Qualitative modeling of biological networks for therapeutic innovation

Arnaud Poret

Qualitative modeling of biological networks for therapeutic innovation

Modélisation qualitative des réseaux biologiques pour l'innovation thérapeutique

Arnaud Poret

Thèse de doctorat en Biologie Computationnelle
Soutenue le 1er Juillet 2015 à Lyon
Université Claude Bernard Lyon 1
École Doctorale E2M2 : Évolution Écosystèmes Microbiologie Modélisation

---

Directeur

Jean-Pierre Boissel . . . . . Professeur à l'Université Claude Bernard Lyon 1

Rapporteurs

Dominique Barbolosi . . . . . Professeur à l'Université Aix-Marseille
Sylvain Sené . . . . . Professeur à l'Université Aix-Marseille

Jury

Dominique Barbolosi . . . . . Professeur à l'Université Aix-Marseille
Jean Clairambault . . . . . Directeur de recherche à l'INRIA
Jacques Demongeot . . . . . Professeur à l'Université Joseph Fourier
Sylvain Sené . . . . . Professeur à l'Université Aix-Marseille
Denis Thieffry . . . . . Professeur à l'École Normale Supérieure
Michel Tod . . . . . Professeur à l'Université Claude Bernard Lyon 1

I dedicate this thesis to my parents Danièle and Joël Toussaint.

Je dédie cette thèse à mes parents Danièle et Joël Toussaint.

Acknowledgments

I thank my parents **Danièle Toussaint** and **Joël Toussaint** for their unfailing support and unwavering trust.

I thank my thesis director **Jean-Pierre Boissel** for his benevolence and valuable advices.

I thank **Dominique Barbolosi** and **Sylvain Sené** for having agreed to be reporter.

I thank **Dominique Barbolosi**, **Jean Clairambault**, **Jacques Demongeot**, **Sylvain Sené**, **Denis Thieffry** and **Michel Tod** for having agreed to be jury member.

I thank **François-Henri Boissel** for his welcome within the company Novadiscovery where I gained a precious experience.

I thank **François Gueyffier** for his welcome within the research team *Evaluation and Modeling with the Therapeutic Effects* of the UMR CNRS 5558 – Laboratory of Biometry and Evolutionary Biology.

I thank my colleagues and friends **Oscar Ayoun**, **Claudio Monteiro-Sousa** and **Julien Wan** for their cheerfulness and wittiness.

Remerciements

Je remercie mes parents **Danièle Toussaint** et **Joël Toussaint** pour leur soutien sans faille et leur confiance indéfectible.

Je remercie mon directeur de thèse **Jean-Pierre Boissel** pour sa bienveillance et ses précieux conseils.

Je remercie **Dominique Barbolosi** et **Sylvain Sené** pour avoir accepté d'être rapporteur.

Je remercie **Dominique Barbolosi**, **Jean Clairambault**, **Jacques Demongeot**, **Sylvain Sené**, **Denis Thieffry** et **Michel Tod** pour avoir accepté d'être membre du jury.

Je remercie **François-Henri Boissel** pour son accueil au sein de la société Novadiscovery où j'ai pu acquérir une précieuse expérience.

Je remercie **François Gueyffier** pour son accueil au sein de l'équipe de recherche *Évaluation et Modélisation des Effets Thérapeutiques* de l'UMR CNRS 5558 – Laboratoire de Biométrie et Biologie Évolutive.

Je remercie mes collègues et amis **Oscar Ayoun**, **Claudio Monteiro-Sousa** et **Julien Wan** pour leur bonne humeur et leur humour.

Abstract

This thesis is devoted to the qualitative modeling of biological networks for therapeutic innovation. It investigates how to use the Boolean network formalism, and how to enhance it, for identifying therapeutic targets through *in silico* approaches. It is composed of two works: i) an algorithm using Boolean network attractors for *in silico* target identification in Boolean models of pathologically disturbed biological networks, and ii) an enhancement of the Boolean network formalism in modeling the dynamics of biological networks through the incorporation of fuzzy operators and edge tuning.

Target identification, one of the steps of drug discovery, aims at identifying biomolecules whose function should be therapeutically altered in order to cure the considered pathology. The first work of this thesis proposes an algorithm for *in silico* target identification using Boolean network attractors. It assumes that attractors of dynamical systems, such as Boolean networks, correspond to phenotypes produced by the modeled biological system. Under this assumption, and given a Boolean network modeling a pathophysiology, the algorithm identifies target combinations able to remove attractors associated with pathological phenotypes. It is tested on a Boolean model of the mammalian cell cycle bearing a constitutive inactivation of the retinoblastoma protein, as seen in cancers, and its applications are illustrated on a Boolean model of Fanconi anemia. The results show that the algorithm returns target combinations able to remove attractors associated with pathological phenotypes and then succeeds in performing the proposed *in silico* target identification. However, as with any *in silico* evidence, there is a bridge to cross between theory and practice, thus requiring it to be used in combination with wet lab experiments. Nevertheless, it is expected that the

algorithm is of interest for target identification, notably by exploiting the inexpensiveness and predictive power of computational approaches to optimize the efficiency of costly wet lab experiments.

