Jérémie Bourdon
LINA CNRS UMR 6241, Université de Nantes and IRISA/INRIA Rennes Bretagne Atlantique

Damien Eveillard
LINA CNRS UMR 6241, Université de Nantes

A JOHN WILEY & SONS, INC., PUBLICATION

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.
Published simultaneously in Canada.

Library of Congress Cataloging-in-Publication Data:

Title, etc
Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

CONTENTS

CHAPTER 1

PROBABILISTIC APPROACHES FOR INVESTIGATING BIOLOGICAL NETWORKS

Last decade saw a significant increase of high throughput experiments. As a major achievement, these novel techniques replicate the molecular experiments, which opens perspectives of quantitative behavior investigations. For illustration, it is now possible to define the concentration for which a protein (i.e. transcription factor) may activate a given gene. This information used to be considered as a limitative factor for producing accurate dynamical models of large biological regulatory networks [5]. Today, one must take it into account for building large quantitative models. Furthermore, high through-put experiments describe, as well, macromolecular processes via their temporal properties. Thus, biological processes can be summarized by the evolutions of their biological compounds over times (*i.e.,* a succession of biological qualitative states or temporal patterns). Such experiments show temporal parameters that refine, in a natural manner, the qualitative models describing biological systems. However, these refinements, that present great biological interests, raise similarly several computational concerns. One is dealing with the complexity that arises from the large amount of experimental data. The challenge hence consists in trimming the experimental information at disposal for extracting the major driving compounds and their respective interactions within a network. Another is taking into account the quantitative impacts of these components on the biological dynamics (in [3],

title, edition. By author

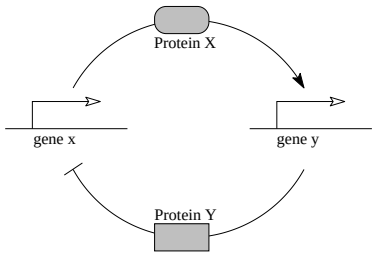


Figure 1.1 Representation of a gene regulatory network with two genes. Gene x activates the transcription of gene y via the production of protein X. Reversely, gene y represses the transcription of gene x via the production of protein Y.

authors propose a probabilistic model allowing to introduced a quantification under the hypothesis that the global behavior of a quantity is an accumulation of small variations). It implies dealing with different ranges of concentration variations for several compounds. Such integrations rise the opportunity to build predicting models that reproduce replicated experimental results. A third issue is integrating temporal behaviors in such complex systems (a study of the temporal effects in *E. Coli* carbon starvation system can be found in [1]). In this case, the complexity lies in introducing partial informations. Indeed, a temporal behavior of a given compound, like a gene, is often known, whereas others remains not well understood.

To sum up, current computational challenges are (i) analyzing large amount of data and their inherent complexity, introducing both (ii) quantitative knowledge and (iii) temporal properties into models of biological regulatory networks. Among the computational biology techniques, probabilistic approaches appear as an accurate consensus that deal with those features. This chapter proposes a short overview of their applications for investigating biological regulatory networks. We will first present the theoretical framework needed for such particular biological systems (Sec. 1.1). It emphasizes a qualitative modeling that comes from both empirical and experimental knowledge. Second, we will show (Sec. 1.2) an overview of the analyses that can be performed on such a kind of model.

1.1 PROBABILISTIC MODELS FOR BIOLOGICAL NETWORKS

Several probabilistic approaches exist, and they are not all accurate for modeling dynamical biological systems. Feedback loops are the most common control process of natural systems, especially for gene regulatory networks (see Fig. 1.1). Taking such loops into account (i.e. positive or negative) is therefore the major criterion for choosing one probabilistic approach among others.

Because it is based on Boolean Networks, Probabilistic Boolean Networks (PBN)[19] is a probabilistic framework that deals with feedback loops. This modeling gives two great advantages. First, it allows a qualitative analysis [4, 5] by investigating the qualitative properties of the "boolean core". Second, it proposes a quantitative

analysis when focusing on the probabilities. Combining probabilities and BN logical informations gives thus insights about both temporal behaviors and predictions of biological compounds quantities, which comes up to recent biological expectations. In this section, we focus on Boolean Networks.

1.1.1 Boolean Networks

The Boolean Networks represent the qualitative core of the PBN. Their interpretation constitutes a key step for a complete understanding of their probabilistic extensions. Introduced by Stuart Kauffman and co-workers[8, 13], the Boolean Networks quickly raise a strong interest from both physical and biological fields. Their applications in computational biology resume the genes by switchs: the gene activity can be either ON or OFF. This assumption comes from a simplification of the step function that represents the activation of a gene by another. Because these genes interact on each other, their interactions build a network, in which the evolution of a given gene activity depends on the activities of other genes (see [5] for review). Following this assumption, a Boolean Network can be seen as a vector of boolean functions, such as:

Definition 1 *A Boolean Network $B = (V, F)$ is a pair where*

- $V = \{x_1, \ldots, x_n\}$ *is a set of boolean variables (i.e. genes),* $\forall i, x_i \in \{0, 1\}$.
- $F = \{f_1, \ldots, f_n\}$ *is a set of boolean functions,* $\forall i, f_i : \{0,1\}^n \rightarrow \{0,1\}$. *Here, f_i describes the evolution of gene i.*

Such networks allow the dynamical description of given phenomena. Formally, if $X(t) = (x_1(t), \ldots, x_n(t))$ represents the value of all variables at time t, then $X(t+1) = (x_1(t+1), \ldots, x_n(t+1))$, where $\forall i$, $x_i(t+1) = f_i(x_1(t), \ldots, x_n(t))$ is the value of all variables after one iteration of the Boolean Network. Note that for n genes, the corresponding number of Boolean Networks is $(2^n)^{(2^n)}$. Among them, few are accurate with biological knowledge. Their identifications is therefore a major issue of Boolean Networks Theory.

