corresponding to an equivalence class of the original STG. After the reduction, the stable state 102 is projected on 10 and all transitions are preserved except the one from the second equivalence class to the third one. This results in the isolation of the non-terminal strongly connected component involving the first two equivalence classes of the original STG, hence generating the attractor $(01, 00)$.

The following lemma establishes the relationships between transitions in $\mathcal{E}$ and $\mathcal{E}^r$.

Lemma 1. *1. Let $z, z' \in \mathcal{S}^r$.*

$$(z, z') \in \mathcal{T}^r \implies \exists x \in \mathcal{S} \text{ s.t. } \pi_r(x) = z' \text{ and } (s_r(z), x) \in \mathcal{T}\,.$$

2. Let x, $y \in \mathcal{S}$. If x is a representative state, then

$$(x, y) \in \mathcal{T} \implies (\pi_r(x), \pi_r(y)) \in \mathcal{T}^r\,.$$

Proof. Recall that $\Delta_i(x) \triangleq \frac{\mathcal{K}_i(x) - x_i}{|\mathcal{K}_i(x) - x_i|}$. For $z \in \mathcal{S}^r$ s.t. $z_i \neq \mathcal{K}_i^r(z)$, we similarly denote:

$$\Delta_i^r(z) \triangleq \frac{\mathcal{K}_i^r(z) - z_i}{|\mathcal{K}_i^r(z) - z_i|} = \frac{\mathcal{K}_i(s_r(z)) - (s_r(z))_i}{|\mathcal{K}_i(s_r(z)) - (s_r(z))_i|} = \Delta_i(s_r(z))\,.$$

1. Consider $z, z' \in \mathcal{S}^r$ such that $(z, z') \in \mathcal{T}^r$. Then $\exists i \neq r$ s.t. $\mathcal{K}_i^r(z) \neq z_i$, and $z' = z + \Delta_i^r(z)\, e^i$. By definition, $\mathcal{K}_i^r(z) = \mathcal{K}_i(s_r(z)) \neq (s_r(z))_i = z_i$. This implies that $(s_r(z), x) \in \mathcal{T}$ with $x \in \mathcal{S}$ and $x = s_r(z) + \Delta_i(s_r(z))\, e^i$, and then $\pi_r(x) = z'$.
2. Consider $x, y \in \mathcal{S}$ such that $\mathcal{K}_r(x) = x_r$. The hypothesis $(x, y) \in \mathcal{T}$ implies that $\exists i \in \mathcal{G}, i \neq r$ s.t. $\mathcal{K}_i(x) \neq x_i$, and $y = x + \Delta_i(x)\, e^i$.

We have $\mathcal{K}_i^r(\pi_r(x)) = \mathcal{K}_i(x)$ (since x is a representative state), and $x_i = (\pi_r(x))_i$, since $i \neq r$. So, $\mathcal{K}_i^r(\pi_r(x)) \neq (\pi_r(x))_i$, and then $\exists z \in \mathcal{S}^r$ s.t. $(\pi_r(x), z) \in \mathcal{T}^r$, with

$$z = \pi_r(x) + \Delta_i^r(\pi_r(x))\, e^i = \pi_r(x) + \Delta_i(s_r \circ \pi_r(x))\, e^i = \pi_r(y)\,. \qquad \square$$

The first item of Lemma 1 states that any transition in $\mathcal{T}^r$ corresponds to at least one transition in $\mathcal{T}$. Clearly, the reverse is not true. The second item of the lemma gives a condition under which transitions are preserved from $\mathcal{T}$ to $\mathcal{T}^r$. Of course, it is important to know which transitions are lost through the reduction.

Definition 5. *The reduction preserves a transition* $(x, y) \in \mathcal{T}$ *if* $(\pi_r(x), \pi_r(y)) \in \mathcal{T}^r$, *or* $\pi_r(x) = \pi_r(y)$. *The reduction preserves a path* $(s_1, \ldots, s_n) \in \mathcal{E}$ *if all its transitions are preserved.*

In other words, a path $(s_1, \ldots, s_n)$ in $\mathcal{E}$ is preserved if the reduction preserves the transitions between equivalence classes, in the required order.