Quantitative modeling in systems biology can be difficult due to the scarcity of quantitative details about biological phenomenons, especially at the subcellular scale, the scale where drugs interact with there targets. An alternative to escape this difficulty is qualitative modeling since it requires few to no quantitative information. Among the qualitative modeling approaches, the Boolean network formalism is one of the most popular. However, Boolean models allow variables to be valued at only true or false, which can appear too simplistic when modeling biological processes. Consequently, the second work of this thesis proposes a modeling approach derived from Boolean networks where fuzzy operators are used and where edges are tuned. Fuzzy operators allow variables to be continuous and then to be more finely valued than with discrete modeling approaches, such as Boolean networks, while remaining qualitative. Moreover, to consider that some interactions are slower and/or weaker relative to other ones, edge states are computed in order to modulate in speed and strength the signal they convey. The proposed formalism is illustrated through its implementation on a tiny sample of the epidermal growth factor receptor signaling pathway. The obtained simulations show that continuous results are produced, thus allowing finer analysis, and that modulating the signal conveyed by the edges allows their tuning according to knowledge about the modeled interactions, thus incorporating more knowledge. The proposed modeling approach is expected to bring enhancements in the ability of qualitative models to simulate the dynamics of biological networks while not requiring quantitative information.

The main prospect of this thesis is to use the proposed enhancement of Boolean networks to build a version of the algorithm based on continuous dynamical systems. This will incorporate the accuracy of continuous simulations while keeping the advantageous of qualitative modeling. This prospect is not trivial since it requires to move from discrete to continuous dynamical systems. This brings some challenging differences such as the infinite cardinality of the state space. However, it is likely that the proposed enhance-

ment of Boolean networks can be mathematically expressed by differential equations. This means that the existing, and advanced, computational tools aimed at handling continuous dynamical systems, such as solvers of ordinary differential equations, could be directly used.

Keywords: biological network, Boolean network, therapeutic target, drug discovery, attractor, Fanconi anemia, multivalued logic, fuzzy logic, logic model, qualitative modeling

Résumé

Cette thèse est consacrée à la modélisation qualitative des réseaux biologiques pour l'innovation thérapeutique. Elle étudie comment utiliser les réseaux Booléens, et comment les améliorer, afin d'identifier des cibles thérapeutiques au moyen d'approches *in silico*. Elle se compose de deux travaux : i) un algorithme exploitant les attracteurs des réseaux Booléens pour l'identification *in silico* de cibles dans des modèles Booléens de réseaux biologiques pathologiquement perturbés, et ii) une amélioration des réseaux Booléens dans leur capacité à modéliser la dynamique des réseaux biologiques grâce à l'utilisation des opérateurs de la logique floue et grâce au réglage des arrêtes.

L'identification de cibles constitue l'une des étapes de la découverte de nouveaux médicaments et a pour but d'identifier des biomolécules dont la fonction devrait être thérapeutiquement modifiée afin de lutter contre la pathologie considérée. Le premier travail de cette thèse propose un algorithme pour l'identification *in silico* de cibles par l'exploitation des attracteurs des réseaux Booléens. Il suppose que les attracteurs des systèmes dynamiques, tel que les réseaux Booléens, correspondent aux phénotypes produits par le système biologique modélisé. Sous cette hypothèse, et étant donné un réseau Booléen modélisant une physiopathologie, l'algorithme identifie des combinaisons de cibles capables de supprimer les attracteurs associés aux phénotypes pathologiques. L'algorithme est testé sur un modèle Booléen du cycle cellulaire arborant une inactivation constitutive de la protéine du rétinoblastome, tel que constaté dans de nombreux cancers, tandis que ses applications sont illustrées sur un modèle Booléen de l'anémie de Fanconi. Les résultats montrent que l'algorithme est à même de retourner des combinaisons de cibles capables de supprimer les attracteurs associés aux phénotypes pathologiques,

et donc qu'il réussit l'identification *in silico* de cibles proposée. En revanche, comme tout résultat *in silico*, il y a un pont à franchir entre théorie et pratique, requérant ainsi une utilisation conjointe d'approches expérimentales. Toutefois, il est escompté que l'algorithme présente un intérêt pour l'identification de cibles, notamment par l'exploitation du faible coût des approches computationnelles, ainsi que de leur pouvoir prédictif, afin d'optimiser l'efficience d'expérimentations coûteuses.