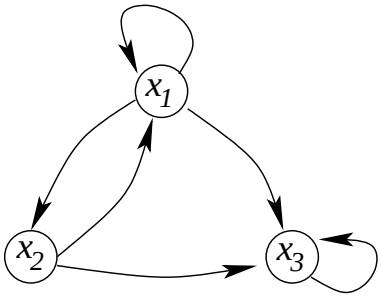
Dependency graph. We are interested in trimming the number of Boolean Networks of interest. In general cases, f_i depends on a subset of the boolean variables only. A boolean variable x_j is so-called *ficticius* for f_i if, and only if, for all $(x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n) \in \{0,1\}^{n-1}$,

$$f_i(x_1, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_n) = f_i(x_1, \ldots, x_{j-1}, 1, x_{j+1}, \ldots, x_n).$$

It shows that some genes are not useful for predicting the behaviors of the system: f_i does not depend on variable x_j, or in other words $\partial f_i(x_1, \ldots, x_n)/\partial x_j = 0$. All other genes are so-called *essential* because they impact the dynamical description. Thus, knowing the values of f_i for all possible affectations of essential variables is sufficient. Furthermore, it is informative to draw the boolean variables dependencies using a directed graph.

Definition 2 *Let $B = (V, F)$ a Boolean Network. The dependency graph $G = (V, E)$ of B is defined by $(j, i) \in E$ if, and only if, x_j is an essential variable for f_i.*

The following figure illustrates an example of dependency graph. Herein, x_1 and x_2 are essential for f_1; x_1 is essential for f_2; x_1, x_2 and x_3 are essential for f_3.



Representations of boolean functions. The Boolean Networks can be complex and one of the major concerns remains the storage of the boolean functions. Indeed, it is necessary to store one or more boolean function per gene. Several ways define efficiently and unambiguously a boolean function.

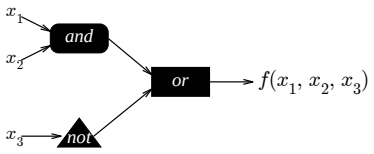
Truth table The simplest way to define a boolean function consists in providing its truth table.

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

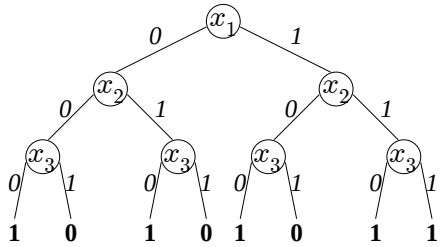
However, based on the function values for all its possible entries, the size of such a table is exponential.

Logic expressions Any boolean function can be expressed via simple boolean operations $\{\wedge$(logical and), $\vee$(logical or)$and\neg$(logical not)$\}$. Combining these logical operators is equivalent to design a logic circuit that mimics a given boolean function. It compacts the expression of the boolean function. Like this, the previous

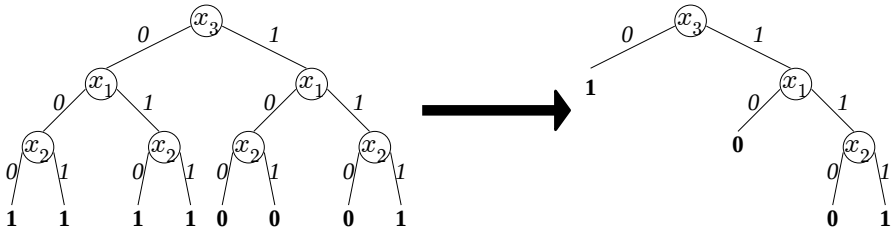
truth table corresponds to $f(x_1, x_2, x_3) = (x_1 \wedge x_2) \vee \neg x_3$, or this logical circuit:



Binary Decision trees Another expression of the truth table relies on a branching process. As illustration, based on the previous truth table, the first half of the table corresponds to all entries when the first variable is false. By using this consideration recursively, one defines a binary decision tree that matches the truth table.



Although the size of such a graph is still exponential, it allows to think about some improvements. Like this, it emphasizes that some part of the decision tree are not that informative, which it is not obvious when reading the truth table only. Indeed, if $x_1 = x_2 = 1$ is true then for any choice of x_3, $f(x_1, x_2, x_3) = 1$. As a consequence, the last part of the tree can be pruned and replaced by a leaf that has value 1. The variable ordering is therefore one of the key features. For illustration, the decision tree can be simplified by considering the ordering (x_3, x_1, x_2) instead of (x_1, x_2, x_3). In this case, half of the binary decision tree is pruned.

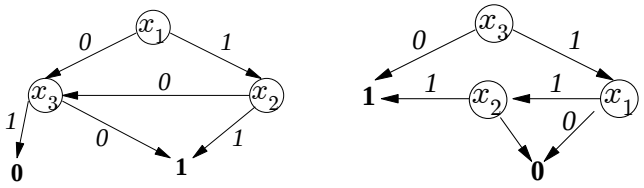


This tree manipulation represents a natural way to emphasize an information of interest. Lee [14] extends this approach and propose a Binary Decision Diagram (BDD). It is a Directed Acyclic Graph obtained when applying two simplification rules to the decision tree:

1. Merging any isomorphic subgraphs.

2. Eliminating any node whose two childs are isomorphic.

When applied on the above Decision tree (respectively for the orders (x_1, x_2, x_3) and (x_3, x_1, x_2)), these two rules give the following binary decision diagrams.