The following property characterises the transitions that are not preserved by the reduction.

Property 1. A transition $(x, y) \in \mathcal{T}$ is *not* preserved by the reduction if and only if the three following conditions are satisfied:

1. x is not a representative state,
2. $y \notin [x]_{\sim r}$ ($\Rightarrow\ \exists i \neq r$ s.t. $y_i \neq x_i$),
3. $\Delta_i(x) \neq \Delta_i(s_r \circ \pi_r(x))$.

The last condition means that there is no call for updating i in the same direction in state $s_r \circ \pi_r(x)$.

Proof. Consider a transition $(x, y) \in \mathcal{T}$, which satisfies the three conditions. Suppose that (x, y) is preserved by the reduction, then $(\pi_r(x), \pi_r(y)) \in \mathcal{T}^r$ (the case $\pi_r(x) = \pi_r(y)$ is not possible because of the second condition). This means that there exists $j \neq r$ s.t. $(\pi_r(x))_j \neq (\pi_r(y))_j$, and $(\pi_r(x))_k = (\pi_r(y))_k$ for any $k \neq j$. With Condition 2 and by definition of π_r, we deduce that $j = i$. Moreover, we know that:

$$\pi_r(y) = \pi_r(x) + \Delta_i^r(\pi_r(x))\, e^i = \pi_r(x) + \Delta_i(s_r \circ \pi_r(x))\, e^i\,.$$

Finally, $y = x + \Delta_i(x)\, e^i$, and, as $y_i = (\pi_r(y))_i$, we have $\Delta_i(x) = \Delta_i(s_r \circ \pi_r(x))$. This contradicts Condition 3. Hence, (x, y) is not preserved by the reduction.

Conversely, let $(x, y) \in \mathcal{T}$ be a transition not preserved by the reduction.

- Condition 1 is satisfied by the second item of Lemma 1.
- Condition 2 is satisfied because $y \in [x]_{\sim r} \ \Rightarrow \pi_r(x) = \pi_r(y) \ \Rightarrow \ (x, y)$ preserved, hence a contradiction.
- We know that $y = x + \Delta_i(x)\, e^i$. As $\mathcal{K}^r(\pi_r(x)) = \pi_r(\mathcal{K}(s_r \circ \pi_r(x)))$ (cf. Remark 1),

$$\begin{aligned} \mathcal{K}_i^r(\pi_r(x)) &= (\pi_r(\mathcal{K}(s_r \circ \pi_r(x))))_i = \mathcal{K}_i(s_r \circ \pi_r(x)) \\ &\neq x_i = (\pi_r(x))_i\,. \end{aligned}$$

Hence, there exists $z \in \mathcal{S}^r$ s.t. $(\pi_r(x), z) \in \mathcal{T}^r$ with

$$\begin{aligned} z_i &= \pi_r(x) + \Delta_i^r(\pi_r(x))\, e^i \\ &= \pi_r(x) + \Delta_i(s_r \circ \pi_r(x))\, e^i = \pi_r(x) + \Delta_i(x)\, e^i\,. \end{aligned}$$

Consequently, $\pi_r(y) = z$ and (x, y) is preserved, hence a contradiction. □

Given C, a set of states in $\mathcal{S}$, we denote $\pi_r(C) \triangleq \{\pi_r(x),\ x \in C\}$. Given C', a set of states in $\mathcal{S}^r$, we denote $s_r(C') \triangleq \{s_r(z),\, z \in C'\}$. Note that $\pi_r(C)$ may contain less elements than C, and that $s_r(C')$ contains only representative states. The following results relate attractors in $\mathcal{E}$ and $\mathcal{E}^r$. Proofs are provided in Appendix D and E.