La modélisation quantitative en biologie systémique peut s'avérer difficile en raison de la rareté des détails quantitatifs concernant les phénomènes biologiques, particulièrement à l'échelle subcellulaire, l'échelle où les médicaments interagissent avec leurs cibles. Une alternative permettant de contourner cette difficulté est la modélisation qualitative étant donné que celle-ci ne requiert que peu ou pas d'informations quantitatives. Parmi les méthodes de modélisation qualitative, les réseaux Booléens en sont l'une des plus populaires. Cependant, les modèles Booléens autorisent leurs variables à n'être évaluées qu'à vrai ou faux, ce qui peut apparaître trop simpliste lorsque des processus biologiques sont modélisés. En conséquence, le second travail de cette thèse propose une méthode de modélisation dérivée des réseaux Booléens où les opérateurs de la logique floue sont utilisés et où les arrêtes peuvent être réglées. Les opérateurs de la logique floue permettent aux variables d'être continues, et ainsi d'être plus finement évaluées qu'avec des méthodes de modélisation discrètes tel que les réseaux Booléens, tout en demeurant qualitatives. De plus, dans le but de considérer le fait que certaines interactions peuvent être plus lentes et/ou plus faibles que d'autres, l'état des arrêtes est calculé afin de moduler en vitesse et en force le signal qu'elles véhiculent. La méthode proposée est illustrée par son implémentation sur un petit échantillon de la signalisation du récepteur au facteur de croissance épidermique. Les simulations obtenues montrent que des résultats continus sont produits, permettant ainsi une analyse plus fine, et que moduler le signal véhiculé par les arrêtes permet leur réglage selon des connaissances concernant les interactions qu'elles modélisent, permettant ainsi d'incorporer davantage d'informations. Il est escompté que la méthode de modalisation proposée apportera des améliorations dans la capacité des modèles qualita-

tifs à simuler la dynamique des réseaux biologiques, tout en ne requérant pas d'informations quantitatives.

La principale perspective de cette thèse est d'utiliser l'amélioration proposée des réseaux Booléens pour construire une version de l'algorithme basée sur les systèmes dynamiques continus. Cela permettra de bénéficier de la précision des simulations continues tout en préservant les avantages de la modélisation qualitative. Toutefois, cette perspective n'est pas triviale car nécessite de passer des systèmes dynamiques discrets aux systèmes dynamiques continus. Cela amène quelques difficultés tel que la cardinalité infinie de l'espace d'état. Cependant, il est possible que l'amélioration proposée des réseaux Booléens puisse être mathématiquement exprimée par des équations différentielles. Cela signifie que les outils informatiques existants, et avancés, capables de gérer les systèmes dynamiques continus, tel que les solveurs d'équations différentielles, pourraient être directement utilisés.

Mots clés : réseau biologique, réseau Booléen, cible thérapeutique, découverte de médicament, attracteur, anémie de Fanconi, logique multivaluée, logique floue, modèle logique, modélisation qualitative

Contents

Chapter 1

Introduction

1.1 Thesis outline

This thesis is devoted to the qualitative modeling of biological networks for therapeutic innovation. It investigates how to use the Boolean network formalism, and how to enhance it, for identifying therapeutic targets through *in silico* approaches. It is composed of two works: i) an algorithm using Boolean network attractors for *in silico* target identification in Boolean models of pathologically disturbed biological networks, and ii) an enhancement of the Boolean network formalism in modeling the dynamics of biological networks through the incorporation of fuzzy operators and edge tuning. Each of these two works have their own, specific, introduction and conclusion, followed by a general conclusion at the end of this thesis. The first work is published, available in its publisher version [1] and author version [2]. Two further releases of the algorithm were done since its publication, all being freely available. The second work is submitted for publication to Comptes Rendus Biologies[2], the author version being already available [3].

[2] `http://www.journals.elsevier.com/comptes-rendus-biologies/`

1.2 The topic

This section introduces the topic of this thesis, namely the qualitative modeling of biological networks for therapeutic innovation. It starts with the modeling in systems biology and then focuses on qualitative modeling. Next, biological networks are introduced before describing how they can be modeled by Boolean networks. Finally, the intended applications in drug discovery, and more precisely for target identification, are introduced.

1.2.1 Modeling in systems biology

Modeling in systems biology allows scientists to produce formal models of biological systems and then to implement them on computers [4,5]. With such computational models, scientists can perform *in silico* experiments which have the advantage of being less costly in time and resources than the traditional wet lab experiments. However, the stumbling block of *in silico* approaches is that they are built from the available knowledge: not all is known about everything. Nevertheless, an impressive and ever increasing amount of biological knowledge is already available in the scientific literature, databases and knowledge bases such as, to name a few, DrugBank [6], KEGG [7], PharmGKB [8], Reactome [9] and TTD [10]. In addition to the difficulty of integrating an increasing body of knowledge comes the inherent complexity of biological systems themselves [11]: this is where computational tools can help owing to their integrative power [12–14]. This interplay between wet lab and computational biology is synergistic rather than competitive [15]. Since wet lab experiments produce factual results, they can be considered as trustworthy sources of knowledge. Once these factual pieces of knowledge are obtained, computational tools can help to integrate them and infer new ones. This computationally obtained knowledge can be subsequently used to direct further wet lab experiments, thus mutually potentiating the whole.

1.2.2 Qualitative modeling

One of the main difficulties encountered when quantitatively modeling biological systems with, for example, systems of differential equations [16] is that the required quantitative parameter values are not straightforward to obtain. One solution to overcome this barrier is qualitative modeling since it requires few to no quantitative information while producing informative predictions [17]. Several qualitative modeling approaches already exist and are mostly based on logic [18, 19] such as Boolean networks which are based on Boolean logic. However, this is at the cost of being qualitative: no quantification is performed. This does not mean that qualitative modeling is a downgrade of the quantitative one. This means that scientists have different approaches at their disposal, each with its advantages and disadvantages, depending on the pursued goals and available resources. If accurate numerical results are expected, quantitative modeling is required. However, if tendencies and global properties are the main concerns, qualitative modeling is entirely fitting and proved itself through several works [20–45].