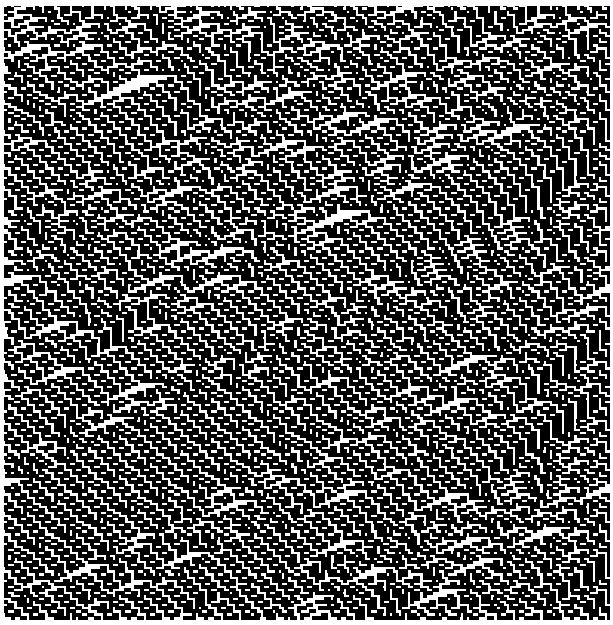


These diagrams show different roots; respectively x_1 for the order (x_1, x_2, x_3) and x_3 for (x_3, x_1, x_2), but both show similar decision patterns and truth tables, as shown above. As a concrete application in the computational biology field, A. Naldi and co-workers [17] propose a similar approach that successfully investigates gene regulatory network boolean models.

Examples of Boolean Networks. A literature review shows several modelings of biological systems using Boolean Networks, and moreover plenty modeling approaches that derive from Boolean Networks. We propose to present herein three of them.

Cellular automata As a classical Boolean Network extension, a cellular automaton [10] is a modeling approach that focuses on the notion of variable locality. Indeed, the cellular automata consider a unique boolean function $f(z_1, \ldots, z_k)$ in k variables and a set of k integers $\mathcal{N} = \{n_1, \ldots, n_k\} \subset \mathbb{Z}$ that corresponds to the neighborhood to be considered. The associated Boolean Network is then defined by setting $f_i(x_1, \ldots, x_n) = f(x_{i+n_1}, \ldots, x_{i+n_k})$. Notice here that the indices belong to the torus $\{1, \ldots, n\}$ (*i.e.,* all operations are assumed to be modulo n). where $x[n]$ denotes the remainder of the ordinary euclidean division of x by n. The following figure shows a trajectory of the cellular automaton corresponding to $\mathcal{N} = \{-1, 0, 1\}$ and $f(x_1, x_2, x_3) = (\neg x_3 \wedge x_2) \vee (x_3 \wedge \neg x_1) \vee (x_3 \wedge x_1 \wedge \neg x_2)$. More precisely, the first line represents the initial values of all variables (pixels in white if 0 or in black

if 1). Each line below is computed from the previous one and the Boolean Network.



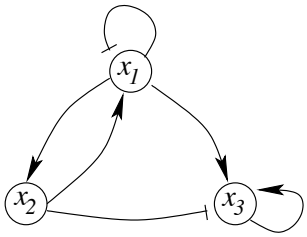
As a major feature, the cellular automaton theory achieves the identification of periodic patterns that can be derived from the evolution of the automaton. For illustration, the above picture shows periodic patterns, like the tiny white triangles that are repeated on the diagonals of the figure. They characterize one of the specific properties of the boolean functions. They might be automatically extracted from the simulation traces via standard pattern finding approaches.

Discrete networks The major biological assumption depicted within the Boolean Networks, remains the discrete abstraction of the gene activity. However, in many cases assuming a gene as a simple interruptor is not appropriate. Its has been noticed that genes may have different behaviors depending on their activation levels [6]. Other genes may present different promotors that respond differentially in function of the transcription factor activity level [18]. Therefore, a single boolean domain for each variable is no longer sufficient. A natural way to deal with this consists in introducing thresholds for modeling gene activity, in order to differentiate several effects of the gene activity. Following this assumption, R. Thomas and co-workers [26, 24] propose a model of discrete networks where the variables are discretized. D. Thieffry and co-workers extend this idea [4, 25, 23]. It allows them to analyze the qualitative properties of the discrete models, which bridges the gap between continuous models (*i.e.,* Ordinary Differential Equation based models) and Boolean Networks. The implementation of such an approach shows fine qualitative results [16], but reversely produces discrete exponential graphs. Hence, they are often too large for allowing a probabilistic extension on the discrete networks.