Theorem 1. *Consider a LRG $\mathcal{R} = (\mathcal{G}, \mathcal{M}ax, \Gamma, \Theta, \mathcal{K})$ and $\mathcal{R}^r$ the reduced LRG. Let $\mathcal{E}$ (resp. $\mathcal{E}^r$) be the full STG of $\mathcal{R}$ (resp. of $\mathcal{R}^r$), then:*

1. *Stable states in $\mathcal{E}$ and $\mathcal{E}^r$ verify:*
   - *x stable state in $\mathcal{E} \implies \pi_r(x)$ stable state in $\mathcal{E}^r$. Furthermore no other stable state is projected on $\pi_r(x)$,*
   - *z stable state in $\mathcal{E}^r \implies s_r(z)$ stable state in $\mathcal{E}$.*

   Hence, the number of stable states is conserved by the reduction.
2. *If $(s_1, \ldots s_n)$ is a cyclic attractor in $\mathcal{E}$, then $(\pi_r(s_1), \ldots \pi_r(s_n))$ is a cyclic attractor in $\mathcal{E}^r$.*
3. *If C is a complex attractor in $\mathcal{E}$, $z \in \pi_r(C)$ and $(z, z') \in \mathcal{T}^r$, then $z' \in \pi_r(C)$. As a consequence, $\pi_r(C)$ contains at least one non-trivial attractor in $\mathcal{E}^r$.*

Theorem 1 characterises the dynamical properties conserved by the reduction. Going further, it is possible to identify the situations leading to the generation of additional non-trivial attractors. A non-trivial attractor in the reduced STG corresponds to a (part of a) strongly connected component of the original STG. This SCC is itself a non-trivial attractor or involves outgoing transitions all in conflict with transitions concerning the removed component. In other words, we can fully characterise the set of states in the original STG giving rise to a non-trivial attractor in the reduced dynamics. Interestingly, this set corresponds to transient oscillatory behaviour from which the system cannot escape provided that updates of the removed component are always faster than other concurrent changes. This is formalised by Theorem 2 in Appendix E.

5 Application: Segment Polarity

We demonstrate the power and flexibility of our reduction method through its application to the segment-polarity network, which plays a key role in the segmentation of the fly embryo. This system has been thoroughly analysed by developmental geneticists and has been already modelled using continuous [10,11,12] and logical approaches [13,14,15]. However, all these studies involved important simplifications of the network, particularly so as a proper modelling of

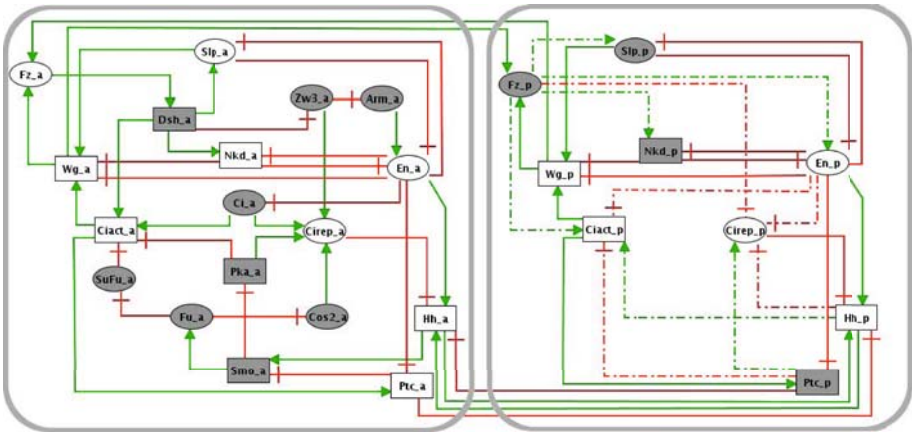


Fig. 4. Logical model of the segment polarity network for two cells, based on [15]. Ellipsoid and rectangular nodes denote Boolean and ternary components, respectively. The two cellular networks have been properly connected to take into account Wg and Hh diffusion, as well as Hh sequestration by Ptc, as in [16]. The anterior cell contains the extended version of the model, where greyed-out components will be removed, leading to the model on the right. Dashed arrows denote indirect interactions resulting from this reduction. Greyed-out components in the posterior cell are candidates for further reduction.

its behaviour requires the chaining of several identical networks to account for inter-cellular interactions through Wingless (Wg) and Hedgehog (Hh) signalling. Describing the most complete model to date, [15] had to discard various components known to play important roles in Wg and Hh signalling to keep dynamical simulations and analyses computationally tractable for up to six cells. Here, we propose a logical model based on their full description of the segment polarity network. The resulting regulatory graph encompasses 18 components and 31 regulatory interactions (left part of Figure 4).

