Methods of robustness analysis for Boolean models of gene control networks *

Madalena Chaves, Eduardo D. Sontag and Réka Albert†

Abstract

As a discrete approach to genetic regulatory networks, Boolean models provide an essential qualitative description of the structure of interactions among genes and proteins. Boolean models generally assume only two possible states (expressed or not expressed) for each gene or protein in the network as well as a high level of synchronization among the various regulatory processes.

In this paper, we discuss and compare two possible methods of adapting qualitative models to incorporate the continuous-time character of regulatory networks. The first method consists of introducing asynchronous updates in the Boolean model. In the second method, we adopt the approach introduced by L. Glass to obtain a set of piecewise linear differential equations which continuously describe the states of each gene or protein in the network.

We apply both methods to a particular example: a Boolean model of the segment polarity gene network of *Drosophila melanogaster*. We analyze the dynamics of the model, and provide a theoretical characterization of the model's gene pattern prediction as a function of the timescales of the various processes.

1 Introduction

Genes and gene products interact on several levels. At the genomic level, transcription factors can activate or inhibit the transcription of genes to give mRNAs. Since these transcription factors are themselves products of genes, the ultimate effect is that genes regulate each other's expression as part of gene regulatory networks. Similarly, proteins can participate in diverse post-translational interactions that lead to modified protein functions or to formation of protein complexes that have new roles; the totality of these processes is called a protein-protein interaction network. In many cases different levels of interactions are integrated - for example, when the presence of an external signal triggers a cascade of interactions that involves biochemical reactions, protein-protein interactions and transcriptional regulation.

During the last decade, genomics, transcriptomics and proteomics have produced an incredible quantity of molecular interaction data, contributing to genome-scale maps of protein interaction networks [1, 2, 3] and transcriptional regulatory networks [4]. Network analysis of these maps revealed intriguing topological similarities [5]. At the same time, it has been increasingly realized that cellular interaction maps only represent a network of possibilities, and not all edges are present and active *in vivo* in a given condition or in a given cellular location [2, 6]. Therefore, only an integration of (time-dependent) interaction and activity information will be able to give the correct dynamical picture of a cellular network.

*M. Chaves is with the Institute for Systems Theory and Automatic Control, University of Stuttgart, Pfaffenwaldring 9, 70550 Stuttgart, Germany chaves@ist.uni-stuttgart.de. E.D. Sontag is with the Department of Mathematics, Rutgers University, Piscataway, NJ 08854 USA sontag@math.rutgers.edu. R. Albert is with the Department of Physics and Huck Institutes for the Life Sciences, Pennsylvania State University, University Park, PA 16802 USA ralbert@phys.psu.edu.

†Corresponding author.

For many biological networks, and in particular genetic control or regulatory networks, detailed information on the kinetic rates of protein-protein or protein-DNA interactions is rarely available. However, for many biological systems, evidence shows that regulatory relationships can be sigmoidal and be well approximated by step functions. In this case, Boolean models, where every variable has only two states (ON/OFF), and the dynamics is given by a set of logical rules, are frequently appropriate descriptions of the network of interactions among genes and proteins. Examples include models of genetic networks in the fruit fly *Drosophila melanogaster* [7, 8] and the flowering plant *Arabidopsis thaliana* [9, 10].

While Boolean models introduce biologically unrealistic time constraints (typically, such models use synchronous updates, which inherently assume that the various biological processes have the same duration), they still provide significant qualitative information on the underlying structure of the network. On the other hand, continuous models certainly have a more realistic time description of a biological system. But, in the absence of information on the kinetic rates, continuous models include many unknown parameters, and analysis of the system involves exploring the (often large) state space of parameters. An important (continuous) model for *Drosophila melanogaster* segment polarity genes was first developed in [11], where a thorough investigation of the parameter space showed that the system is very robust with respect to variations in the kinetic constants. Both this continuous model and the discrete model [8] agree in their overall conclusions regarding the robustness of the segment polarity gene network.

In this paper, we propose two new approaches to the analysis of Boolean models, which combine discrete logical rules and structure with more realistic assumptions regarding the relative timescales of the genetic processes. The first method introduces asynchronous updates in the Boolean model; since update times are randomly chosen, the model is now stochastic (see also [12]). The second method associates to the discrete variables a set of continuous variables, whose dynamics is given by a piecewise linear system of differential equations, thus introducing a simple "hybrid" model in the manner first suggested by L. Glass and collaborators in [13, 14, 15]. These methods allow us to naturally probe the system with respect to perturbations in the time dynamics to analyze its performance. Both methods uncover the robustness of the segment polarity gene model [8], and its ability to correctly predict the final gene expression pattern.

Section 2 summarizes the Boolean model [8], to be analyzed with our methods. The asynchronous updates and the Glass-type methods are described and applied in Sections 3 and 4, respectively. In Section 5, we discuss the effects of perturbations in initial gene expression, and the development of mutant phenotypes. Finally, in Section 6 we show that under a timescale separation between posttranslational and translational processes in the model [8], we are able to analytically and exactly compute how frequently the model will predict the correct gene pattern.

2 The segment polarity gene network in *Drosophila*

We will apply our analysis to a Boolean model of the interactions among the *Drosophila melanogaster* segment polarity genes. This gene network represents the last step in the hierarchical cascade of gene families initiating the segmented body of the fruit fly. While genes in preceding stages of development act transiently, the segment polarity genes are expressed throughout the life of the fly. The best characterized segment polarity genes include *engrailed* (*en*), *wingless* (*wg*), *hedgehog* (*hh*), *patched* (*ptc*), *cubitus interruptus* (*ci*) and *sloppy paired* (slp), coding for the corresponding proteins, which we will represent by capital letters (*EN*, *WG*, *HH*, *PTC*, *CI* and *SLP*). Two additional proteins, resulting from transformations of the protein *CI*, also play important roles: *CI* may be converted into a transcriptional activator, *CIA*, or may be cleaved to form a transcriptional repressor *CIR*.

The expression pattern of the *Drosophila* segment polarity genes (see Table 2) is maintained almost unmodified for three hours, during which time the embryo is divided into 14 parasegments. Each of these

Table 1: Regulatory functions governing the states of segment polarity gene products in the model. Each node is labeled by its biochemical symbol and subscripts signify cell number.

Node	Boolean updating function (synchronous algorithm)
SLP_i	$SLP_i(k+1) = \begin{cases} 0 & \text{if} \quad i \in \{1,2\} \\ 1 & \text{if} \quad i \in \{3,4\} \end{cases}$
wg_i	$wg_i(k+1) = (CIA_i(k)$ and $SLP_i(k)$ and not $CIR_i(k))$ or $[wg_i(k)$ and $(CIA_i(k)$ or $SLP_i(k))$ and not $CIR_i(k)]$
WG_i	$WG_i(k+1) = wg_i(k)$
en_i	$en_i(k+1) = (WG_{i-1}(k)$ or $WG_{i+1}(k))$ and not $SLP_i(k)$
EN_i	$EN_i(k+1) = en_i(k)$
hh_i	$hh_i(k+1) = EN_i(k)$ and not $CIR_i(k)$
HH_i	$HH_i(k+1) = hh_i(k)$
ptc_i	$ptc_i(k+1) = CIA_i(k)$ and not $EN_i(k)$ and not $CIR_i(k)$
PTC_i	$PTC_i(k+1) = ptc_i(k)$ or $(PTC_i(k)$ and not $HH_{i-1}(k)$ and not $HH_{i+1}(k))$
ci_i	$ci_i(k+1) =$ not $EN_i(k)$
CI_i	$CI_i(k+1) = ci_i(k)$
CIA_i	$CIA_i(k+1) = CI_i(k)$ and [not $PTC_i(k)$ or $HH_{i-1}(k)$ or $HH_{i+1}(k)$ or $hh_{i-1}(k)$ or $hh_{i+1}(k)$]
CIR_i	$CIR_i(k+1) = CI_i(k)$ and $PTC_i(k)$ and not $HH_{i-1}(k)$ and not $HH_{i+1}(k)$ and not $hh_{i-1}(k)$ and not $hh_{i+1}(k)$

parasegments is composed of about 4 cells, delimited by furrows positioned between the *wg* and *en* -expressing cells [16].

The Boolean model that we will study was introduced and developed by one of us in [8]. (Further robustness analysis was also developed in [12].) In this model, a parasegment of four cells is considered: the variables are the expression levels of the segment polarity genes and proteins (listed above) in each of the four cells (the total number of nodes in the network is thus $4 \times 13 = 52$.). The expression level of each gene or protein is assumed to be either 0 (OFF) or 1 (ON). The model successfully describes the transition from the initial expression pattern (1) to a final pattern two or three developmental stages later, when the embryo has been clearly divided into parasegments (see first entry of Table 2). As discussed in [8], the evolution of these gene expression patterns is well described by a set of logical rules, which are depicted in Table 1.

We adopt the notation "wg_1^k" or "$wg_1(k)$" to represent the state of *wingless* mRNA in the first cell of the parasegment at time k. Similar notations apply for other mRNAs and proteins. Periodic boundary conditions are assumed, meaning that: $node_{4+1} = node_1$ and $node_{1-1} = node_4$. The wild type initial pattern corresponds to:

$$wg_4^0 = 1,\ en_1^0 = 1,\ hh_1^0 = 1,\ ptc_{2,3,4}^0 = 1,\ ci_{2,3,4}^0 = 1, \tag{1}$$

with the remaining nodes zero.

2.1 Steady states of the Boolean model

A complete analysis of the steady states is found in [8]. Table 2 summarizes these results, indicating the expressed nodes in each of the six steady-states. We note that three of the four main steady states agree perfectly with experimentally observed states corresponding to wild type, to *ptc* knockout mutant (broad striped) and to *en*, *wg* or *hh* knockout mutant (non-segmented) embryonic patterns ([17, 18]; see [8] for more references).

Table 2: Complete characterization of the model's steady states.