Influence graphs Following similar gene regulatory network modeling motivations, A. Siegel and co-workers [22] propose an another extension with a restricted type of boolean functions. Indeed, they consider a function $f_i(x_1, \ldots, x_n) = y_1 \vee \cdots \vee y_n$ where $y_k = 0$ if variable x_k is ficticious for f_i; and $y_k = x_k$ or $y_k = \neg x_k$ where x_k is essential for f_i. The choice between these two cases relies on the fact that x_k is an activator or a repressor for x_i. As a direct consequence, each essential variable appears only once in the boolean function expression, either as an activator or an inhibitor. From this compact expression of the gene regulatory network, they define a so-called *Generalized Dependency Graph* $\widetilde{G} = (V, E)$ of the Boolean Network $B = (V, F)$, where there exists an edge $(i, j, s) \in E \subset V \times V \times \{-, +\}$ if, and only if, x_j is an essential variable for f_i and $s = \text{sign}\ (\partial f_i(x_1, \ldots, x_n)/\partial x_j)$. For illustration, when considering three boolean functions that represent a given biological knowledge:

$$\begin{aligned} f_1(x_1, x_2, x_3) &= \neg x_1 \vee x_2 \\ f_2(x_1, x_2, x_3) &= x_1 \\ f_3(x_1, x_2, x_3) &= x_1 \vee \neg x_2 \vee x_3 \end{aligned}$$

We can picture these constraints by the following Generalized Dependency Graph:



This graph represents an elegant summary of the logical (i.e. qualitative) properties that emerge from the set of biological constraints. Note that these boolean constraints might be automatically investigated. In particular, P. Veber and co-workers [27] propose an *in silico* protocol that checks the consistencies of these constraints with experimental knowledges at disposal. It represents an elegant validation of the qualitative properties of the *"boolean core"* before further probabilistic extensions.

1.1.2 Probabilistic Boolean Networks: a natural extension

Boolean Networks do not always reflect the correct behaviors of complex biological models. In fact, at some point, quantitative models need flexibility for taking into account the inherent complexity of gene interactions (i.e. non linearity due to post-transcriptional regulations) or to deal with incomplete data. In this purpose, Shmulevich and co-workers [19] introduced Probabilistic Boolean Networks (PBN), that is a probabilistic extension of Boolean Networks.

Definition 3 *A Probabilistic Boolean Network $B = (V, \mathcal{F})$ is a pair where*

- $V = \{x_1, \ldots, x_n\}$ *is a set of boolean variables (i.e. genes),* $\forall i, x_i \in \{0, 1\}$*;*
- $\mathcal{F} = \{F_1, \ldots, F_n\}$ *is a set where* $F_i = \{(f_i^{(1)}, p_i^{(1)}), \ldots, (f_i^{(l_i)}, p_i^{(l_i)})\}$ *is a set of pairs composed by a boolean function and a probability. For all i, one has* $\sum_{k \in \{1,\ldots,l_i\}} p_i^{(1)} = 1$*. Here, the evolution of gene i is predicted by $f_i^{(k)}$ with probability $p_i^{(k)}$.*

The dynamics of the biological system are now described using boolean random variables $(X_1(t), \ldots, X_n(t))$, that satisfy:

$$\forall i, \forall k \in \{1, \ldots, l_i\}, \text{Prob}\{X_i(t+1) = f_i^{(k)}(X_1(t), \ldots, X_n(t))\} = p_i^{(k)}.$$

From the simulation viewpoint, it opens other perspectives. If $(x_1, \ldots, x_n)$ is the current boolean affectation of all gene activities, the activity of gene i becomes $f_i^{(k)}(x_1, \ldots, x_n)$ with probability $p_i^{(k)}$. Like this, the probabilities add an uncertainty feature to the model. Intuitively, one disposes of several predictors for a gene activity. One can trust a predictor with a given probability.

Note that PBN can be decomposed as a finite set of $\prod_{i=1}^{n} |F_i|$ constituent Boolean Networks with some transitions probabilities between them that are determined by the predictor probabilities.

1.1.3 Inferring probabilistic models from experiments

The concern is to fit the previous probabilistic framework with biological knowledges at disposal. Several approaches propose to built them from experiments using an automatic manner[19, 20, 21]. Mainly, experiments correspond to long time courses $C = ((x_1(1), \ldots, x_n(1)), \ldots, (x_1(T), \ldots, x_n(T)))$ involving measures of n genes at T different time steps. The goal is to build a PBN that reproduces C as a trajectory with a high probability. Notice that if the time course arises from a single Boolean Network, classical methods such as Viterbi or Baum-Welch algorithms can be adapted from the inference of Hidden Markov Model herein for contructing the most probable Boolean Network. We observe that a PBN can be seen as a composition of a finite number of constituent Boolean Networks. In this context, perturbation probabilities allow to swap from a Boolean Network to another. Based on this observation, the inference of a proper PBN from long time courses consists in three distinct steps:

1. First separate the time course into subsequences arising from the same constituent Boolean Network;
2. Infer separately every constituent Boolean Networks;
3. Retrieve the swapping probabilities between constituent Boolean Networks and construct the PBN.

Notice that the first step is crucial. The choice of the optimization methods is also a major factor to ensure a convergence to the right PBN.

1.2 INTERPRETATION AND QUANTITATIVE ANALYSIS OF PROBABILISTIC MODELS

Previous analyses of the boolean core allow to investigate the qualitative properties of the biological regulatory networks. Others approaches exist for investigating both temporal and quantitative properties that emerge from models probabilistically extended.