1.2.3 Biological networks

A biological network is a way to conceptualize a set of interacting biological entities where entities are represented by nodes and interactions by edges [46, 47]. It is based on graph theory [48–50], thus bringing formal tools to encode information about biological systems, particularly their topology [51]. Moreover, being graphs, biological networks offer a convenient visualization [52] of the complex interconnections lying in biological systems. As said Napoleon Bonaparte:

> "A good sketch is better than a long speech."

Several types of biological networks can be encountered, depending on the scale, the involved entities and their interconnections. For example, at the ecological scale, food webs are biological networks where nodes represent species and edges represent trophic relations [53, 54]. At the subcellular scale there is, for example, gene regulatory networks where nodes represent gene

products and edges represent gene expression modulations [55,56]. Whatever the scale or entities, the principle remains the same: given a biological system, nodes represent entities and edges represent interactions between them.

1.2.4 Boolean networks

Boolean networks [57], pioneered in biology by Kauffman [58], Glass [59], Ostrander [60] and Thomas [61], are one of the existing qualitative modeling approaches. While being conceptually simple, Boolean networks are able to predict and reproduce features of biological systems and then to bring relevant insights [62–67]. This makes them an attractive and efficient approach, especially when the complexity of biological systems renders quantitative approaches unfeasible due to the amount of quantitative details they require.

As their name indicates, Boolean networks are based on Boolean logic [68] and, like biological networks, are also based on graph theory: nodes represent Boolean variables and edges represent interdependencies between them. Belonging to Boolean logic, Boolean variables can be valued at either true or false. This can appear somewhat simplistic when modeling biological processes, a point addressed in the second work of this thesis. Interdependencies between Boolean variables are mathematically implemented by Boolean functions: a Boolean function is assigned to each variable and defines its value according to the one of the variables interacting with it. Boolean functions manipulate the value of their arguments through Boolean operators, namely AND, OR and NOT.

Starting from an initial state, that is a vector containing the initial value of all the variables of the network, its is possible to simulate its dynamics by computing the value of the state vector along a given number of iterations. To do so, at each iteration, the value of the state vector is updated through the Boolean functions according to its current and/or previous value, depending on the updating scheme. Therefore, Boolean networks can be seen as discrete dynamical systems able to model the dynamics of biological networks. To this end, each entity of the biological network is modeled by a Boolean variable and the interactions between them are encode in Boolean functions.

Boolean networks can be classified according to their updating scheme as synchronous or asynchronous. If all the variables are updated simultaneously at each iteration then the network is synchronous, otherwise it is asynchronous. While there is only one synchronous updating scheme, different asynchronous updating schemes exist:

- the random order asynchronous updating scheme where, at each iteration, an updating order for the variables is randomly selected
- the general asynchronous updating scheme where, at each iteration, a randomly selected variable is updated
- the deterministic asynchronous updating scheme where a divisor is assigned to each variable and then, at each iteration, a variable is updated if and only if the iteration is a multiple of its divisor

With the exception of deterministic asynchronous Boolean networks, only synchronous Boolean networks are deterministic since, at each iteration, variables have only one possible successor. This makes synchronous Boolean networks easier to compute than asynchronous ones [69].

1.2.5 Target identification

Drug discovery, as its name indicates, aims at discovering new drugs against diseases. This process can be segmented into three steps: i) disease model provision, where experimental models are developed, ii) target identification, where therapeutic targets are proposed, and iii) target validation, where the proposed therapeutic targets are assessed. The applications of this thesis in therapeutic innovation, namely through the first work, focus on the second step of drug discovery: target identification [70, 71].

Given an organism suffering from a disease, target identification aims at finding where to act among its multitude of biomolecules in order to alleviate, or ultimately cure, the physiological consequences of the disease. These biomolecules on which perturbations should be applied are called targets and are targeted by drugs [72]. This raises two questions: which target should be

therapeutically perturbed and what type of perturbation should be applied on it. Broadly, the functional perturbation of a target by a drug can be either activating or inactivating, regardless the way the drug achieves it.

One solution is to test all, or at least a large number of, biomolecules for activation and inactivation. Knowing that targeting several biomolecules is potentially more effective [73, 74], the number of possibilities is consequently huge. This rather brute-force screening can be refined with knowledge about the pathophysiology of interest by identifying potential targets based on the role they play in it [75]. Even with this knowledge, experimentally assessing the selected potential targets through wet lab experiments is far from straightforward since such experiments are costly in time and resources [76]. Fortunately, owing to their integrative power and low cost compared to wet lab approaches, *in silico* approaches appear as valuable tools in improving the efficiency of target identification [77–88], as demonstrated through several works using various computational methods [89–99].

Chapter 2

Therapeutic target discovery using Boolean network attractors: avoiding pathological phenotypes

This chapter describes the first work of this thesis. The first four sections are mostly the corresponding article as published, while the last section describes improvements done after publication.