Steady state	Expressed nodes
wild type	wg_4, WG_4, en_1, EN_1, hh_1, HH_1, $ptc_{2,4}$, $PTC_{2,3,4}$, $ci_{2,3,4}$, $CI_{2,3,4}$, $CIA_{2,4}$, CIR_3
broad stripes	$wg_{3,4}$, $WG_{3,4}$, $en_{1,2}$, $EN_{1,2}$, $hh_{1,2}$, $HH_{1,2}$, $ptc_{3,4}$, $PTC_{3,4}$, $ci_{3,4}$, $CI_{3,4}$, $CIA_{3,4}$
no segmentation	$ci_{1,2,3,4}$, $CI_{1,2,3,4}$, $PTC_{1,2,3,4}$, $CIR_{1,2,3,4}$
wild type variant	wg_4, WG_4, en_1, EN_1, hh_1, HH_1, $ptc_{2,4}$, $PTC_{1,2,3,4}$, $ci_{2,3,4}$, $CI_{2,3,4}$, $CIA_{2,4}$, CIR_3
ectopic	wg_3, WG_3, en_2, EN_2, hh_2, HH_2, $ptc_{1,3}$, $PTC_{1,3,4}$, $ci_{1,3,4}$, $CI_{1,3,4}$, $CIA_{1,3}$, CIR_4
ectopic variant	wg_3, WG_3, en_2, EN_2, hh_2, HH_2, $ptc_{1,3}$, $PTC_{1,2,3,4}$, $ci_{1,3,4}$, $CI_{1,3,4}$, $CIA_{1,3}$, CIR_4

2.2 The regulatory function of *sloppy paired* gene

The rule for protein *SLP* summarizes in a simple way experimental observations on the regulatory activity of *sloppy paired* gene in the segment polarity network. A more detailed rule for *SLP* can in fact be created, describing the *sloppy paired* expression pattern to incorporate recent evidence of engrailed protein (EN) inhibiting *slp* transcription [19]. However, inhibition by EN accounts only partially for the experimentally observed restriction of *slp* to the posterior half of the parasegment. Thus we need to invoke an additional regulatory effect, which we denote by *RX*. *RX* probably represents a combination of regulation by the pair-rule genes responsible for the establishment of *slp*, namely *runt*, *opa* and Factor X [20] and of *slp* autoregulation.

Therefore, *SLP* expression in Table 1 can be replaced by the following set of equations:

$$\begin{aligned} RX_i(k+1) &= \begin{cases} 0, & \text{if} \quad i \in \{1,2\} \\ 1, & \text{if} \quad i \in \{3,4\} \end{cases} \\ slp_i(k+1) &= RX_i(k) \text{ and not} EN_i(k) \\ SLP_i(k+1) &= slp_i(k)\ . \end{aligned} \tag{2}$$

The initial conditions would then be:

$$slp^0_{3,4} = 1,\ SLP^0_{3,4} = 1. \tag{3}$$

This generalization of the segment polarity network model also introduces other steady states, such as a two-stripe *en* pattern characterized by $slp_4 = SLP_4 = 1$, $wg_4 = WG_4 = 1$, $en_{1,3} = EN_{1,3} = 1$, $hh_{1,3} = HH_{1,3} = 1$, $ptc_{2,4} = PTC_{2,4} = 1$, and $ci_{2,4} = CI_{2,4} = CIA_{2,4} = 1$ (this pattern was also found in [8], as a result of *slp* mutation). On the other hand, it is not difficult to see that, starting from conditions (1) and (3), any of the "new" steady states are not reachable, since these initial conditions imply that neither *slp* nor *SLP* can change their expression at any time. Indeed we have the following result:

Lemma 2.1 Consider the extended model in Table 1 together with (2). Assume that initial conditions are given by (1) and (3). Then $slp_i(t) = slp_i^0$ and $SLP_i(t) = SLP_i^0$ for all times .

To prove this, note first that $slp_{1,2}(t) = SLP_{1,2}(t) \equiv 0$ follows from $RX_{1,2}(t) = 0$ for all t. Next, observe that:

$$slp_4(t_1) = 0 \Rightarrow EN_4(t_2) = 1 \Rightarrow en_4(t_3) = 1 \Rightarrow SLP_4(t_4) = 0 \Rightarrow slp_4(t_5) = 0,$$

where $t_1 \geq t_2 \geq t_3 \geq t_4 \geq t_5 \geq 0$. Thus, in order for slp_4 to become zero at any time, it had to be so at some previous instant ($t_5 \leq t_1$). If $slp_4(0) = 1$, and if $T > 0$ is the first instant such that $slp_4(T) = 0$ then we have a contradiction. Hence $slp_4(t) = 1$ for all times. (Note that this "proof" is independent of the order in which the nodes are updated.) Similar arguments show that $slp_3(t) = slp_3(0) = 1$ and $SLP_{3,4}(t) = 1$, for all t.

For this reason, our present analysis is focused on the simpler biologically relevant model of Table 1.

3 Asynchronous algorithms

In general, for a network of N gene products (denoted x_1, ..., x_N), the dynamics of a Boolean model is typically studied by simultaneously updating the state of all the nodes in the network, according to

$$X_i^{k+1} = F_i(X_1^k, X_2^k, \ldots, X_N^k), \quad i = 1, \ldots, N$$

where F_i is the regulating function for mRNA or protein X_i. An underlying hypothesis is the existence of perfect synchronization among the various regulatory processes. However, it is well known that the timescales of transcription, translation, and degradation processes can vary widely from gene to gene and can be anywhere from minutes to hours.

In analogy with task coordination and data communication procedures in the context of parallel computation systems [21], we have previously developed several methods that introduce different timescales for the different regulatory processes within the network [12]. These include algorithms that randomly choose the order in which the nodes are updated (this random order algorithm is summarized below in 3.1, and in Section 6), and a totally asynchronous algorithm, where the next updating times for each node are randomly chosen at each instant.

We now introduce a more intuitive asynchronous algorithm, where each node is updated according to its own specific time unit. The time units for the nodes are randomly chosen from a uniform distribution in an interval $[1-\varepsilon, 1+\varepsilon]$, where $\varepsilon \in (0, 1)$. The updating times of i-th node are then pre-specified as:

$$T_i^1, T_i^2, \ldots, T_i^k, \ldots \quad k \in \mathbb{N},$$

with

$$T_i^{k+1} = T_i^k + \gamma_i = k\gamma_i, \quad k \in \mathbb{N}. \tag{4}$$

For instance, $\gamma_{WG_4} < \gamma_{wg_4}$ means that wingless protein in the fourth cell is translated at a faster rate (shorter time intervals) then *wingless* mRNA is produced. At any given time t, the next node(s) to be updated is(are) i such that $T_i^k = \min_{j,\ell}\{T_j^\ell \geq t\}$, for some k. The variables X_i are updated according to:

$$X_i(T_i^k) = F_i(\, X_1(\tau_{1i}^k), \ldots, X_N(\tau_{Ni}^k)\,), \tag{5}$$

where τ_{ji}^k defines the most recent instant when node j was updated, that is

$$\tau_{ji}^k = \max_\ell \{\, T_j^\ell :\; T_j^\ell < T_i^k \,\}. \tag{6}$$

By ordering all the time sequences $\{T_i^k:\ i=1\ldots N,\ k=1,2,\ldots\}$, into a single nondecreasing sequence, say $\{t_1,t_2,\ldots\}$, the asynchronous model can also be written in the form

$$X_i(t_{k+1}) = \begin{cases} F_i(X_1(t_k), X_2(t_k), \ldots, X_N(t_k)), & \text{if } t_k = \ell\gamma_i \text{ for some } i,\ \ell \\ X_i(t_k), & \text{otherwise.} \end{cases}$$

It is clear that the steady states of this model must satisfy $X_i(t_{k+1}) = X_i(t_k) = F_i(X(t_k))$, and therefore are the same as those of the synchronous boolean model in Table 2.

Note that the case $\varepsilon = 0$ reduces to the synchronous model, where every node is updated simultaneously ($\gamma_i = 1$), at the same time instants: $T_i^k = k$, for all $i = 1,\ldots,N$. This algorithm allows great variability in each process' duration, exploring the gene expression patterns due to all possible combinations of individual timescales. Implementation of this asynchronous algorithm shows that, if started from the initial wild type state (1), any of the steady states of the model (Table 2) may occur with a certain probability. The probability of occurrence of each pattern depends on the range over which the individual time units γ_i are allowed to vary (see Fig. 2). For $\varepsilon = 0$, the wild type steady state is attained with probability 100% (corresponding to the synchronous Boolean model). As ε increases to 0.01 (resp. 0.1) this value decreases to 60% (resp. 44%). However, further increase in ε (hence larger time intervals) unexpectedly leads to an increase in the occurrence of the wild type state, up to 51% for $\varepsilon = 0.9$. Other final states observed are the broad-striped pattern ($25\% - 38\%$) observed in heat-shock experiments and *ptc* mutants [17] and the pattern with no segmentation ($12\% - 15\%$) observed in *en*, *hh* or *wg* mutants [18], the latter two corresponding to embryonic lethal phenotypes [17]. Each of the other three steady states occurs with frequencies less than 5%. (These values were obtained from 10000 numerical experiments.)

A possible extension of this algorithm would be to consider a discrete model with a finite number of logical levels describing ON, OFF as well as other intermediate states of the system [7]. This would involve decisions about the number of intermediate states, their values, and development of new transition rules. Instead, in this paper we will focus on a "hybrid" model, that takes into account the continuous nature of the biological processes, while still using Boolean rules to describe ON/OFF transitions (Section 4).

3.1 Random order algorithm

For comparison purposes, we briefly summarize an alternative asynchronous algorithm, which guarantees that every node is updated exactly once during each unit time interval [12]. A random order of updates for the N nodes is generated as a permutation ϕ^k of $\{1,\ldots N\}$. This permutation is randomly chosen out of a uniform distribution over the set of all $N!$ possible permutations, at the beginning of the time unit k. The updating times for each node are now written as

$$T_i^k = N(k-1) + \phi^k(i), \quad k \in \mathbb{N},$$

so that $\phi^k(j) < \phi^k(i)$ implies $T_j^k < T_i^k$, and node j is updated before node i at the k-th iteration. The results of this algorithm are qualitatively similar to those of the asynchronous algorithm (Table 2).

4 Glass-type networks

The asynchronous algorithm defined by (4), (5) and (6) allows the introduction of distinct timescales for each regulation process in a Boolean model. We next propose an alternative method which provides a bridge between discrete and continuous approaches, resulting in a more realistic model, but without the necessity of specifying any kinetic or binding parameters (which are typically unknown). In this method,

Table 3: The frequencies of the six steady states observed with the three different methods when starting from the wild type initial condition.