1.2.1 Dynamical analysis and temporal properties

One of the major biological expectations when studying regulatory networks, is to extract general properties from the evolution of gene activities. This evolution is inherently encoded by the network. It can be represented by a Dynamical Graph (or state space) defined as follows:

Definition 4 *Let $B = (V, F)$ be a Boolean Network and $n = |V|$. The dynamical graph $\mathcal{G} = (\{0,1\}^n, E)$ associated to B is a directed graph possessing an edge from $(x_1, \ldots, x_n)$ to $(x'_1, \ldots, x'_n)$ if this edge defines a possible update (several update strategies are described in the sequel).*

The dynamical graph may show attractors that represent key features of the system dynamic. It has been shown that phenotypes are very often associated to these attractors [11, 12]. Therefore, studying these properties of the dynamical graph presents great interest in a biological context. It explains why several approaches have been proposed. They focus on distinct viewpoints: simulation of the steady state distribution, algorithms from the graph theory to study the topological aspects of graphs, model checking approaches... Since they introduce time, all these approaches are sensitive to the update of the biological system. For the same network, several update strategies were proposed. It results in distinct dynamical graphs.

1.2.1.1 Synchronous update It is the most natural update. It corresponds to a succession of observations of the gene activities at some fixed time. The synchronous update strategy assumes that all genes are updated at the same time. Like this, if $(x_1(t), \ldots, x_n(t))$ is the boolean affectation of all gene activities at time t, $(x_1(t+1), \ldots, x_n(t+1))$, where for all i, $x_i(t+1) = f_i(x_1(t), \ldots, x_n(t))$, is the boolean affectation of all gene activities at time $t+1$. For illustration, the Figure 1.2 draws the synchronous dynamical graph associated to the Boolean Network

$$\begin{aligned}
f_1(x_1, x_2, x_3, x_4) &= x_1 \wedge x_2 \vee \neg x_2 \wedge x_3,\\
f_2(x_1, x_2, x_3, x_4) &= x_1 \wedge x_3 \wedge x_4 \vee \neg x_1 \wedge x_2,\\
f_3(x_1, x_2, x_3, x_4) &= x_1 \wedge \neg x_2 \wedge x_4 \vee \neg x_2 \wedge \neg x_4,\\
f_4(x_1, x_2, x_3, x_4) &= x_1 \vee x_2 \wedge x_3 \vee \neg x_3 \wedge x_4.
\end{aligned}$$

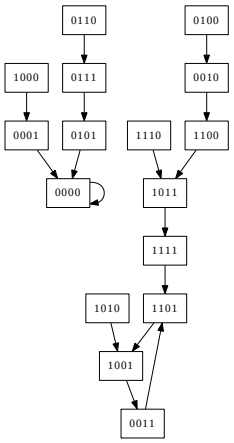


Figure 1.2 Dynamical graph with a synchronous update strategy

1.2.1.2 Asynchronous update One might observe that it is rare that two independent genes change their activity level together at the same time. Regarding the dynamical evolution of discretizations of continuous variables, this observation is inconsistent with the previous synchronous update strategy. We must suppose herein that only one gene can change at a given time. Like this, if $(x_1(t), \ldots, x_n(t))$ is the boolean affectation of all gene activities at time t, there exists at most n possible affectations $(x_1(t+1), \ldots, x_n(t+1))$ where $x_i(t+1) = f_i(x_1(t), \ldots, x_n(t))$ if $i = j$ and $x_i(t+1) = x_i(t)$ otherwise, for all $j \in \{1, \ldots n\}$. For illustration, Figure 1.3 pictures the dynamical graph of the Boolean Network above, with an asynchronous strategy.

1.2.1.3 Mixed updates with priorities Biological systems often need plasticity in their update strategies. Indeed, some genes are co-regulated , whereas others are independent. Thus, the asynchronous assumption is not fulfilled due to regulation specificities or an incomplete knowledge. In this context, Naldi and co-workers [16] proposed a mixed strategy that combines synchronous updates for genes having a similar regulation speed and asynchronous updates for others. For that, they define a synchronization partition of genes. Let $P = \{I_1, \ldots, I_m\}$, be a disjoint partition of $\{1, \ldots, n\}$ composed by non empty sets (*i.e.,* $\forall u, I_u \neq \emptyset$, $\bigcup_{u=1}^{m} I_u = \{1, \ldots, n\}$ and $\forall u \neq v,\ I_u \cap I_v = \emptyset$). This partition describes synchronizations between genes. The two extremal cases corresponds to $P = \{\{1, \ldots, n\}\}$ for synchronous updates and $P = \{\{1\}, \ldots, \{n\}\}$ for asynchronous updates. Then, if $(x_1(t), \ldots, x_n(t))$ is the boolean affectation of all gene activities at time t, there exists at most m possible affectations $(x_1(t+1), \ldots, x_n(t+1))$ where for all $u \in \{1, \ldots, m\}$, $x_i(t+1) = f_i(x_1(t), \ldots, x_n(t))$ if $i \in I_u$ and $x_i(t+1) = x_i(t)$ otherwise. For illustration, Figure 1.4 represents the dynamical graph of the previous Boolean Network with a mixed update strategy with synchronization partition $P = \{\{1, 2\}, \{3\}, \{4\}\}$.