2.1 Background

The goal of this work is to propose a computational methodology implemented in an algorithm for *in silico* target identification using Boolean network attractors. It assumes that Boolean network attractors correspond to phenotypes produced by the modeled biological network, an assumption successfully applied in several works [20–23, 27, 28, 32–34, 36, 39, 40, 43–45, 100, 101]. Assuming that a phenotype is an observable and thus a relatively stable state of a biological system and assuming that the state of a biological system results from its dynamics, a phenotype is likely to correspond to an attractor. This assumption can be stated for any dynamical model but, in this work, only Boolean networks are considered. Reasons are that, in their most basic form, Boolean networks do not require quantitative information and that

quantitative information is often not easy to obtain due to experimental limitations, particularly at the subcellular scale, the scale where drugs interact with their targets. Moreover, since synchronous Boolean networks are easier to compute than asynchronous ones, this work only considers synchronous Boolean networks. This does not exclude the possibility, at a later stage, to extend the algorithm for both synchronous and asynchronous updating schemes.

For a biological network involved in a disease, two possible variants are considered: the physiological variant, exhibited by healthy organisms, which produces physiological phenotypes, and the pathological variant, exhibited by ill organisms, which produces pathological phenotypes, or which fails to produce physiological ones. A physiological phenotype does not impair life quantity/quality whereas a pathological phenotype does. It should be noted that the loss of a physiological phenotype is also a pathological condition. The physiological and pathological variants differ in that the latter results from the occurrence of some alterations known to be responsible for disorders. With a pathological variant, there are two non-exclusive pathological scenarios: pathological phenotypes are gained or physiological phenotypes are lost.

The primary goal of the proposed algorithm is to identify, in a pathological variant, target combinations together with the perturbations to apply on them, here called bullets, which render it unable to exhibit pathological phenotypes. The secondary goal is to classify the obtained bullets according to their ability at rendering the pathological variant able to exhibit previously lost physiological phenotypes, if any.

2.2 Methods

This section introduces some basic principles, namely biological and Boolean networks, defines some concepts and then describes the proposed algorithm. An example network to illustrate how it works plus a case study to illustrate its intended applications are also described. Finally, details about implementation and code availability are mentioned.

2.2.1 Basic principles

2.2.1.1 Biological networks

A network can be seen as a digraph $G = (V, E)$ where $V = \{v_1, \ldots, v_n\}$ is the set of cardinality n containing exactly all the nodes v_i of the network and where $E = \{(v_{i,1}, v_{j,1}), \ldots, (v_{i,m}, v_{j,m})\} \subseteq V^2$ is the set of cardinality m containing exactly all the edges (v_i, v_j) of the network. In practice, nodes represent entities and edges represent binary relations $R \subseteq V^2$ involving them: $v_i \ R \ v_j$. For example, in gene regulatory networks, nodes represent gene products and edges represent gene expression modulations.

2.2.1.2 Boolean networks

A Boolean network is a network where nodes are Boolean variables x_i and where edges (x_i, x_j) represent the binary *is input of* relation: x_i *is input of* x_j. Each x_i has $b_i \in [\![0, n]\!]$ inputs $x_{i,1}, \ldots, x_{i,b_i}$. The variables which are not inputs of x_i have no direct influence on it. If $b_i = 0$ then x_i is a parameter and does not depend on other variables. At each iteration $k \in [\![k_0, k_{end}]\!]$ of the simulation, the value $x_i(k) \in \{0, 1\}$ of each x_i is updated to the value $x_i(k+1)$ using a Boolean function f_i and the values $x_{i,1}(k), \ldots, x_{i,b_i}(k)$ of its inputs, as in the following pseudocode:

```
1 for k ∈ [[k_0, k_end − 1]] do
2       x_1(k + 1) = f_1(x_{1,1}(k), ..., x_{1,b_1}(k))
3       ...
4       x_n(k + 1) = f_n(x_{n,1}(k), ..., x_{n,b_n}(k))
5 end for
```

which can be written in a more concise form:

```
1 for k ∈ [[k_0, k_end − 1]] do
2       x(k + 1) = f(x(k))
3 end for
```

where $\boldsymbol{f} = (f_1, \ldots, f_n)$ is the Boolean transition function and $\boldsymbol{x} = (x_1, \ldots, x_n)$ is the state vector. The value $\boldsymbol{x}(k) = (x_1(k), \ldots, x_n(k)) \in \{0, 1\}^n$ of $\boldsymbol{x}$ at k belongs to the state space $S = \{0, 1\}^n$ which is the set of cardinality 2^n

containing exactly all the possible states.

If the values of all the x_i are updated simultaneously at each k then the network is synchronous, otherwise it is asynchronous. With synchronous Boolean networks, $\boldsymbol{x}(k)$ has a unique possible successor $\boldsymbol{x}(k+1)$: synchronous Boolean networks are deterministic. In the particular case where $k = k_0$, $\boldsymbol{x}(k_0) = \boldsymbol{x}_0$ is the initial state and, in deterministic dynamical systems, determines entirely the trajectory $w = (\boldsymbol{x}(k_0), \ldots, \boldsymbol{x}(k_{end}))$. In this work, it is assumed that $k_0 = 1$, so w is a sequence of length k_{end} resulting from the iterative computation of $\boldsymbol{x}(k)$ from k_0 up to k_{end}. This iterative computation can be seen as the discretization of a time interval: Boolean networks are discrete dynamical systems as they simulate discretely the time course of the state vector.