Steady state pattern	Asynchronous algorithm	Random order algorithm	Glass-type model
wild type	44-51%	56%	89-100%
broad stripes	25-38%	24%	0-6%
no segmentation	12-15%	15%	0-3%
wild type variant	4-5.6%	4.2%	0-1%
ectopic	0.4-1%	0.98%	0%
ectopic variant	0.1-0.5%	0.68%	0%

the gene and protein levels are represented as continuous variables, and their time evolution is described by differential equations, but the interactions among nodes are still modeled by Boolean functions [13, 14, 15, 22]. Glass [13] introduced a class of piecewise linear differential equations that combine logical rules for the synthesis of gene products with linear (free) decay by describing each node with two variables, one discrete and one continuous. For simplicity of notation, in what follows we will let $\widehat{X}_i$ denote the continuous variable associated with node i, its discrete variable X_i, and the discrete variable's Boolean rule by F_i. The Glass-type model is then

$$\frac{d\widehat{X}_i}{dt} = -\widehat{X}_i + F_i(X_1, X_2, \dots, X_N), \quad i = 1, \dots, N. \tag{7}$$

At each instant t, the discrete variable X_i is defined as a function of the continuous variable according to a threshold value:

$$X_i(t) = \begin{cases} 0, & \widehat{X}_i(t) \le \theta \\ 1, & \widehat{X}_i(t) > \theta \,, \end{cases} \tag{8}$$

where $\theta \in (0, 1)$. The discrete variables X_i represent the ON and OFF levels of the nodes in the Boolean model. The underlying assumption in this Glass model is that the decay rate and activation threshold of each gene product is identical. Since the initial condition for the piecewise linear system (7) is also (1) (i.e., $X(0) = \widehat{X}(0)$) and $F_i \in \{0, 1\}$, it is easy to see that solutions of (7) evolve in the hypercube $[0, 1]^N$. Under these conditions, the limiting values "0" and "1" of the continuous variable $\widehat{X}_i$ represent, respectively, "absence of species i" and "maximal concentration of species i" – thus we can view the $\widehat{X}_i$ as dimensionless variables, scaled to attain their maximal values at 1. The continuous dynamics is translated into a Boolean ON/OFF response, according to θ: as soon as $\widehat{X}_i$ increases above θ, species i is considered to be in the ON state; otherwise it remains in the OFF state (see also [22]). Thus the parameter θ defines the fraction of "maximal concentration" necessary for a protein or mRNA to effectively perform its biological function. This method allows us to study the continuous evolution of the genetic network simply by specifying θ, the fraction of maximal concentration that is effective as ON level, avoiding the need to specify any kinetic parameters. Below and in Section 6.4, we will see that system (7) exhibits distinct dynamics in the two regions $\theta \le 1/2$ and $\theta > 1/2$. It is easy to see that the steady states of the piecewise linear equations (7) are still those of the Boolean model, since:

$$\frac{d\widehat{X}_i}{dt} = 0 \;\Leftrightarrow\; \widehat{X}_i = F_i(X_1, X_2, \dots, X_N),\; i = 1, \dots, N,$$

independently of θ. Applying this method to the *Drosophila* segment polarity gene network, we find an exact convergence to the wild type steady state when started from the wild type initial condition (Fig. 1),

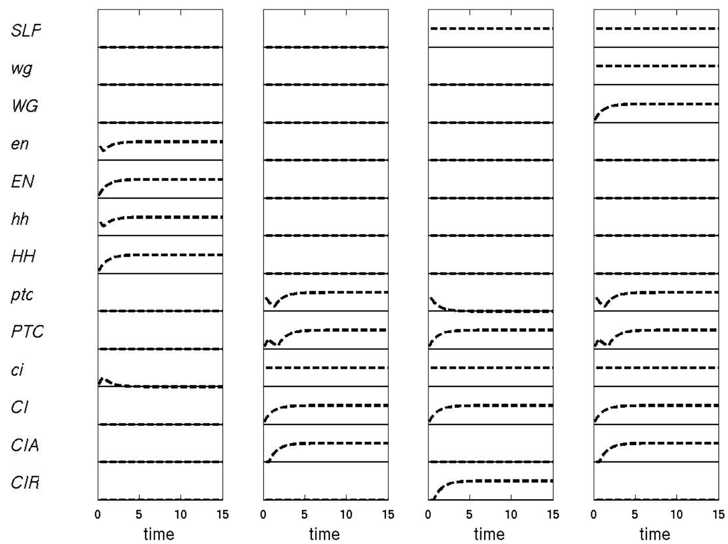

Figure 1: The solution to the system of piecewise linear equations (7) (dashed lines). Each column represents one cell, and each of the 13 rectangles represents the continuous-time variable for proteins or mRNAs (as labeled at left). In each rectangle, the y-axis ranges from 0 to 2 units. The nodes for which the trajectories converge to 1 (middle of the rectangle) are exactly those expressed in the wild type steady state. (The time units are arbitrary.)

independently of the ON/OFF threshold value θ. This result supports the Boolean model as a suitable description of the underlying network of gene interactions.

4.1 Introducing distinct timescales

The assumption of equal decay rates and activation thresholds for all nodes is an oversimplification similar to that made in synchronous Boolean models. However, for further robustness analysis, one may introduce different timescales for the different processes, by scaling the time units in each differential equation according to:

$$\frac{d\widehat{X}_i}{dt} = \alpha_i(-\widehat{X}_i + F_i(X_1, X_2, \ldots, X_N)), \tag{9}$$

with $\alpha_i \geq \varepsilon$ for some fixed $\varepsilon > 0$, $i = 1, \ldots, N$. Each X_i is a discrete variable defined as before (note that steady states of this new system are still those of the Boolean model).[1]

This method represents a continuous equivalent to the asynchronous algorithm described in Section 3. Here, the (inverse) scaling factors α_i^{-1} may be viewed as half-lives of mRNA or proteins. These may be directly compared to the individual time units γ_i as follows. Using Euler's method to discretize system (9), one obtains:

$$\widehat{X}_i(t + \Delta t) = \widehat{X}_i(t) + \alpha_i \Delta t(-\widehat{X}_i(t) + F_i(X(t))).$$

[1]We will analyze the behaviors of trajectories of systems of the form (7), assuming that trajectories are well-defined. Since the right-hand sides of equations of these type are discontinuous, it is very difficult to give general existence and uniqueness theorems for solutions of inital-value problems. One must impose additional assumptions, insuring that only a finite number of switches can take place on any finite time interval, and often tools from the theory of differential inclusions must be applied, see for instance [23] for more discussion. See also [22].

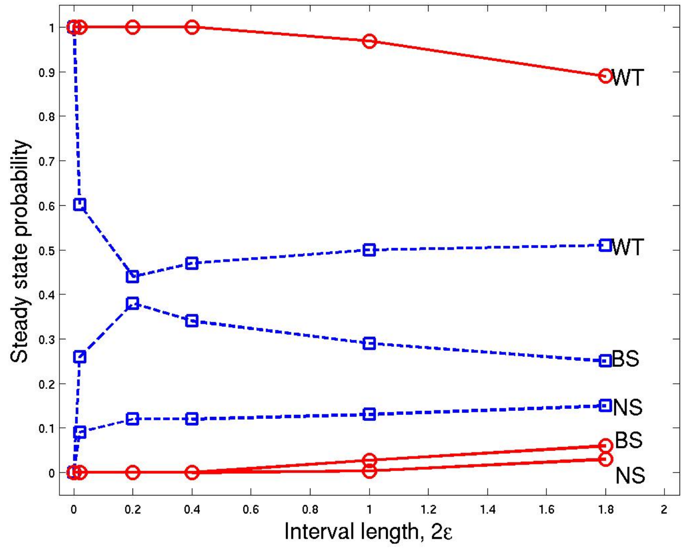

Figure 2: Probability of occurrence of the three most frequent patterns: wild type (WT), broad stripes (BS), and no segmentation (NS), under variable range of timescales. Dashed lines/squares represent asynchronous algorithm results, while solid lines/circles represent Glass-type model results (out of 1000 runs). Results were obtained with $\theta = 0.5$.

Now notice that choosing the integrating time interval to be such that $\alpha_i \Delta t = 1$ recovers the discrete asynchronous algorithm with specific time units

$$\gamma_i = \Delta t = \alpha_i^{-1}.$$

For comparison to the discrete algorithm, we choose both the scale factors α_i^{-1} and the time units γ_i randomly from a uniform distribution in intervals of the form $[1-\varepsilon, 1+\varepsilon]$, $\varepsilon \in (0,1)$. The numerical experiments are reported in Fig. 2, where the threshold value θ (8) was set to $1/2$. We again observed that, starting from the wild type initial condition (1), all steady states may occur with a certain frequency (see Table 3). But, in contrast to the asynchronous Boolean model, the wild type pattern occurs with frequencies that decrease monotonically with ε, down to 89% for $\varepsilon = 0.9$ (Fig. 2). The next more frequently achieved patterns are the broad-stripes (Fig. 3), with probability 6% for $\varepsilon = 0.9$, the no segmentation (Fig. 4), with probability 3%, and the wild type variant, with probability 1%.

The three methods we have described (Table 3) produce qualitatively compatible results, in the sense that the wild type pattern is always the most frequently occurring steady state, followed by the broad stripes, no segmentation, and wild type variant patterns.

4.2 Fraction of maximal concentration that defines an ON state

The piecewise linear system (9) follows the threshold (8) to decide whether a given node is ON or OFF. While this value θ did not affect the dynamics of the system in the case $\alpha_1 = \alpha_2 = \cdots = \alpha_N$, it plays a significant role in the general case. In Figure 5, it is immediate to see that the effect of θ depends on the length of the interval allowed for the timescales. Indeed, there is a marked difference between narrow ($\varepsilon \leq 0.5$) and wide ($\varepsilon \geq 0.6$) intervals. For narrow intervals, numerical experiments indicate that, starting from initial condition (1), system (9) converges to wild type steady state with probability around 90% or more; whereas, for wider intervals, this probability may decrease down to 68% ($\varepsilon = 0.9$, $\theta = 0.9$).

On the other hand, the threshold value also divides the dynamics into two regions. For $\theta \leq 0.5$, the probability that initial condition (1) leads to the wild type steady state is above 90%, independently of ε. For

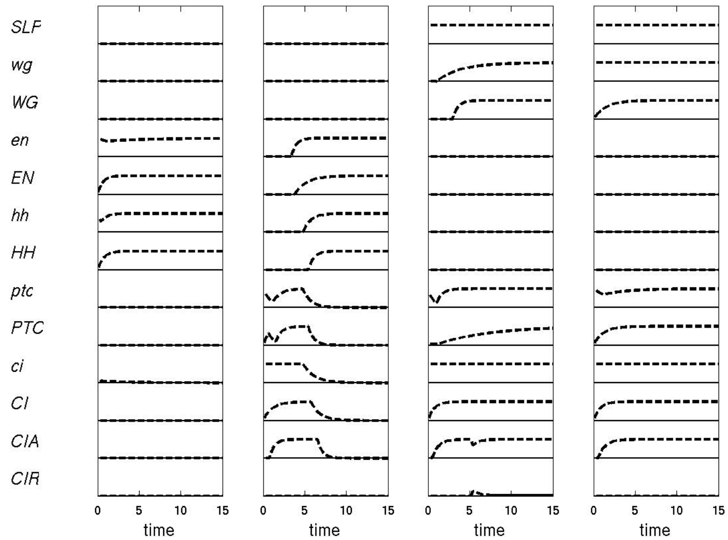

Figure 3: A solution to the system of piecewise linear equations (9) (dashed lines), in an example where the steady state corresponds to the broad stripes pattern. In each rectangle, the y-axis ranges from 0 to 2 units. Notice that *wingless* is expressed in two adjacent cells, as opposed to the wild type pattern, where *wingless* is expressed in only one cell (similarly for *engrailed* and *hedgehog*).