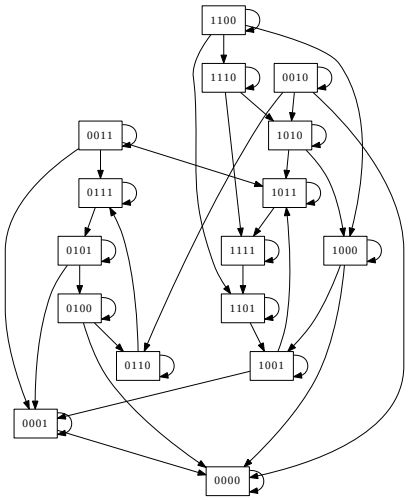


Figure 1.3 Dynamical graph with an asynchronous update strategy

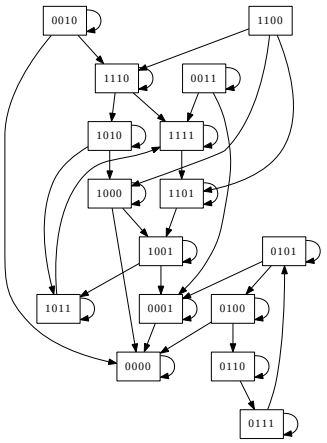


Figure 1.4 Dynamical graph with a mixed update strategy with synchronization partition $P = \{\{1, 2\}, \{3\}, \{4\}\}$

1.2.2 Impact of update strategies for analyzing Probabilistic Boolean Networks

In a probabilistic context, edges of a dynamical graph are endowed with a transition probability. Consequently, the dynamical graph becomes a Markov Chain. Naturally, update strategies that impact the Boolean Network interpretation, play a major role for investigating probabilistic Boolean Networks too. We propose to illustrate this point by showing how mixed updates are extended in the probabilistic context of

PBN. Here, the dynamical graphs become Markov chains with 2^n states, whose transition matrix $T = (p_{i\to j})_{i,j\in(\{0,1\}^n)^2}$ is defined as follow.

Let $P = \{I_1, \ldots, I_m\}$ be a synchronization partition of $\{1, \ldots, n\}$. For all $u \in \{1, \ldots, m\}$, let $I_u = \{s_1, \ldots, s_{p_u}\}$ and define the sets

- $H_u = \prod_{s\in I_u}\{1, \ldots, l_s\}$, where l_s is the number of statistical predictors of gene s;
- $U_{i,j,u} = \{(k_{s_1}, \ldots, k_{s_{p_u}}) \in H_u, (x_1, \ldots, x_n) = i, x'_s = f_s^{(k_s)}(x_s)$ if $s \in I_u$ and $x'_s = x_s$ otherwise and $(x'_1, \ldots, x'_n) = j\}$.

Then, if one has

$$p_{i\to j} = \sum_{u=1}^{m} \sum_{\substack{(k_{s_1}, \ldots, k_{s_{p_u}}) \in U_{i,j,u},\\ I_u = \{s_1, \ldots, s_{p_u}\}}} \prod_{s\in I_u} p_s^{(k_s)}.$$

This framework achieves a probabilized dynamical graph for all update strategies applied on a PBN. For illustration, let consider PBN defined by:

$$\begin{array}{lcll}
f_1^{(1)}(x_1, x_2, x_3, x_4) & = & x_1 \wedge x_2 \vee \neg x_2 \wedge x_3, & p_1^{(1)} = 0.3 \\
f_1^{(2)}(x_1, x_2, x_3, x_4) & = & \neg x_1 \wedge x_2 \vee x_3 \wedge x_4, & p_1^{(2)} = 0.7 \\
f_2^{(1)}(x_1, x_2, x_3, x_4) & = & x_1 \wedge x_3 \wedge x_4 \vee \neg x_1 \wedge x_2, & p_2^{(1)} = 1 \\
f_3^{(1)}(x_1, x_2, x_3, x_4) & = & x_1 \wedge \neg x_2 \wedge x_4 \vee \neg x_2 \wedge \neg x_4, & p_3^{(1)} = 1 \\
f_4^{(1)}(x_1, x_2, x_3, x_4) & = & x_1 \vee x_2 \wedge x_3 \vee \neg x_3 \wedge x_4, & p_4^{(1)} = 1.
\end{array}$$

This PBN is a generalization of the BN previously presented, the only change being on the rule for gene 1.

Figure 1.5, 1.6 and 1.7 represent the dynamical graphs with probabilities on edges for distinct strategy updates; respectively synchronous, asynchronous and mixed strategy with $P = \{\{1, 2\}, \{3\}, \{4\}\}$.

1.2.3 Simulations of a Probabilistic Boolean Network

Previous approaches emphasize the qualitative properties of the Probabilistic Boolean Networks. However, biologists might be interested by quantifying each biological compounds that interact within the biological network. Simulations represent a natural way to predict the biological compounds quantities. Note here that such a quantitative information is in accordance with qualitative results previously shown. The quantitative information results from the transitions taken more or less, in a manner that is conformed with probabilities on the network transitions. In other words, qualitative properties indicate the potential qualitative transitions, whereas quantitative features represent the integration of all qualitative transitions. The Monte-Carlo approach achieves such an integration by computing numerical values, given a probability distribution. In the PBN context, probabilities are related to the interactions. When one stay at one state in the graph, Monte-Carlo algorithm indicates