In order to model the intercellular interactions involved in the formation of segment boundaries, we have to connect neighbouring cells (along the anterior-posterior axis) through Wg and Hh signalling. Wg is known to bind its receptor, Frizzled (Fz), only at very short range, amounting here to neighbouring cells. This can be represented by positive arcs linking each Wg node to Fz nodes of neighbouring cells. In contrast, Hh is able to reach more distant cells, but can be sequestered by its receptor Patched (Ptc). Similar interactions have been modelled in [16] in terms of positive arcs between Hh nodes in neighbouring cells (diffusion) and negative arcs from Ptc onto the Hh node of neighbouring cells (sequestering). Figure 4 illustrates the intercellular network obtained after coupling two cells and reducing one of the cellular sub-networks down to nine components.

The reduction method described above can be advantageously applied to ease the identification of all attractors of such intercellular models (Sánchez *et al.*

Table 1. Dynamical characteristics of different reduced models derived from that of Figure 4 (involving 2 x 9 nodes after applying the same reduction to both cells). The number of reachable states decreases drastically with the number of considered nodes. Note that the three stable states remain reachable for all reductions listed, but the last one (removal of Slp).

LRG size	Removed components	Number of reached states	Reached stable states
2x9	–	$> 10^6$	TT, WE, EW
2x7	Fz,Ptc	12476	TT, WE, EW
2x6	Fz,Ptc,Nkd	1625	TT, WE, EW
2x8	Slp	11350	TT, WE

considered six cells). The modeller can select the sets of nodes to discard from the network, depending on biological considerations (*e.g.* different time scales, specific mutations, etc.). In a first step, it is reasonable to conserve transcription factors and components involved in intercellular communications: Wg, Hh and their receptors (Fz and Ptc). However, since the transcription factor Cubitus interruptus is represented by three nodes here (full length immature Ci protein, activator Ci-act and repressor Ci-rep forms), we choose to retain only the two nodes corresponding to active regulatory forms. These choices correspond to the removal of the greyed-out components in the left part of Figure 4.

The reduced model involves half of the nodes of the original one, which amounts to a much higher reduction of the number of possible states, as this grows exponentially with the number of regulatory nodes. The resulting regulatory graph (Figure 4, right) remains easy to grasp as it reasonably unfolds most intra-cellular and inter-cellular regulatory pathways. As we shall see, this logical model can be further reduced to facilitate analyses encompassing more cells.

For proper logical rules (cf. [15] and supplementary material), one can check that the detailed and the reduced two-cells models have exactly three stables states (as predicted by Theorem 1). These multi-cellular stable states combine three types of cellular states: a Wg expressing state (denoted W), an En expressing state (E), and a *trivial* state (T) expressing neither Wg, nor En. The three stable states for the two connected cells correspond to TT, WE and EW cell combinations reported by Sánchez *et al.* All three stable states are reachable from biologically relevant initial conditions (significant amounts of Wg and Slp in the anterior cell, significant amount of En in the posterior cell), provided as an outcome of the activity of the pair-rule system, cf. [17,15]. However, the size of the corresponding state transition graph still impedes detailed dynamical analyses (see Table 1).

As shown in Table 1, the removal of Fz, Ptc and Nkd drastically reduces the number of reached states without changing the reachability of the three stable states from the considered initial state. However, the sole removal of Slp impedes the reachability of the stable state with inverted Wg and En expressing cells. It also suppresses the functionality contexts of all negative circuits [9,15], implying that the state transition graph does not contain any cyclic attractor. Indeed, after further reduction to three nodes per cell (Wg, En and Hh), we were able

to check the absence of non-trivial attractors in the full STG. As the reduction cannot delete existing non-trivial attractors (see Theorem 1), this implies that all attractors of the original model are stable states.