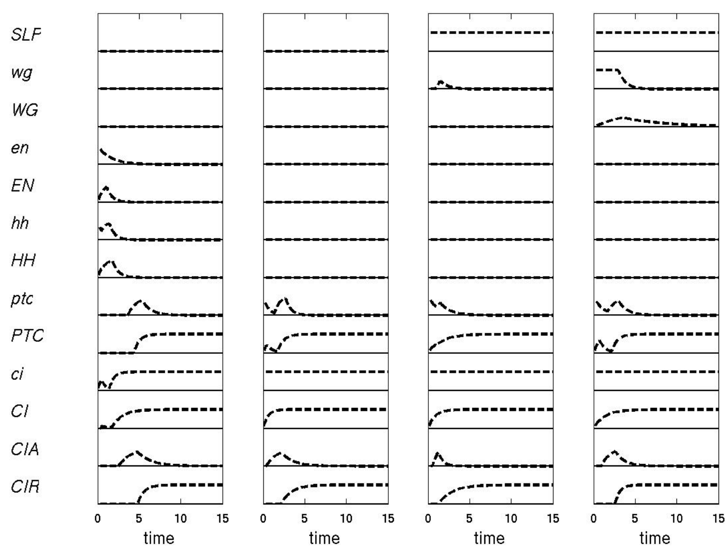

Figure 4: A solution to the system of piecewise linear equations (9) (dashed lines), in an example where the steady state corresponds to the no segmentation pattern. In each rectangle, the y-axis ranges from 0 to 2 units. Notice that *wingless*, *engrailed* and *hedgehog* are not expressed in any cell, thus no segments are visibly detected in the embryo.

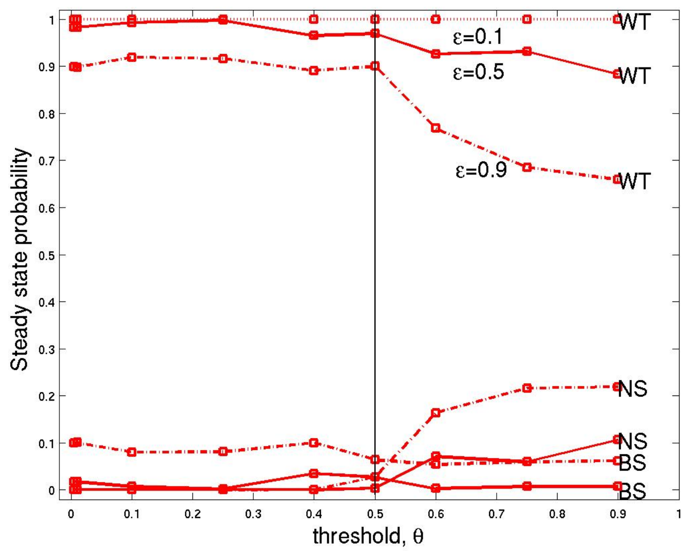

Figure 5: Effect of effective ON concentration, θ, on the probability of occurrence of the three most frequent patterns: wild type (WT), broad stripes (BS), and no segmentation (NS). The results are for the Glass-type model with α_i^{-1} randomly chosen in an interval $[1-\varepsilon, 1+\varepsilon]$. Solid lines represent the case $\varepsilon = 0.5$, dotted lines represent the case $\varepsilon = 0.1$ and dash-dotted lines represent the case $\varepsilon = 0.9$. (results out of 1000 runs).

$\theta \leq 0.5$ it is less probable to reach the non segmented (around 1%) than the broad striped pattern (around 10%). For higher $\theta \geq 0.6$, the probability of reaching the wild type steady state (from initial condition (1)) decreases very significantly, and in addition, it becomes more problable to reach the non segmented (around 20%) than the broad striped pattern (around 7%).

In our Glass-type model, θ represents the fraction of maximal concentration above which an mRNA or protein is considered ON, or biologically effective. Our results indicate that quite small fractions of the maximal concentration can (and should) be interpreted as sufficient amounts for an mRNA or protein to be in the ON state. Namely, when the concentration of mRNAs or proteins has increased to a fraction up to half its maximum possible value, it is already present in a sufficient amount to perform its function. This follows from observation of Figure 5: $0 < \theta \leq 0.5$ leads to a good (realistic) performance of the model, with 90% convergence to wild type pattern. Our results also indicate that a higher threshold is, on the contrary, not very realistic. For instance, setting $\theta \geq 0.6$ leads to a fairly high incidence on the mutant patterns. But by letting $\theta \geq 0.6$ one is assuming that an mRNA or protein is not present in a sufficient amount to perform its function until it reaches at least sixty per cent of its maximal value. Typically such high thresholds are not observed: indeed, in [11] (supplementary material) where continuous dynamics are also transformed into an ON/OFF response, a threshold of 10% is used, and considered very reasonable. In our numerical simulations, unless otherwise indicated, we have used $\theta = 0.5$.

To further generalize system (9), it is natural to allow each variable to respond to its own threshold and consider distinct θ_i values. We will address this problem in Section 6.3, and see that the system's dynamics is preserved in each θ region. In fact, our simulations with distinct θ_i recover many of the theoretical results we obtained for a species-independent θ (Section 6.4).

Table 4: The steady states corresponding to gene knockouts in the segment polarity network model, calculated according to (10). Here, for the *ci* knockout, the "wild type" state is interpreted as the wild type pattern in all but the *ci*, *CI*, *CIA*,*CIR* mRNA/proteins.

Knockouts	Mutant steady states
wg, *en*, hh	no segmentation
ptc	broad stripes
ci	wild type, broad stripes , ectopic

5 Pre-patterning errors and knockout mutant situations

In [8, 12] we identified some sufficient or necessary initial conditions for obtaining the wild type steady state (minimal pre-patterns). We now analyze how mutant patterns arise from gene "knockout" experiments or delay in establishing the initial pre-pattern.

Gene knockout experiments consist of completely supressing the expression of a given gene in all cells. In our models this is equivalent to setting the corresponding mRNA permanently zero in all equations. Thus a *wingless* knockout can be analyzed by setting $wg_i = 0$ for all $i = 1, 2, 3, 4$, and in every equation of the Boolean rules in Table 1. The steady states of the resulting system can now be computed from:

$$X^* = F^*(X^*), \tag{10}$$

where the vector X^* and functions F^* include all but the knockout variables. For example, for the *wingless* knockout, we drop the functions F_{wg_i} and for the other mRNAs and proteins we have

$$\begin{aligned}
&WG_i = F^*_{WG_i} = 0, \;\; i = 1, 2, 3, 4 \\
&\text{node } = F^*_{\text{node}} = F_{\text{node}}, \;\; \text{for nodes } en, hh, ci, ptc, EN, HH, PTC, CI, CIA, CIR,
\end{aligned}$$

because *wg* appears explicitly only in the rules F_{WG_i}. So, it follows immediately that $WG_i = 0$, and the remaining nodes are then also easy to compute. It is easy to check (see also [8]) that knockouts of *wg*, *en*, *hh* or *ptc* exhibit only one steady state, while knockouts of *ci* exhibit three steady states (summarized in Table 4) in both asynchronous and Glass-type models.

From another point of view, one may consider a delay in the establishment of the pre-pattern (that is, the full initial condition (1)). If expression of a given gene is delayed, does the system recover, and how soon? To answer this question, we simulated a delay in expression of gene X, by setting the corresponding discrete variable $X(t) = 0$ in all cells, for all $t \leq T_{\text{delay}}$. We then varied T_{delay} between 0 and 7 time units, and measured the frequency of occurrence of each steady state, both for the asynchronous and Glass-type models. (In the latter we set the concentration threshold θ equal to $1/2$ for all nodes.)

The results are shown in Figures 6 and 7.

We can see that, for short T_{delay}, both models recover their original frequencies of occurrence of each steady state, while for long T_{delay} both models converge to the corresponding mutant steady state. A curious exception is the case of *ci*, where long delays in its initial expression do not significantly change the probability that the wild type steady state is achieved (and even slightly increase it in the asynchronous model). This agrees with the conclusion of [8] that *cubitus interruptus* knockout coupled with an otherwise wild-type initial condition converges to a state close to the wild type steady state. Remarkably, delays in both

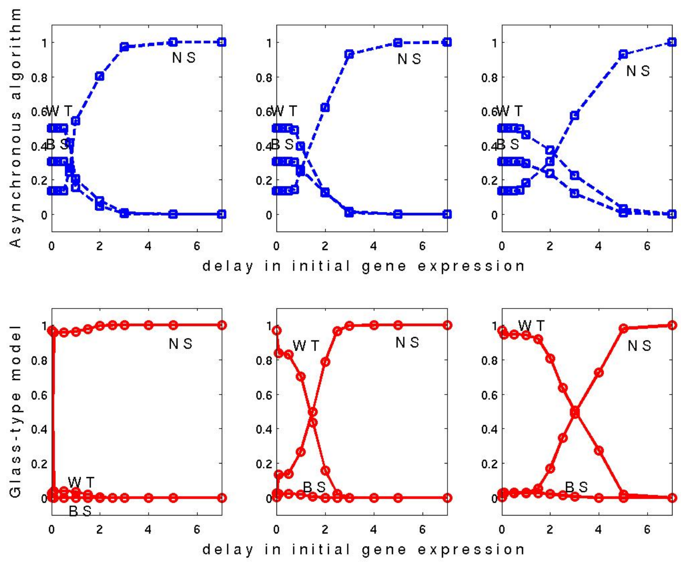

Figure 6: Effect of initially delayed expression in occurrence of steady state patterns. Left: delay in *wingless* expression. Middle: delay in *engrailed* expression. Right: delay in *hedgehog* expression. (results out of 1000 runs).

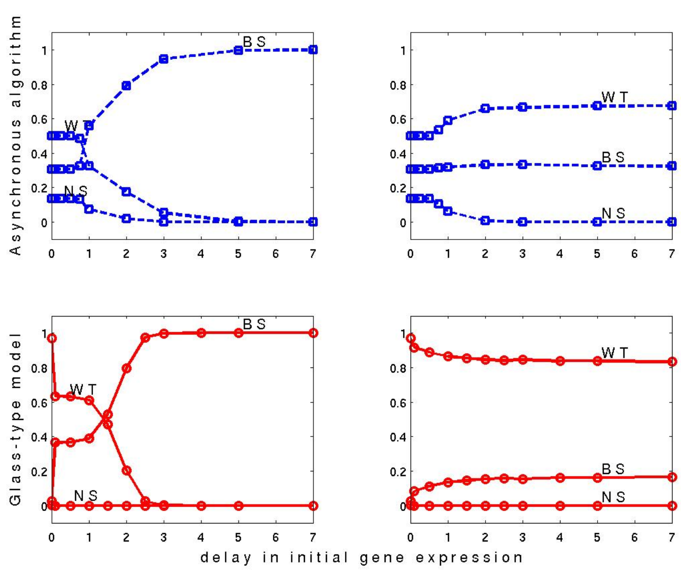

Figure 7: Effect of initially delayed expression in occurrence of steady state patterns. Left: delay in *patched* expression. Right: delay in *cubitus interruptus* expression. (results out of 1000 runs).

cubitus interruptus transcription factors (CIA, CIR) have a lesser effect than an imbalance in their expression (see [12]). This leads us to predict that, during the pre-segmentation stage of embryo development, the cubitus interruptus proteins' expression is the last to be established.