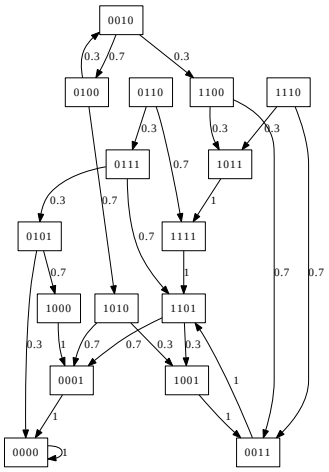


Figure 1.5 Dynamical graph of a PBN with a synchronous strategy

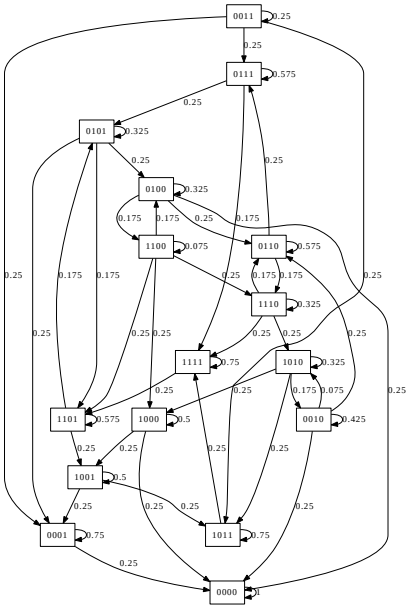


Figure 1.6 Dynamical graph of a PBN with an asynchronous strategy

what is the transition to take, in accordance with the probabilities associated with the transitions that go out from the given state (see Figure 1.8 for illustration). Following a Bernoulli law and a significant number of random walk through the graph, one can estimate the distribution of the biological quantities. In this context, we describe here a simulation based on a Markov Chain Monte-Carlo approach that estimates the

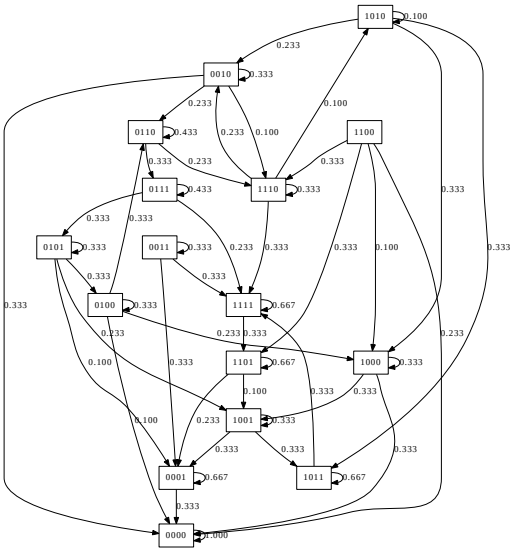


Figure 1.7 Dynamical graph of a PBN with a mixed Strategy considering $P = \{\{1,2\},\{3\},\{4\}\}$.

equilibrium distribution. Several algorithms implement distinct approaches, among which Metropolis-Hasting algorithm and Gibbs sampling are the most applied in computational biology (see [2] for details).

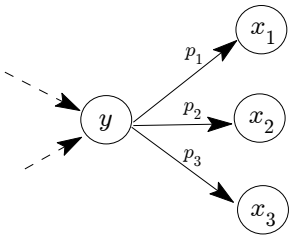


Figure 1.8 Illustration of the Monte-Carlo algorithms. The algorithm produces a random walk in the network in accordance with the probability distribution. Here, when one is in the state y, the choice to take the path through x_1, x_2 or x_3 is made in accordance with probabilities p_1, p_2 and p_3.

The major issue of Monte Carlo approaches is to determine how many steps are necessary for an accurate estimation of the equilibrium. Moreover, biologists might be interested by the evolution of the quantities. It implies adding an estimation of time between two given quantities. In this purpose, Gillespie algorithm [7] is a refinement of the Monte Carlo approach. It uses a complementary information: the production rate τ for each interaction. The algorithm simulates the evolution

of the biological compound over time by determining what kind, and when, the next interaction will occur. Like this, the simulation shows the result of a random walk in the discrete network for given initial conditions: the quantity of biological components in presence.

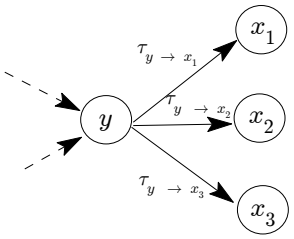


Figure 1.9 Illustration of the Gillespie algorithms. The algorithm produces a random walk in the network in accordance with the production rates. Here, when one is in the state y, the choice to take the path through x_1, x_2 or x_3 is made in accordance with probabilities estimated from $\tau_{y \to x_1}$, $\tau_{y \to x_2}$, and $\tau_{y \to x_3}$.

Gillespie algorithm applied on Probabilistic Boolean Networks For a given quantity of all biological compounds of the system, and some production rates $\tau_{i \to j}$ from state i to j for all edges, the Gillepsie algorithm consists in a repetition of four basic steps. First suppose that the initial state is i.

1. Let $\tau_{tot} = \sum_j \tau_{i \to j}$.
2. Choose a random number T following an exponential distribution with parameter τ_{tot}. Here, T is the total duration ellapsed in state i. Increase the total time by T.
3. Choose randomly the next state, each state j is reached with probability $\tau_{i \to j} / \tau_{tot}$.
4. Update the production rates.

These simulations are particularly appropriate for estimating quantities of biological compounds that interact on large biological networks. The efficiency of the quantitative prediction can then be estimated using multi-regression approaches. Note herein, that automatic probabilistic verifications can be performed on smaller networks [9]. It emphasizes the impact of specific probabilities on the overall quantitative behaviors of the system. From the biological viewpoint, it indicates the genes, or other biological compounds, that can be tuned for a better fitting of the model with experimental data; or corner-stone genes that might impact the overall behavior when modified (i.e. mutation or environmental condition modifications). The probabilistic

model-checkers hence appear as a natural complement to the automatic qualitative verification techniques of the Boolean core, as previously mentioned in Sec. 1.1.1.