6 Conclusions and Prospects

We have defined a reduction method that can be applied to multi-valued logical models while preserving important dynamical properties. In particular, all attractors of the original dynamics have a counterpart in the dynamics of the reduced model. Furthermore, trajectories in the reduced model can be formally related to trajectories in the original one. This enables to infer the existence of paths in the dynamics of a detailed model whenever it is possible to show (by simulation and graph analysis) that paths exist between the corresponding states in a reduced version of the model. However, the reverse is not true. Indeed, a reduction can lead to the loss of reachability properties. Whenever several asynchronous component updates are possible at a given state, the elimination of one of the updated components amounts to consider it as "faster" than the concurrent ones, leading to the possible exclusion of some transitions in the reduced STG. Such reductions relate to the delineation of specific priority class configurations [18].

One particular feature of the reduction method defined here is that the removal of (functional) autoregulated components is forbidden. This rule is related to previous work on the dynamical roles of the regulatory circuits. Indeed, it has been recently proven in the discrete framework that positive regulatory circuits are necessary to generated multiple attractors, whereas negative circuits are necessary to generate cyclic attractors (cf. [19] and references therein). At least in the discrete framework, these properties depend only on the sign of the regulatory circuit, *i.e.* on the product of the sign of the involved interactions and not on their number. From a qualitative dynamical point of view, it is thus possible to reduce the number of components of a circuit down to a single autoregulated component, while keeping the corresponding property, as long as we conserve the sign of the circuit (along with some functionality constraints).

Our formal presentation of the reduction method mainly focuses on the removal of a single component. However, iterating this process enables the removal of several components. This raises the question of the impact of the order in which reductions are performed. As shown in Appendix B, the removal of a component may be possible only after the prior removal of others. If we aim at removing as many components as possible, the ordering of removals may thus be crucial. Further work is needed to properly define optimal or maximal reductions for the general case. When the removal of a set of components is possible in several orders, we suspect that the dynamics of the resulting model does not depend on the order (work in progress).

The worst case complexity of the algorithm for the reduction of a node r that regulates k targets is in $O(m^d)$, where m is the highest number of levels of the involved components and d is the depth of the MDDs representing the revised logical functions associated to the target nodes. In most cases, $m \leq 3$ and $d \leq 5$.

Applying our reduction method to a detailed model of the segment-polarity network, we were able to show the absence of non-trivial attractors in a state transition graph too large to be stored. As indicated for this application, the reduction method offers a great flexibility to the modeller. Biological arguments (*e.g.* information on relative reaction speeds) can be used to select sets of nodes for consistent model reduction. In the course of the dynamical analysis of complex networks (*e.g.* multicellular networks), further reduction can be performed to identify all attractors and check their reachability from specific initial states.

To ease the maintenance of a detailed model along with its reduced versions, the *GINsim* implementation enables the user to define and record various reductions for the same reference model. In order to handle still larger and more complex networks, such reduction could be combined with algorithmic methods enabling the analysis of large state transition graphs ([20] and references therein), or yet with model checking techniques ([21] and references therein).

Supplementary Materials. *GINsim* can be downloaded from `http://gin.univ-mrs.fr/GINsim`. The Appendix, and the models, are available at the following URL: `http://gin.univ-mrs.fr/GINsim/publications/naldi2009.html`.

Acknowledgments. A.N. has been supported by PhD grant from the French Ministry of Research and Technology. C.C. acknowledges the support provided by the Calouste Gulbenkian Foundation. This work was further supported by research grants from the French National Agency (project ANR-08-SYSC-003), from EU FP7 (APO-SYS Project), and by the Belgian Science Policy Office (IAP BioMaGNet).