The set $A = \{a_1, \ldots, a_p\}$ of cardinality p containing exactly all the attractors a_i is called the attractor set. Due to the determinism of synchronous Boolean networks, all the attractors are cycles. A cycle is a sequence $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_q)$ of length q such that $\forall j \in [\![1, q]\!],\ \boldsymbol{x}_{j+1} = \boldsymbol{f}(\boldsymbol{x}_j)$ and $\boldsymbol{x}_{q+1} = \boldsymbol{x}_1$: once the system reaches a state $\boldsymbol{x}_j$ belonging to a cycle, it successively visits its states $\boldsymbol{x}_{j+1}, \ldots, \boldsymbol{x}_q, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_j$ for infinity. In the particular case where $q = 1$, a_i is a point attractor. The set $B_i \subseteq S$ containing exactly all the $\boldsymbol{x} \in S$ from which a_i can be reached is called its basin of attraction. With deterministic dynamical systems, the family of sets $(B_1, \ldots, B_p)$ constitutes a partition of S.

2.2.2 Definitions

Some concepts used in this work should be formally defined.

- **physiological phenotype**: A phenotype which does not impair the life quantity/quality of the organism which exhibits it.
- **pathological phenotype**: A phenotype which impairs the life quantity/quality of the organism which exhibits it.
- **variant (of a biological network)**: Given a biological network of interest, a variant is one of its versions, namely the network plus even-

tually some modifications. It should be noted that this does not exclude the possibility that a variant can be the network of interest as is.

- **physiological variant**: A variant which produces only physiological phenotypes. It is the biological network of interest as it should be, namely the one of healthy organisms.
- **pathological variant**: A variant which produces at least one pathological phenotype. It is a dysfunctional version of the biological network of interest, namely a version found in ill organisms.
- **physiological attractor set**: The attractor set A_{physio} of the physiological variant.
- **pathological attractor set**: The attractor set A_{patho} of the pathological variant.
- **physiological Boolean transition function**: The Boolean transition function $\boldsymbol{f}_{physio}$ of the physiological variant.
- **pathological Boolean transition function**: The Boolean transition function $\boldsymbol{f}_{patho}$ of the pathological variant.
- **run**: An iterative computation of $\boldsymbol{x}(k)$ starting from an $\boldsymbol{x}_0$ until an a_i is reached. It returns $w = (\boldsymbol{x}(k_0), \ldots, \boldsymbol{x}(k_{end}))$ where k_{end} depends on when a_i is reached, and then on $\boldsymbol{x}_0$.
- **physiological attractor**: An a_i such that $a_i \in A_{physio}$.
- **pathological attractor**: An a_i such that $a_i \notin A_{physio}$.
- **modality**: The functional perturbation $moda_i$ applied on a node $v_j \in V$ of the network, either activating ($moda_i = 1$) or inactivating ($moda_i = 0$): at each k, $moda_i$ overwrites $f_j(\boldsymbol{x}(k))$ making $x_j(k+1) = moda_i$.
- **target**: A node $targ_i \in V$ of the network on which a $moda_i$ is applied.

- **bullet**: A couple (c_{targ}, c_{moda}) where $c_{targ} = (targ_1, \ldots, targ_r)$ is a combination without repetition of $targ_i$ and where $c_{moda} = (moda_1, \ldots, moda_r)$ is an arrangement with repetition of $moda_i$, $r \in [\![1, n]\!]$ being the number of targets in the bullet. Here, $moda_i$ is intended to be applied on $targ_i$.
- **therapeutic bullet**: A bullet which makes $A_{patho} \subseteq A_{physio}$.
- **silver bullet**: A therapeutic bullet which makes $A_{patho} \subsetneq A_{physio}$.
- **golden bullet**: A therapeutic bullet which makes $A_{patho} = A_{physio}$.

The assumed link between phenotypes and attractors is the reason why attractors are qualified as either physiological or pathological according to the phenotype they produce. This is also the reason why, in this work, target identification aims at manipulating attractor sets of pathological variants.

2.2.3 Steps of the algorithm

The algorithm has two goals: i) finding therapeutic bullets, and ii) classifying them as either golden or silver. A therapeutic bullet makes the pathological variant unable at reaching pathological attractors, that is $A_{patho} \subseteq A_{physio}$. If such a bullet is applied on a pathological variant, the organism bearing it no longer exhibits the associated pathological phenotypes. However, a therapeutic bullet does not necessarily preserve/restore the physiological attractors. If a therapeutic bullet preserves/restores all the physiological attractors, that is if $A_{patho} = A_{physio}$, then it is a golden one, but if $A_{patho} \subsetneq A_{physio}$ then it is a silver one.