Another noteworthy observation is the fact that small delays in *wg* expression have much more drastic effects on the system than the same delay in *en* or *hh*. This phenomenon reflects the one-way signaling cascade starting with expression of *wg*, which induces *en*, which in turn induces *hh*. We see that a total disruption in the system is caused by a delay of only 3 time units in *wg* or *en* expression, which cause the system to fail to reproduce the wild type pattern, and settle into a non-segmented pattern. However, if only *hh* is delayed, the system is disrupted only after a delay of 5 time units. In other words, recovery of the system back to the "good" developmental process is more probable in the event of a *hh* expression delay, than a *wg* or *en* expression delay.

6 Robustness of the model under timescale separation

In the previous algorithms, the space of all possible timescales for protein/mRNA regulatory processes was explored, with no assumptions on the characteristic duration of translational or post-translational processes. As a consequence, the robustness analysis shows that the model diverges from the wild type pattern very often, with the biologically inviable states occurring with a noticeable frequency. However, it is also well known that post-translational processes such as protein conformational changes or complex formation, usually have shorter durations than transcription, translation or mRNA decay. This fact justifies the introduction of a distinct timescale separation among processes, by choosing to update proteins first and mRNAs later.

6.1 Timescale separation in the random order algorithm

Timescale separation is straightforwardly implemented in the random order algorithm presented in Section 3. At the k-th updating step we generate two random permutations, ϕ^k_{Prot} and ϕ^k_{mRNA}, within the set of proteins and mRNAs, respectively. Then the N nodes are updated in the order given by

$$\phi^k = (\phi^k_{Prot},\ \phi^k_{mRNA}).$$

This method again shows that the Boolean model is very robust, in the sense that when started from the wild type initial condition, the wild type pattern occurs with a frequency of 87.5% and only one other steady state is observed, the broad striped pattern, with a frequency of 12.5%. Furthermore, these frequencies are exact as we show in [12], where we also completely characterize the model resulting from incorporation of a protein/mRNA timescale separation into the random order algorithm. We show that the wild type state is in fact an attractor for the system, while the pathway to the broad stripes state may exhibit oscillatory cycles. We summarize this and other results in the next theorem, stated without proof, and refer to [12] for more details.

Theorem 1 In the random order algorithm with timescale separation, let $wg^0_3 = 0$, $ptc^0_3 = 1$, $hh^0_{2,4} = 0$ and $ci^0_3 = 1$ (as satisfied by the initial pattern (1)). Then system diverges from the wild type pattern if and only if the permutation ϕ^1 satisfies the following sequence among the proteins *CI*, *CIA*, *CIR* and *PTC*:

$$\begin{array}{cccccc} CIR_3 & CI_3 & & CIA_3 & & PTC_3, \\ & CI_3 & CIR_3 & CIA_3 & & PTC_3, \\ & CI_3 & & CIA_3 & CIR_3 & PTC_3, \end{array} \tag{11}$$

while the other proteins may appear in any of the remaining slots. ∎

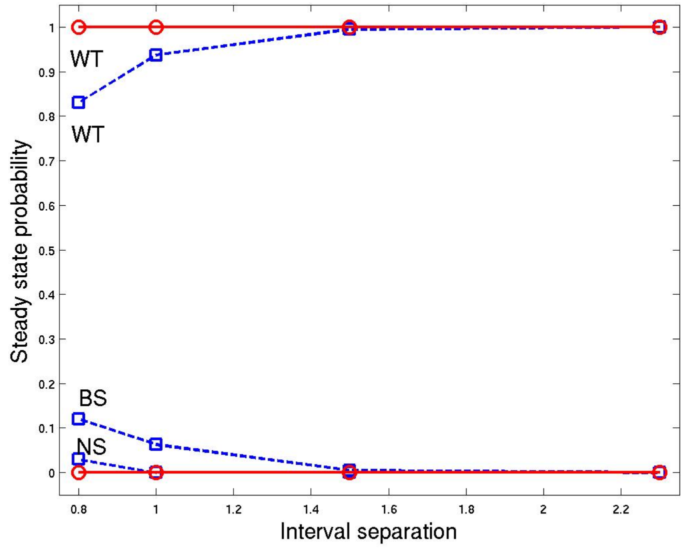

Figure 8: Probability of occurrence of the three most frequent patterns: wild type (WT), broad stripes (BS), and no segmentation (NS), with separation of timescales. Dashed lines/squares represent asynchronous algorithm results, while solid lines/circles represent Glass-type model results. The x-axis represents the level of separation, computed by $\min\{b \in A_{Prot}\}/\max\{a \in A_{mRNA}\}$ (out of 1000 runs). Results were obtained with $\theta = 0.5$.

Thus we can compute the exact probability with which the random order algorithm (with timescale separation) leads to either the wild type or broad stripes pattern: the latter is simply the fraction of sequences of the form (11), and equals 12.5% [12].

6.2 Timescale separation in the Glass-type and asynchronous algorithms

As shown in Section 4, the (discrete) asynchronous algorithm and the (piecewiese continuous) Glass-type system provide equivalent representations of a gene expression network. Indeed, the "specific time units" γ_i and the inverse "scaling factors" α_i^{-1}, both represent the rate of dynamical evolution of each individual node. For these two models, we implement time separation among processes by using two non-overlapping intervals for the scaling factors:

$$\begin{aligned} &\gamma_i^{-1}, \alpha_i \in A_{mRNA}, \quad \text{if } \ X_i \in \{wg, en, hh, ptc, ci\} \\ &\gamma_i^{-1}, \alpha_i \in A_{Prot}, \quad \text{if } \ X_i \in \{WG, EN, HH, PTC, CI, CIA, CIR\}, \end{aligned}$$

with, for instance, $A_{mRNA} = [0.2, 0.6]$ and $A_{Prot} = [1.4, 1.8]$. Under these conditions, choosing the factors α_i from a uniform distribution in these intervals, numerical experiments indicate that the two methods respond in mostly similar ways, with only two patterns occurring at steady state when the systems start from (wild type) initial condition (1). The two possible steady states are the wild type and broad stripes patterns.

For the asynchronous algorithm, the probabilities of convergence to each of the steady states clearly depend on the *distance* between the two intervals: convergence to wild type is 93% in the case where intervals A_{mRNA}, A_{Prot} are consecutive, and up to 100% for the case $2a < b$, $a \in A_{mRNA}$, $b \in A_{Prot}$.

For the Glass-type model, two cases can be distinguished: $\theta \leq 0.5$ and $\theta > 0.5$. For $\theta \leq 0.5$, numerical simulations show that the model reaches wild type pattern with probability near 100%, even when there is some overlap between A_{mRNA} and A_{Prot} (see Fig. 8).

Table 5: Probabilities of convergence to a given steady state, under the separation of timescales assumption. These values are theoretically exact for both the random order algorithm and Glass-type model.

Steady state pattern	Random order algorithm	Asynchronous algorithm	Glass-type model
wild type	87.5%	93.7-100%	100%
broad stripes	12.5%	0-6.3%	0%
other	0%	0%	0%

In fact, we next theoretically prove that *the wild type pattern is indeed the unique possible steady state* of the hybrid system (9) and initial condition (1), as indicated by the simulations, when there is a suitable distance between the intervals, and a lower bound on θ (Theorem 2, Section 6.4). For $\theta > 0.5$, we have found no condition that guarantees convergence to the wild type steady state, and indeed numerical simulations show that, even for large interval separation, the system may converge to one of the mutant patterns.

A comparison of Theorems 1 and 2 emphasizes differences and similarities between discrete and continuous models: intuitively, the single discrete event described by Theorem 1 cannot take place in a continuous model. Therefore wg_3 remains "0" (OFF) for all times, ruling out the possibility that the broad stripes pattern is reached. Indeed, Theorem 1 establishes that (in the discrete case, with the random order algorithm) divergence from wild type pattern occurs if and only if $wg_3^1 = 1$. This fact involves a jump in wg_3 from "0" to "1" at precisely the first iteration. On the other hand, in the Glass-type model, the continuous variable $\widehat{wg}_3$ cannot instantaneously jump from "0" to "1". Since the discrete ON/OFF levels are defined by a threshold on $\widehat{wg}_3$, there will necessarily be a smoothing effect on any transition between "0" and "1". This is what happens in case (a) of Theorem 2.

The second (sufficient) condition of Theorem 2 guarantees convergence to the wild type steady state for all $0 < \theta \leq 0.5$ (while condition (a) is only for $0.382 \leq \theta \leq 0.5$), but assumes that $\alpha_{PTC_3} > \alpha_{CI_3}$. This is an analog to Theorem 1: if $\alpha_{PTC_3} > \alpha_{CI_3}$, then (starting from $PTC_3(0) = CI_3(0) = 0$ and assuming $F_{PTC_3} = F_{CI_3} = 1$) $\widehat{PTC}_3$ increases faster than $\widehat{CI}_3$, implying that PTC_3 becomes ON faster than $\widehat{CI}_3$. Such response prevents the events listed in Theorem 1, which would lead to a mutant state. Thus, both discrete and piecewise linear model predict that the sequence of *PTC*, *CI* expression in the third cell is one of the fundamental pieces in establishing the correct development of embryo segmentation.

It is also worth pointing out that the three methods provide qualitatively similar results under timescales separation, all predicting that only wild type and broad striped patterns to occur, the latter with considerably smaller frequency.

6.3 Distinct concentration thresholds for ON state

A natural question arising in the analysis of equation (9) concerns the dynamics of the system under more general concentration thresholds. We have seen that θ (even when equal for all species) plays an important role in establishing basins of attraction to each of the steady states of the model.

We have, in particular, identified three distinct regions of behavior:

$$\text{Region 1}: \quad 0 < \theta < \tfrac{3-\sqrt{5}}{2}$$

$$\text{Region 2}: \quad \tfrac{3-\sqrt{5}}{2} \leq \theta \leq \tfrac{1}{2}$$

$$\text{Region 3}: \quad \tfrac{1}{2} < \theta \leq 1$$

Table 6: Probabilities of convergence to a given steady state, with distinct concentration thresholds θ_i and distinct timescales α_i, in the Glass-type model. (Probabilities computed out of 1000 simulations for each case.)

Steady state pattern	$(0,1)$	$(0.5,1)$	$(0,0.5]$	$(0,0.5]$ $\alpha_{PTC_3} > \alpha_{CI_3}$	$[0.4,0.5]$	0.5	θ_i / α_i
wild type	45.6%	57.1%	84.1%	90%	92.6%	94.2%	$A_{mRNA} =$
broad stripes	27.8%	15.1%	12%	6.2%	7.3%	4.5%	$A_{Prot} =$
no segmentation	24.4%	25.8%	0.9%	0.9%	0.05%	1.3%	$[0.5, 1.5]$
wild type, variant	2.1%	1.9%	2.9%	2.9%	0%	0%	
wild type	74.1%	52.7%	96.6%	97.1%	100%	100%	$A_{mRNA} =$
broad stripes	10.8%	3.3%	3.3%	2.8%	0%	0%	$[0.2, 0.6]$
no segmentation	14.1%	43.9%	0%	0%	0%	0%	$A_{Prot} =$
wild type, variant	1.0%	0%	0.1%	0.1%	0%	0%	$[1.4, 1.8]$

When θ is equal for all nodes, the probability that the system evolves into the wild type pattern is above 90% in regions 1 and 2, but may be as low as 68% in region 3. Furthermore, for region 2, we theoretically prove that that probability is exactly 100% under the timescale separation assumption.