1.3 CONCLUSION

In this chapter, we summarized the essential features of the Probabilistic Boolean Networks. They represent a general probabilistic model that possesses plenty of applications in the context of biological networks, when dedicated extensions are proposed. Notice that plenty other probabilistic models not shown herein exist. Bayesian Network is one of them that deals with biological informations. It is a probabilistic graph model that represents the biological compound interactions via a directed acyclic graph. As itself, it is not able to take into account the feedback loops. For taking them into account, one introduces Dynamical Bayesian Networks. They consist in a repetition of an elementary Bayesian Network, as previously defined, that are linked together in order to abstract the dynamical effect, which includes the feedback loops. For further reading about this method as an extension of this chapter, we recommend the study [15] that compares the Probabilistic Boolean Networks and the Dynamical Bayesian Networks in a gene regulatory context.

Acknowledgments

The authors of this chapter would like to thank Mathieu Giraud and Pierre Peterlongo for their precious comments.

REFERENCES

1. J. Ahmad, J. Bourdon, D. Eveillard, J. Fromentin, O. Roux, and C. Sinoquet. Temporal constraints of a gene regulatory network: Refining a qualitative simulation. *BioSystems*, May 2009.
2. C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan. An introduction to mcmc for machine learning. *Machine Learning*, 50:5–43, 2003.
3. J. Bourdon, D. Eveillard, S. Gabillard, and T. Merle. Using a probabilistic approach for integrating heterogeneous biological knowledges. *Proc. of RIAMS 2007, Lyon*, page 8p., 2007.
4. C. Chaouiya, H. de Jong, and D. Thieffry. Dynamical modeling of biological regulatory networks. *Biosystems*, 84(2):77–80, May 2006.
5. H. de Jong. Modeling and simulation of genetic regulatory systems: a literature review. *J Comput Biol*, 9(1):67–103, 2002.
6. A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry. Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. *Bioinformatics*, 22(14):e124–31, Jul 2006.
7. D.T. Gillespie. Stochastic simulations of coupled chemical reactions. *J. Phys. Chem.*, 81(25):2340–2361, 1977.

8. K. Glass and S.A. Kauffman. The logical analysis of continuous, non-linear biochemical control networks. *J. Theor. Biol.*, 39:103–129, 1973.

9. J. Heath, M. Kwiatkowska, G. Norman, D. Parker, and O. Tymchyshyn. Probabilistic model checking of complex biological pathways. *Theor. Comput. Sci.*, 391(3):239–257, 2008.

10. J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.

11. S. Huang. Gene expression profiling, genetic networks, and cellular states: an integrating concept for tumorigenesis and drug discovery. *J Mol Med*, 77(6):469–80, Jun 1999.

12. S. Huang. Genomics, complexity and drug discovery: insights from boolean network models of cellular regulation. *Pharmacogenomics*, 2(3):203–22, Aug 2001.

13. S.A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J Theor Biol*, 22(3):437–67, Mar 1969.

14. C.Y. Lee. Representation of switching circuits by binary-decision programs. *Bell Systems Technical Journal*, 38:985—999, 1959.

15. P. Li, C. Zhang, E.J. Perkins, P. Gong, and Y. Deng. Comparison of probabilistic boolean network and dynamic bayesian network approaches for inferring gene regulatory networks. *BMC Bioinformatics*, 8 Suppl 7:S13, 2007.

16. A. Naldi, D. Berenguier, A. Fauré, F. Lopez, D. Thieffry, and C. Chaouiya. Logical modelling of regulatory networks with ginsim 2.3. *Biosystems*, May 2009.

17. A. Naldi, D. Thieffry, and C. Chaouiya. Decision diagrams for the representation and analysis of logical models of genetic networks. In *Computational Methods in Systems Biology (CMSB'07)*, volume volume LNCS/LNBI 4695, pages 233–247, 2007.

18. D. Ropers, H. de Jong, M. Page, D. Schneider, and J. Geiselmann. Qualitative simulation of the carbon starvation response in escherichia coli. *Biosystems*, 84(2):124–52, May 2006.

19. I. Shmulevich, E.R. Dougherty, S. Kim, and W. Zhang. Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, 18(2):261–74, Feb 2002.

20. I. Shmulevich, E.R. Dougherty, and W. Zhang. Gene perturbation and intervention in probabilistic boolean networks. *Bioinformatics*, 18(10):1319–31, Oct 2002.

21. I. Shmulevich and W. Zhang. Binary analysis and optimization-based normalization of gene expression data. *Bioinformatics*, 18(4):555–65, Apr 2002.

22. A. Siegel, O. Radulescu, M. Le Borgne, P. Veber, J. Ouy, and S. Lagarrigue. Qualitative analysis of the relation between dna microarray data and behavioral models of regulation networks. *Biosystems*, 84(2):153–74, May 2006.

23. D. Thieffry. Dynamical roles of biological regulatory circuits. *Brief Bioinform*, 8(4):220–5, Jul 2007.

24. D. Thieffry and R. Thomas. Dynamical behaviour of biological regulatory networks–ii. immunity control in bacteriophage lambda. *Bull Math Biol*, 57(2):277–97, Mar 1995.

25. D. Thieffry and R. Thomas. Qualitative analysis of gene networks. *Pac Symp Biocomput*, pages 77–88, 1998.

26. R. Thomas, D. Thieffry, and M. Kaufman. 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):247–76, Mar 1995.

27. P. Veber, C. Guziolowski, M. Le Borgne, O. Radulescu, and A. Siegel. Inferring the role of transcription factors in regulatory networks. *BMC Bioinformatics*, 9:228, 2008.