Given a physiological and a pathological variant, that is $\boldsymbol{f}_{physio}$ and $\boldsymbol{f}_{patho}$, the algorithm follows five steps:

1. with $\boldsymbol{f}_{physio}$ it computes the control attractor set A_{physio}
2. it generates bullets and, for each of them, it performs the three following steps
3. with $\boldsymbol{f}_{patho}$ plus the bullet, it computes the variant attractor set A_{patho}

4. it assesses the therapeutic potential of the bullet by comparing A_{physio} and A_{patho} to detect pathological attractors

5. if the bullet is therapeutic then it is classified as either golden or silver by comparing A_{physio} and A_{patho} for equality

These steps can be written in pseudocode as:

```
1  with f_physio compute A_physio
2  generate bullet_set
3  for bullet ∈ bullet_set do
4       with f_patho plus bullet compute A_patho
5       if A_patho ⊆ A_physio then
6            bullet is therapeutic
7            if A_patho = A_physio then
8                 bullet is golden
9            else
10                bullet is silver
11           end if
12      end if
13 end for
```

The algorithm is described step by step but can be found as one block of pseudocode in *Appendix A* page 85.

2.2.3.1 Step 1: computing A_{physio}

First of all, A_{physio} has to be computed since it is the control and, as such, determines what is pathological. To do so, runs are performed with $\boldsymbol{f}_{physio}$ and the reached a_i are stored in A_{physio}. However, $\boldsymbol{x}_0 \in S$ and $card\ S$ increases exponentially with n. Even for reasonable values of n, $card\ S$ explodes: more than 1 000 000 possible $\boldsymbol{x}_0$ for $n = 20$. One solution ensuring that all the a_i are reached is to start a run from each of the possible $\boldsymbol{x}_0$, that is from each of the $\boldsymbol{x} \in S$. Practically, this is unfeasible for an arbitrary value of n since the required computational capacity can be too demanding. For example, assuming that a run requires 1 millisecond and that $n = 50$,

performing a run from each of the 2^{50} $\boldsymbol{x} \in S$ requires nearly 36 000 years.

Given that with deterministic dynamical systems $(B_1, \ldots, B_p)$ is a partition of S, a solution is to select a subset $D \subseteq S$ of a reasonable cardinality containing the $\boldsymbol{x}_0$ to start from. In this work, D is randomly selected from a uniform distribution. The stumbling block of this solution is that it does not ensure that at least one $\boldsymbol{x}_0$ per B_i is selected and then does not ensure that all the a_i are reached. This stumbling block holds only if $card\ D < card\ S$.

Again given that synchronous Boolean networks are deterministic, if a run visits a state already visited during a previous run then its destination, that is the reached attractor, is already found. If so, the run can be stopped and the algorithm can jump to the next one. To implement this, the previous trajectories are stored in a set H, the history, and at each k the algorithm checks if $\exists w \in H \colon \boldsymbol{x}(k) \in w$. If this check is positive then the algorithm jumps to the next run.

To detect the attractors, since with deterministic dynamical systems they are cycles, the algorithm checks at each k if $\boldsymbol{x}(k+1)$ is an already visited state of the current run, namely if $\exists k' \in [\![1, k]\!] \colon \boldsymbol{x}(k+1) = \boldsymbol{x}(k')$. If this check is positive then $a_i = (\boldsymbol{x}(k'), \ldots, \boldsymbol{x}(k))$.

This step can be written in pseudocode as:

prompt *card D*
$card\ D = min(card\ D, 2^n)$
generate $D \subseteq S$
$H = \{\}$
$A_{physio} = \{\}$
for $x_0 \in D$ **do**
    $k = 1$
    $\boldsymbol{x}(k) = x_0$
    while true do
        if $\exists w \in H \colon \boldsymbol{x}(k) \in w$ **then**
            break
        end if
        $\boldsymbol{x}(k+1) = \boldsymbol{f}_{physio}(\boldsymbol{x}(k))$
        if $\exists k' \in [\![1, k]\!] \colon \boldsymbol{x}(k+1) = \boldsymbol{x}(k')$ **then**

15 $A_{physio} = A_{physio} \cup \{(\boldsymbol{x}(k'), \ldots, \boldsymbol{x}(k))\}$
16 **break**
17 **end if**
18 $k = k + 1$
19 **end while**
20 $H = H \cup \{(\boldsymbol{x}(1), \ldots, \boldsymbol{x}(k))\}$
21 **end for**
22 **return** A_{physio}
23 do step 2

Line 2 catches the mistake $card\ D > card\ S$.

It should be noted that the purpose of this work is not to propose an algorithm for finding Boolean network attractors since advanced algorithms for such tasks are already published [102–106]. The purpose is to propose a computational methodology exploiting Boolean network attractors for *in silico* target identification, a methodology which requires *de facto* these attractors to be found. This point is discussed in the *Discussion* section page 46.

2.2.3.2 Step 2: generating bullets

Bullets are candidate perturbations to apply on the pathological variant to make it unable at reaching pathological attractors and then unable at producing pathological phenotypes. Generating a bullet requires a choice of $targ_i \in V$ and associated $moda_i \in \{0, 1\}$. In this work, there is no sequencing in target engagement nor in modality application. This means that, given a bullet and during a given run, all the $moda_i$ are applied on their corresponding $targ_i$ throughout the run. As a consequence, for a given bullet, choosing the same $targ_i$ more than once is senseless, while it is possible to choose the same $moda_i$ for more than one $targ_i$. Therefore, a bullet is a combination c_{targ} without repetition of $targ_i$ together with an arrangement c_{moda} with repetition of $moda_i$.