References

1. Saez-Rodriguez, J., Simeoni, L., Lindquist, J., Hemenway, R., Bommhardt, U., Arndt, B., Haus, U., Weismantel, R., Gilles, E., Klamt, S., Schraven, B.: A logical model provides insights into t cell receptor signaling. PLoS Comput. Biol. 3(8), e163 (2007)
2. Franke, R., Müller, M., Wundrack, N., Gilles, E.D., Klamt, S., Kähne, T., Naumann, M.: Host-pathogen systems biology: logical modelling of hepatocyte growth factor and helicobacter pylori induced c-met signal transduction. BMC Syst. Biol. 2, 4 (2008)
3. Chaouiya, C., Remy, E., Mossé, B., Thieffry, D.: Qualitative analysis of regulatory graphs: a computational tool based on a discrete formal framework. LNCIS, vol. 294, pp. 119–126 (2003)
4. Naldi, A., Berenguier, D., Fauré, A., Lopez, F., Thieffry, D., Chaouiya, C.: Logical modelling of regulatory networks with GINsim 2.3. BioSystems (in press)
5. Thomas, R.: Regulatory networks seen as asynchronous automata: A logical description. J. Theor. Biol. 153, 1–23 (1991)
6. Thomas, R., Thieffry, D., Kaufman, M.: Dynamical behaviour of biological regulatory networks–i. biological role of feedback loops and practical use of the concept of the loop-characteristic state. Bull. Math. Biol. 57(2), 247–276 (1995)

7. Bryant, R.E.: Graph-based algorithms for boolean function manipulation. IEEE Trans. Comput. 35, 677–691 (1986)
8. Kam, T., Villa, T., Brayton, R.K., Sangiovanni-Vincentelli, A.L.: Multi-valued decision diagrams: Theory and applications. Int. J. Multi. Logic 4, 9–12 (1998)
9. Naldi, A., Thieffry, D., Chaouiya, C.: Decision diagrams for the representation and analysis of logical models of genetic networks. In: Calder, M., Gilmore, S. (eds.) CMSB 2007. LNCS (LNBI), vol. 4695, pp. 233–247. Springer, Heidelberg (2007)
10. Meinhardt, H.: Hierarchical inductions of cell states: a model for segmentation in drosophila. J. Cell Sci. Suppl. 4, 357–381 (1986)
11. von Dassow, G., Meir, E., Munro, E.M., Odell, G.M.: The segment polarity network is a robust developmental module. Nature 406(6792), 188–192 (2000)
12. Ingolia, N.T.: Topology and robustness in the drosophila segment polarity network. PLoS Biol. 2(6), e123 (2004)
13. Albert, R., Othmer, H.G.: The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in drosophila melanogaster. J. Theor. Biol. 223(1), 1–18 (2003)
14. Chaves, M., Albert, R., Sontag, E.D.: Robustness and fragility of boolean models for genetic regulatory networks. J. Theor. Biol. 235(3), 431–449 (2005)
15. Sánchez, L., Chaouiya, C., Thieffry, D.: Segmenting the fly embryo: logical analysis of the role of the segment polarity cross-regulatory module. Int. J. Dev. Biol. 52(8), 1059–1075 (2008)
16. González, A., Chaouiya, C., Thieffry, D.: Logical modelling of the role of the hh pathway in the patterning of the drosophila wing disc. Bioinformatics 24(16), i234–i240 (2008)
17. Sánchez, L., Thieffry, D.: Segmenting the fly embryo: a logical analysis of the pair-rule cross-regulatory module. J. Theor. Biol. 224(4), 517–537 (2003)
18. Fauré, A., Naldi, A., Chaouiya, C., Thieffry, D.: Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. Bioinformatics 22(14), e124–e131 (2006)
19. Remy, E., Ruet, P.: From minimal signed circuits to the dynamics of boolean regulatory networks. Bioinformatics 24(16), i220–i226 (2008)
20. Garg, A., Di Cara, A., Xenarios, I., Mendoza, L., De Micheli, G.: Synchronous versus asynchronous modeling of gene regulatory networks. Bioinformatics 24(17), 1917–1925 (2008)
21. Monteiro, P.T., Ropers, D., Mateescu, R., Freitas, A.T., de Jong, H.: Temporal logic patterns for querying dynamic models of cellular interaction networks. Bioinformatics 24(16), i227–i233 (2008)