We now associate to each node a specific θ_i, so that (8) is modified to:

$$X_i(t) = \begin{cases} 0, & \widehat{X}_i(t) \leq \theta_i \\ 1, & \widehat{X}_i(t) > \theta_i \ . \end{cases} \tag{12}$$

To test the performance of the system and compare it to previous results, we considered two timescale situations: $\alpha_i \in [0.5, 1.5]$ for all i, or the timescale separation $A_{mRNA} = [0.2, 0.6]$, $A_{Prot} = [1.4, 1.8]$. In each case, we randomly assigned values to θ_i from uniform distributions in the intervals $(0, 1)$, $(0, 0.5)$ and $(0.4, 0.5)$. (Note that 0.4 is close to $(3 - \sqrt{5})/2$.) The numerical results with varying θ_i extend and confirm our previous observations for $\theta_i = \theta$, $i = 1, \ldots, N$.

Table 6 summarizes the results for all combinations of θ_i and α_i regions. The most general case, allowing a large degree of freedom in both timescales and concentration thresholds, indicates the vulnerability of the network, with a very high incidence on mutant patterns ($\theta_i \in (0, 1)$, $\alpha_i \in [0.5, 1.5]$). Again we see that there is a marked difference in the θ_i regions below or above 0.5. Comparing the reasonable results obtained for $\theta_i \leq 0.5$ with the bad performance for $0.5 < \theta_i < 1$, we conclude that the optimal ON concentratin for proteins or mRNA is below 50% of maximal concentration.

Restricting θ_i even further to one of the conditions given in Theorem 2 (Section 6.4; conditions developed for the case $\theta_i = \theta$, $i = 1, \ldots, N$) dramatically increases the probability that the system develops in the correct way.

These simulations suggest, moreover, that some of the theoretical results obtained for each of the three regions may extend to the case of distinct θ_i. Indeed, note that, under timescale separation, when θ_i are chosen from region 2, convergence to wild type steady state is 100% – compare to part (a) of Theorem 2.

These simulations further confirm the role of α_{PTC_3} and α_{CI_3} in the segmentation network. Requiring $\alpha_{PTC_3} > \alpha_{CI_3}$ decreases the probability of formation of the broad stripes pattern (any timescales), but doesn't influence the probability of a non segmented embryo. The latter mutant is prevented only by a complete separation of timescales of the regulatory processes.

6.4 Glass-type model provides exact convergence to wild type pattern

In this section we will require that the intervals A_{mRNA} and A_{Prot} do not overlap, by satisfying the following assumption:

$$\text{For all} \quad a \in A_{mRNA} \;\text{ and }\; b \in A_{Prot}: \quad 0 < 2a < b. \tag{13}$$

A second assumption is that the effective maximal concentration is equal for all nodes and satisfies $\theta \leq 1/2$, which is equivalent to:

$$\ln \frac{1}{1-\theta} \leq \ln \frac{1}{\theta} \tag{14}$$

Theorem 2 Consider system (9) with initial condition (1). Assume that the scaling factors α_i satisfy (13). Assume also that one of the following conditions holds:

(a) $\theta \leq 1/2$ and $(1-\theta)^2 \leq \theta$ or equivalently $0.382 \approx (3-\sqrt{5})/2 \leq \theta \leq 1/2$;

(b) $\theta \leq 1/2$ and $\alpha_{PTC_3} > \alpha_{CI_3}$;

then $wg_3(t) = 0$ for all t.

This shows that the steady state representing the broad stripes pattern cannot ever be reached in system (9) from the initial condition (1), when $\theta \leq 1/2$ and either of the extra conditions holds.

Theorem 3 Consider system (9) with initial condition (1). Assume that the scaling factors α_i satisfy (13), and that $\theta \leq 1/2$. Then $wg_4(t) = 1$ and $PTC_1(t) = 0$ for all t.

This shows that the steady states represented by the no segmentation, wild type variant or the two ectopic patterns also cannot ever be reached in system (9) from the initial condition (1). From Theorems 2 and 3 we conclude that, under the timescale separation assumption, the Glass-type model (9) can only converge to the wild type pattern, when starting from the initial condition (1), and for appropriate θ values (Table 5).

We first summarize some useful observations. Let X denote any of the nodes in the network, and α its time rate. Since equations (9) are either of the form $d\widehat{X}/dt = \alpha(-\widehat{X}+1)$ or $d\widehat{X}/dt = -\alpha\widehat{X}$, their solutions are continous functions, piecewise combinations of:

$$\begin{aligned} \widehat{X}^1(t) &= 1 - (1 - \widehat{X}^1(t_0))\, e^{-\alpha(t-t_0)} \quad (15)\\ \widehat{X}^0(t) &= \widehat{X}^0(t_0)\, e^{-\alpha(t-t_0)} \quad (16) \end{aligned}$$

$\widehat{X}^1(t)$ (resp. $\widehat{X}^0(t)$) is monotonically increasing (resp. decreasing). In addition, note that discrete variables X can only switch between 0 and 1 at those instants when $\widehat{X}(t_{\text{switch}}) = \theta$, that is:

$$\begin{aligned} t^1_{\text{switch}} &= t_0 + \frac{1}{\alpha}\, \ln \frac{(1-\widehat{X}(t_0))}{1-\theta} \quad (17)\\ t^0_{\text{switch}} &= t_0 + \frac{1}{\alpha}\, \ln \frac{\widehat{X}(t_0)}{\theta} \quad (18) \end{aligned}$$

From the initial conditions, together with the constant values of SLP_i ($i = 1,2,3,4$), we can immediately conclude:

$$\widehat{wg}_{1,2}(t) = WG_{1,2}(t) = 0, \tag{19}$$

$$\begin{aligned} &\widehat{en}_{3,4}(t) = \widehat{EN}_{3,4}(t) = 0,\\ &\widehat{hh}_{3,4}(t) = \widehat{HH}_{3,4}(t) = 0, \end{aligned} \tag{20}$$

for all $t \geq 0$. Then, because $ci_{3,4}(0) = 1$ and $F_{ci_{3,4}} = \text{not } EN_{3,4}$,

$$\widehat{ci}_{3,4}(t) = 1 \;\text{ and }\; \widehat{CI}_{3,4}(t) = 1 - e^{-\alpha_{CI_{3,4}}\, t}. \tag{21}$$

Lemma 6.1 Let $0 \le t_0 < t_3 \le t_1$ and $0 \le t_2 < t_3$. Define $\delta = \ln \frac{1}{1-\theta} / \max_{1,\ldots,N} \alpha_i$. Assume $CIA_3(t) = 0$ for $t \in (t_2, t_3)$, and $wg_3(t) = 0$ for $t \in [0, t_3)$. Then

(a) $wg_3(t) = 0$ for $t \in [0, t_3 + \delta)$;

(b) $WG_3(t) = 0$ for $t \in [0, t_3 + \delta)$;

(c) $en_2(t) = EN_2(t) = 0$ for $t \in [0, t_3 + \delta)$;

(d) $hh_2(t) = HH_2(t) = 0$ for $t \in [0, t_3 + \delta)$.

Assume further that $PTC_3(t) = 1$ for $t \in (t_0, t_1)$. Then

(e) $PTC_3(t) = 1$ for all $t \in (t_0, t_3 + \delta)$.

(f) $CIA_3(t) = 0$ for all $t \in (t_2, t_3 + \delta)$.

Proof: Part (a) follows directly from the fact that $F_{wg_3}(t) = 0$ on $[0, t_3)$, and from (17).

To prove parts (b), (c), and (d), first note that initial conditions together with $wg_3(t) = 0$ for $t \in [0, t_3)$ imply

$$\widehat{WG}_3(t) = 0, \; \widehat{en}_2(t) = \widehat{EN}_2(t) = 0,$$
$$\widehat{hh}_2(t) = \widehat{HH}_2(t) = 0,$$

for $t \in [0, t_3]$. Then, from equations (15) to (18) we conclude that the corresponding discrete variables cannot switch from 0 to 1 during an interval of the form $[0, t_3 + \frac{1}{\alpha_j} \ln \frac{1}{1-\theta})$. Taking the largest common interval yields the desired results.

To prove parts (e) and (f), assume also that $PTC_3(t) = 1$ for $t \in (t_0, t_1)$. From (20) and part (d), it follows that function F_{PTC_3} does not switch in the interval $(t_0, t_3 + \delta)$ and in fact $PTC_3(t) = 1$ for all t in this interval. This, together with (20) and part (d) yield $F_{CIA_3}(t) = 0$ for $(t_0, t_3 + \delta)$, so that $\widehat{CIA}_3$ cannot increase in this interval and the discrete level satisfies $CIA_3(t) = 0$ for all $t \in (t_2, t_3 + \delta)$, as we wanted to show. ∎

Corollary 6.2 Let $0 \le t_0 < t_3 \le t_1$ and $0 \le t_2 < t_3$. If $PTC_3(t) = 1$ for $t \in (t_0, t_1)$, $CIA_3(t) = 0$ for $t \in (t_2, t_3)$, and $wg_3(t) = 0$ for $t \in [0, t_3)$, then $wg_3(t) = 0$ for all t.

Proof: Applying Lemma 6.1 we conclude that, given any $k \ge 0$:

$$\begin{aligned} &CIA_3(t) = 0, \;\text{ for }\; t \in (t_2, t_3 + k\delta) \\ &wg_3(t) = 0, \;\text{ for }\; t \in [0, t_3 + k\delta) \\ &PTC_3(t) = 1 \;\text{ for }\; t \in (t_0, t_3 + k\delta) \end{aligned}$$

imply

$$\begin{aligned} &CIA_3(t) = 0, \;\text{ for }\; t \in (t_2, t_3 + (k+1)\delta) \\ &wg_3(t) = 0, \;\text{ for }\; t \in [0, t_3 + (k+1)\delta) \\ &PTC_3(t) = 1 \;\text{ for }\; t \in (t_0, t_3 + (k+1)\delta). \end{aligned}$$

Since δ is finite, we conclude by induction on k that $wg_3(t) = 0$ for all t. ∎

Proof of Theorem 2: The rule for CIA_3 may be simplified to (by (20))

$$F_{CIA_3} = CI_3 \text{ and } [\text{not} PTC_3 \text{ or } hh_2 \text{ or } HH_2].$$

From equation (21), we have that

$$CI_3(t) = 1, \ \ \text{for all } t > \frac{1}{\alpha_{CI_3}} \ln \frac{1}{1-\theta}. \tag{22}$$

On the other hand, since $ptc_3(0) = 1$, by continuity of solutions $ptc_3(t) = 1$ for all $t < \frac{1}{\alpha_{ptc_3}} \ln \frac{1}{\theta}$. This implies that the Patched protein satisfies

$$\widehat{PTC}_3(t) = 1 - e^{-\alpha_{PTC_3} t}, \ \ 0 \le t \le \frac{1}{\alpha_{ptc_3}} \ln \frac{1}{\theta}$$

and therefore

$$PTC_3(t) = \begin{cases} 0, & 0 \le t \le \frac{1}{\alpha_{PTC_3}} \ln \frac{1}{1-\theta} \\ 1, & \frac{1}{\alpha_{PTC_3}} \ln \frac{1}{1-\theta} < t < \frac{1}{\alpha_{ptc_3}} \ln \frac{1}{\theta}. \end{cases} \tag{23}$$

By assumption, $\alpha_{PTC_3} > \alpha_{ptc_3}$ and also $\ln \frac{1}{1-\theta} \le \ln \frac{1}{\theta}$, defining a nonempty interval where PTC_3 is expressed. Now let $t_c = \frac{1}{\alpha_{CI_3}} \ln \frac{1}{1-\theta}$ and $t_p = \frac{1}{\alpha_{PTC_3}} \ln \frac{1}{1-\theta}$. $\widehat{CIA}_3(t)$ starts at zero and must remain so while $CI_3 = 0$, so that

$$CIA_3(t) = 0 \ \ \text{for } 0 < t < t_c.$$

In the case $t_c > t_p$, letting $t_0 = t_p$, $t_1 = \frac{1}{\alpha_{ptc_3}} \ln \frac{1}{\theta}$, $t_2 = 0$, and $t_3 = t_c$ in Corollary 6.2, obtains $wg_3(t) = 0$ for all t. This proves item (b) of the theorem, and part of (a).

To finish the proof of item (a), we assume that $(1-\theta)^2 < \theta$ and must now consider the case $t_c \le t_p$. Then

$$\widehat{CIA}_3(t) = \begin{cases} 0, & 0 \le t \le t_c \\ 1 - e^{-\alpha_{CIA_3}(t-t_c)}, & t_c < t \le t_p \\ \widehat{CIA}_3(t_p)\, e^{-\alpha_{CIA_3}(t-t_p)}, & t_p < t \le \frac{1}{\alpha_{ptc_3}} \ln \frac{1}{\theta}, \end{cases}$$

Following equation (17) with $t_0 = t_c$ and $\widehat{CIA}_3(t_0) = 0$, CIA_3 might become expressed at time $t_c < t_a < t_p$:

$$t_a = t_c + \frac{1}{\alpha_{CIA_3}} \ln \frac{1}{1-\theta},$$

but it would then become zero again at (equation (18) with $t_0 = t_p$)

$$t_b = t_p + \frac{1}{\alpha_{CIA_3}} \ln \frac{\widehat{CIA}_3(t_p)}{\theta}.$$

Finally, we show that, even if $CIA_3(t) = 1$ for $t \in (t_a, t_b)$, wg_3 cannot become expressed in this interval. In this interval, $\widehat{wg}_3$ evolves according to $\widehat{wg}_3(t) = 1 - e^{-\alpha_{wg_3}(t-t_a)}$, and wg_3 can switch to 1 at time

$$t_w = t_a + \frac{1}{\alpha_{wg_3}} \ln \frac{1}{1-\theta}.$$

We will show that $t_w > t_b$, so $wg_3(t) = 0$ in the interval $[0, t_b)$. Writing

$$\begin{aligned} \ln \frac{\widehat{CIA}_3(t_p)}{\theta} &= \ln \frac{\widehat{CIA}_3(t_p)}{1-\theta} \frac{1-\theta}{\theta} \\ &= \ln \frac{\widehat{CIA}_3(t_p)}{1-\theta} + \ln \frac{1-\theta}{\theta} \\ &\le \ln \frac{1}{1-\theta} + \ln \frac{1}{1-\theta} \end{aligned}$$

where we have used $\widehat{CIA}_3(t_p) \leq 1$ and the assumption on θ: $\frac{1-\theta}{\theta} \leq \frac{1}{1-\theta}$. Therefore

$$\begin{aligned} t_b &\leq t_p + \frac{2}{\alpha_{CIA_3}} \ln \frac{1}{1-\theta} \\ &< \frac{1}{\alpha_{wg_3}} \ln \frac{1}{1-\theta} + \frac{1}{\alpha_{CIA_3}} \ln \frac{1}{1-\theta} < t_w \end{aligned}$$

where we have used the timescale separation assumption (13). Letting $t_0 = t_p$, $t_2 = 0$, and $t_1 = t_3 = \min\{t_b, \alpha_{ptc_3}^{-1} \ln \frac{1}{\theta}\}$ in the Corollary, obtains $wg_3(t) = 0$ for all t. ∎

We will next show that if $wg_4(t) = 1$ in a given interval $[0, T)$, then in fact $wg_4(t)$ remains expressed for a longer time, up to $T + \delta$, with $\delta > 0$. This is mainly due to assumption (13), which says that mRNAs take longer than proteins to update their discrete values, because they have longer half-lives: $\alpha_{mRNA}^{-1} > \alpha_{Prot}^{-1}$. This allows the initial signal "$wg_4 = 1$" to travel down the network, sequencially affecting the wingless protein, *engrailed*, *hedgehog* and CIA, and feed back into *wingless* allowing wg_4 to remain expressed for a further time interval.

Lemma 6.3 Let $T \geq \frac{1}{\alpha_{wg_4}} \ln \frac{1}{\theta}$ and define

$$\delta = \frac{1}{\alpha_{WG_4}} \ln \frac{(1 - e^{-\frac{\alpha_{WG_4}}{\alpha_{wg_4}} \ln \frac{1}{\theta}})}{\theta}. \tag{24}$$

If $wg_4(t) = 1$ for $0 \leq t < T$, then

(a) $WG_4(t) = 1$ for $t \in (\frac{1}{\alpha_{WG_4}} \ln \frac{1}{1-\theta}, T + \delta)$;

(b) $en_1(t) = 1$ for $t \in [0, T + \delta)$;

(c) $\widehat{EN}_1(t) = 1 - e^{-\alpha_{EN_1} t}$ for $t \in [0, T + \delta)$, and $EN_1(t) = 1$ for $(\frac{1}{\alpha_{EN_1}} \ln \frac{1}{1-\theta}, T + \delta)$;

(d) $ci_1(t) = 0$, $CI_1(t) = 0$, $CIA_1(t) = 0$, and $CIR_1(t) = 0$ for $t \in [0, T + \delta)$;

(e) $hh_1(t) = 1$, for $t \in [0, T + \delta)$;

(f) $CIA_4(t) = 1$, for $t \in (\frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta} + \frac{1}{\alpha_{CIA_4}} \ln \frac{1}{1-\theta}, T + \delta)$, and $CIR_4(t) = 0$, for $t \in [0, T + \delta)$;

(g) $wg_4(t) = 1$ for $t \in [0, T + \delta)$.

Proof: Let $T \geq \frac{1}{\alpha_{wg_4}} \ln \frac{1}{\theta}$, and assume that $wg_4(t) = 1$ for $0 \leq t < T$. To prove part (a), note that $\widehat{WG}_4(t)$ is of the form (15) (with $t_0 = 0$, and $\widehat{WG}_4(0) = 0$) and the corresponding discrete variable is $WG_4(t) = 1$, for $t \in (\frac{1}{\alpha_{WG_4}} \ln \frac{1}{1-\theta}, T)$. Moreover, suppose that $wg_4(t) = 0$ for $t > T$, then

$$\widehat{WG}_4(t) = (1 - e^{-\alpha_{WG_4} T}) e^{-\alpha_{WG_4}(t-T)}, \quad t > T.$$

But WG_4 remains 1 until the switching threshold is attained, that is up to time

$$\begin{aligned} & T + \frac{1}{\alpha_{WG_4}} \ln \frac{(1 - e^{-\alpha_{WG_4} T})}{\theta} \\ \geq \; & T + \frac{1}{\alpha_{WG_4}} \ln \frac{(1 - e^{-\alpha_{WG_4} \frac{1}{\alpha_{wg_4}} \ln \frac{1}{\theta}})}{\theta} \\ \equiv \; & T + \delta. \end{aligned}$$

Thus we conclude that $WG_4(t) = 1$ in the desired interval.

To prove part (b), observe that $F_{en_1}(t) = WG_4(t)$ for all t, from (19), and recall that $en_1(0) = 1$. From part (a), $F_{en_1}(t) = 1$ for $t \in (\frac{1}{\alpha_{WG_4}} \ln \frac{1}{1-\theta}, T + \delta)$. On the other hand, en_1 can only switch from 1 to 0 at $t = \alpha_{en_1}^{-1} \ln \frac{1}{\theta}$ which is larger than $\alpha_{WG_4}^{-1} \ln \frac{1}{1-\theta}$. So, in fact, $en_1(t) = 1$ for all $0 \le t < T + \delta$.

Part (c) follows immediately by integration of the $\widehat{EN}_1$ equation.

To prove part (d), first recall $F_{ci_1} =$ not EN_1 and the initial conditions $ci_1(0) = 0 = CI_1(0) = CIA_1(0) = CIR_1(0)$. Therefore $\widehat{ci}_1(t)$ increases up to $t = \frac{1}{\alpha_{EN_1}} \ln \frac{1}{1-\theta}$ and then decreases in $\alpha_{EN_1}^{-1} \ln \frac{1}{1-\theta} < t < T + \delta$. Now note that the discrete variable $ci_1(t)$ remains 0 in the whole interval $[0, T+\delta)$. This is because $\widehat{ci}_1$ never reaches the θ threshold: this would be attained at some $t \ge \alpha_{ci_1}^{-1} \ln \frac{1}{1-\theta}$ but, since $\alpha_{ci_1}^{-1} \ln \frac{1}{1-\theta} > \alpha_{EN_1}^{-1} \ln \frac{1}{1-\theta}$, the function $\widehat{ci}_1$ starts decreasing before it could reach the value θ. Finally, from the rules of the Cubitus proteins it is immediate to see that $CI_1(t) = CIA_1(t) = CIR_1(t) = 0$ for $t \in [0, T + \delta)$.

To prove part (e), recall that $F_{hh_1} = EN_1$ and not CIR_1. From part (a), it follows that $F_{hh_1}(t) = 0$ in the interval $[0, \alpha_{EN_1}^{-1} \ln \frac{1}{1-\theta})$ and $F_{hh_1}(t) = 1$ in the interval $(\alpha_{EN_1}^{-1} \ln \frac{1}{1-\theta}, T + \delta)$. Since $hh_1(0) = 1$, $\widehat{hh}_1(t)$ decreases in the interval $[0, \alpha_{EN_1}^{-1} \ln \frac{1}{1-\theta})$ but increases in $(\alpha_{EN_1}^{-1} \ln \frac{1}{1-\theta}, T + \delta)$. The discrete value is $hh_1(t) = 1$ in the whole interval, since $\widehat{hh}_1(t)$ remains above the θ threshold. (The justification is similar to the case of $ci_1(t)$ in part (d).)

To prove part (f), note that part (e) and then the use of (21), allows us to simplify F_{CIA_4}:

$$F_{CIA_4}(t) = CI_4(t) \text{ and } hh_1(t) = 1, \quad t \in (\frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta}, T + \delta).$$

Thus

$$\widehat{CIA}_4(t) = \begin{cases} 0, & 0 \le t \le \frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta} \\ 1 - e^{-\alpha_{CIA_4}\left(t - \frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta}\right)}, & \frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta} < t \le T + \delta, \end{cases}$$

and $CIA_4(t) = 1$ for $t \in [\frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta} + \frac{1}{\alpha_{CIA_4}} \ln \frac{1}{1-\theta}, T + \delta)$. Observe that this interval is indeed nonempty, by assumption (13). Finally, $F_{CIR_4}(t) = CI_4(t)$ and not$hh_1(t) = 0$, and hence $CIR_4(t) = 0$ for $t \in [0, T+\delta)$.

To prove part (g), we note that (from part (f))

$$F_{wg_4}(t) = 1, \quad t \in (\frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta} + \frac{1}{\alpha_{CIA_4}} \ln \frac{1}{1-\theta}, T + \delta),$$

implying that $\widehat{wg}_4(t)$ increases in this interval. On the other hand, we know that $\widehat{wg}_4(t) \ge \theta$ and $wg_4(t) = 1$ up to at least $t = \frac{1}{\alpha_{wg_4}} \ln \frac{1}{\theta} > \frac{1}{\alpha_{CI_4}} \ln \frac{1}{1-\theta} + \frac{1}{\alpha_{CIA_4}} \ln \frac{1}{1-\theta}$. This shows that in fact $wg_4(t) = 1$ for all $t \in [0, T + \delta)$. ∎

Proof of Theorem 3: Since $wg_4(0) = 1$, from equations (17), (18), we know that the earliest possible switching time from 1 to 0 is $\alpha_{wg_4}^{-1} \ln \frac{1}{\theta}$. Applying Lemma 6.3 with $T = \alpha_{wg_4}^{-1} \ln \frac{1}{\theta}$ establishes that $wg_4(t) = 1$ for $t \in [0, T + \delta)$, with δ given by (24). Next, applying Lemma 6.3 with $T = \alpha_{wg_4}^{-1} \ln \frac{1}{\theta} + k\delta$, $k \in \mathbb{N}$, shows that $wg_4(t) = 1$ for $t \in [0, T + (k+1)\delta)$. Since δ is finite, we can conclude by induction that $wg_4(t) = 1$ for all $t \ge 0$.

To prove that $PTC_1(t) \equiv 0$, note that $CIA_1(t) \equiv 0$ (Lemma 6.3, with $T = +\infty$) implies $ptc_1(t) \equiv 0$. Since $PTC_1(0) = 0$ and PTC_1 cannot become expressed unless ptc_1 is first expressed, the desired result follows. ∎

7 Conclusions

We discussed two alternative methods for modeling gene expression networks: purely discrete Boolean methods and piecewise linear differential systems, which combine continuous degradation with discrete synthesis. For both methods we introduced new techniques for a deeper analysis of the networks with respect to perturbations in the timescales of the system. For the piecewise linear system we also studied the effect of the ON concentration thresholds.

We find that unrestricted variability in the duration of the diverse processes present in the network may lead to significant deviations from experimentally observed results, thus suggesting the fragility of the developmental process under severe perturbations (the asynchronous algorithm fails to predict the correct pattern with probability 50%, and the Glass-type system fails with probability at least 10%).

Another set of numerical experiments introduces a separation between timescales of post-translational and transcription/translation processes, and in practice "updates mRNAs later than proteins". In this context, the piecewise linear Glass-type system indicates a remarkable robustness of the Boolean model in predicting the final gene expression pattern. Indeed, we provide a theoretical proof that the Glass-type system always correctly generates the wild type development (i.e. the convergence to the wild type steady state when started from the wild type initial state), under the separation of timescales assumption, and appropriate OFF/ON thresholds. The asynchronous model's predictions depend very much on the degree of separation between timescales. As the intervals A_{Prot} and A_{mRNA} become closer, the asynchronous algorithm increasingly fails to generate the wild type pattern. This leads us to conclude that a strong separation between timescales of post-translational and transcription/translation processes is necessary for establishing the regular gene expression pattern in the segment polarity network. From analysis of the piecewise linear model we conclude that the fraction of maximal concentration above which a protein or mRNA is effectively ON needs to be quite small, below 50%. Higher concentration thresholds may disrupt the development process.

The comparison between discrete and continuous models shows clearly that sudden transitions may happen in discrete systems and lead to a false result; such sudden transitions are smoothed out in continuous models which prevent generation of false results (see also [24]). Nevertheless, we conclude that both models agree in predicting the fundamental sequences of gene expression that irreversibly lead to a deviation in the development towards a mutant state.

By combining continuous-time techniques with discrete events, we can with great generality explore and sample the space of all possible timescales as well as of effective ON levels. Moreover, as information about the mRNA/protein lifetimes, decay rates or activation thresholds becomes available, it can be straightforwardly incorporated by fixing the corresponding inverse scaling factor α_i^{-1}. The hybrid model retains the ease of Boolean models in determining the steady states corresponding to gene knockouts and perturbed initial conditions. It is straightforward to calculate which mutant patterns result from each gene knockout. Here we also studied the the effect of perturbations on the prepattern, by simulating delay in initial expression of each gene. We find that the system is vulnerable to large delays (larger than two time units) in expression of any gene – except for *ci* – and, in such delayed conditions, the mutant state characteristic to that gene knockout is generated. For low order delays, the system typically recovers and proceeds through the correct wild type development.

The Glass-type system with time separation, as a model of the segment polarity gene network, reflects the conclusion of von Dassow et al. that the topology of the network is more important than the fine-tuning of the kinetic parameters [11], since its results are robust for a large region of parameter (scaling factor, activation threshold) space. Due to its underlying Boolean structure, the model also intrinsically incorporates the recent finding of Ingolia that parameter sets need to satisfy certain constraints - that ensure the bistability of certain genes- to lead to correct solutions[25]. Taken together, the results of the synchronous [8], asynchronous [12] Boolean and hybrid models convincingly demonstrate the Boolean models' capability for effectively describing the basic structure and functioning of gene control networks when detailed kinetic information is unavailable.

References

[1] L. Giot, J. S. Bader, C. Brouwer, A. Chaudhuri, and B. Kuang et al. A protein interaction map of drosophila melanogaster. *Science*, 302:1727–1736, 2003.

[2] J. D. Han, N. Bertin, T. Hao, D. S. Goldberg, and G. F. Berriz et al. Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430:88–93, 2004.

[3] S. Li, C. M. Armstrong, N. Bertin, H. Ge, and et al. S. Milstein. A map of the interactome network of the metazoan c. elegans. *Science*, 303:540–543, 2004.

[4] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, and et al. Z. Bar-Joseph. Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 298:799–804, 2002.

[5] R. Albert. Scale-free networks in cell biology. *Journal of cell science*, 118:4947–4957, 2005.

[6] N. M. Luscombe, M. M. Babu, H. Y. Yu, M. Snyder, S. S. Teichmann, and M. Gerstein. Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, 431:308–312, 2004.

[7] L. Sánchez and D. Thieffry. A logical analysis of the *drosophila* gap-gene system. *J. Theor. Biol.*, 211:115–141, 2001.

[8] R. Albert and H. G. Othmer. The topology of the regulatory interactions predicts the expression pattern of the *drosophila* segment polarity genes. *J. Theor. Biol.*, 223:1–18, 2003.

[9] L. Mendoza, D. Thieffry, and E. R. Alvarez-Buylla. Genetic control of flower morphogenesis in *arabidopsis thaliana*: a logical analysis. *Bioinformatics*, 15:593–606, 1999.

[10] C. Espinosa-Soto, P Padilla-Longoria, and E. R. Alvarez-Buylla. A gene regulatory network model for cell-fate determination during arabidopsis thaliana flower development that is robust and recovers experimental gene expression profiles. *Plant Cell*, 16:2923–2939, 2004.

[11] G. von Dassow, E. Meir, E.M. Munro, and G.M. Odell. The segment polarity network is a robust developmental module. *Nature*, 406:188–192, 2000.

[12] M. Chaves, R. Albert, and E.D. Sontag. Robustness and fragility of boolean models for genetic regulatory networks. *J. Theor. Biol.*, 235:431–449, 2005.

[13] L. Glass. Classification of biological networks by their qualitative dynamics. *J. Theor. Biol.*, 54:85–107, 1975.

[14] R. Edwards and L. Glass. Combinatorial explosion in model gene networks. *Chaos*, 10:691–704, 2000.

[15] L. Glass and S.A. Kauffman. The logical analysis of continuous, nonlinear biochemical control networks. *J. Theor. Biol.*, 39:103–129, 1973.

[16] J. E. Hooper and M.P. Scott. The molecular genetic basis of positional information in insect segments. In W. Hennig, editor, *Early Embryonic Development of Animals*, pages 1–49. Springer, Berlin, 1992.

[17] A. Gallet, C. Angelats, S. Kerridge, and P.P. Thérond. Cubitus interruptus-independent transduction of the hedgehog signal in *drosophila*. *Development*, 127:5509–5522, 2000.

[18] T. Tabata, S. Eaton, and T.B. Kornberg. The *drosophila hedgehog* gene is expressed specifically in posterior compartment cells and is a target of *engrailed* regulation. *Genes & Dev.*, 6:2635–2645, 1992.

[19] C. Alexandre and J. P. Vincent. Requirements for transcriptional repression and activation by engrailed in *drosophila* embryos. *Development*, 130:729–739, 2003.

[20] D. Swantek and J. P. Gergen. Ftz modulates runt-dependent activation and repression of segment -polarity gene transcription. *Development*, 131:2281–2290, 2004.

[21] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and Distributed Computation, Numerical Method*. Prentice Hall, Englewood Cliffs, New Jersey, 1989.

[22] H. de Jong, J.L. Gouzé, C. Hernandez, M. Page, T. Sari, and J. Geiselmann. Qualitative simulation of genetic regulatory networks using piecewise linear models. *Bull. Math. Biol.*, 66:301–340, 2004.

[23] T. Gedeon. Attractors in continuous-time switching networks. *Communications on Pure and Applied Analysis*, 2:187–209, 2003.

[24] A. Mochizuki. An analytical study of the number of steady states in gene regulatory networks. *J. Theor. Biol.*, 236:291–310, 2005.

[25] N.T. Ingolia. Topology and robustness in the drosophila segment polarity network. *PLoS Biology*, 2:0805–0815, 2004.