If bullets containing r targets have to be generated then there are $n!/(r! \cdot (n-r)!)$ possible c_{targ} and, for each of them, there are 2^r possible c_{moda}. This

raises the same computational difficulty than with the state space explosion since there are $(n!\cdot 2^r)/(r!\cdot(n-r)!)$ possible bullets. For example, with $n = 50$ and $r = 3$ there are more than 150 000 possible bullets. Knowing that the algorithm, as explained below, computes one attractor set per bullet, the computation time becomes practically unfeasible. To overcome this barrier, the algorithm asks for r as an interval $[\![r_{min}, r_{max}]\!]$, asks for a maximum number max_{targ} of c_{targ} to generate and asks for a maximum number max_{moda} of c_{moda} to test for each c_{targ}. The algorithm then generates a set C_{targ} of c_{targ} with $card\ C_{targ} \leq max_{targ}$ by randomly selecting, from a uniform distribution and without repetition, nodes in the network. In the same way, the algorithm generates a set C_{moda} of c_{moda} with $card\ C_{moda} \leq max_{moda}$ by randomly choosing, from a uniform distribution and with repetition, modalities as either activating (1) or inactivating (0). The result is the bullets: per $r \in [\![r_{min}, r_{max}]\!]$, a C_{targ} together with a C_{moda}. As with the state space explosion, the stumbling block of this method is that it does not ensure that all the possible c_{targ} together with all the possible c_{moda} are tested. This stumbling block holds only if $max_{targ} < n!/(r! \cdot (n-r)!)$ or $max_{moda} < 2^r$.

This step can be written in pseudocode as:

prompt $r_{min}, r_{max}, max_{targ}, max_{moda}$
$r_{max} = min(r_{max}, n)$
$golden\_set = \{\}$
$silver\_set = \{\}$
for $r \in [\![r_{min}, r_{max}]\!]$ **do**
    $max^r_{targ} = min(max_{targ}, n!/(r! \cdot (n-r)!))$
    $max^r_{moda} = min(max_{moda}, 2^r)$
    $C_{targ} = \{\}$
    $C_{moda} = \{\}$
    while $card\ C_{targ} < max^r_{targ}$ **do**
        generate $c_{targ} \notin C_{targ}$
        $C_{targ} = C_{targ} \cup \{c_{targ}\}$
    end while
    while $card\ C_{moda} < max^r_{moda}$ **do**
        generate $c_{moda} \notin C_{moda}$

16 $C_{moda} = C_{moda} \cup \{c_{moda}\}$
17 **end while**
18 do steps 3 to 5
19 **end for**
20 **return** $golden\_set, silver\_set$

Line 2 catches the mistake $r > n$. Lines 3 and 4 create the sets in which the therapeutic bullets found in step 4 are classified as either golden or silver in step 5. Lines 6 and 7 catch the mistake where max_{targ} or max_{moda} is greater than its maximum, which depends on r, hence the creation of max^r_{targ} and max^r_{moda} to preserve the initially supplied value. Lines 11 and 15 ensure that only new c_{targ} and c_{moda} are generated.

2.2.3.3 Step 3: computing A_{patho}

Having the control attractor set A_{physio} and a bullet $(c_{targ}, c_{moda}) \in C_{targ} \times C_{moda}$, the algorithm computes the variant attractor set A_{patho} under the effect of (c_{targ}, c_{moda}) by almost the same way A_{physio} is computed in step 1. However, $\boldsymbol{f}_{patho}$ is used instead of $\boldsymbol{f}_{physio}$ and (c_{targ}, c_{moda}) is applied: at each k, $f_j(\boldsymbol{x}(k))$ is overwritten by $moda_i \in c_{moda}$, that is $x_j(k+1) = moda_i$, provided that $v_j = targ_i \in c_{targ}$. In order to apply all the generated bullets, the algorithm uses two nested for loops. For each $c_{targ} \in C_{targ}$ it uses successively all the $c_{moda} \in C_{moda}$. For each (c_{targ}, c_{moda}), the algorithm computes the corresponding A_{patho} and does steps 4 and 5.

This step can be written in pseudocode as:

1 **for** $c_{targ} \in C_{targ}$ **do**
2 **for** $c_{moda} \in C_{moda}$ **do**
3 $H = \{\}$
4 $A_{patho} = \{\}$
5 **for** $x_0 \in D$ **do**
6 $k = 1$
7 $\boldsymbol{x}(k) = x_0$
8 **while true do**
9 **if** $\exists w \in H : \boldsymbol{x}(k) \in w$ **then**

```
10                      break
11                  end if
12                  x(k+1) = f_patho(x(k))
13                  for targ_i ∈ c_targ do
14                      for v_j ∈ V do
15                          if v_j = targ_i then
16                              x_j(k+1) = moda_i
17                          end if
18                      end for
19                  end for
20                  if ∃k' ∈ [[1, k]]: x(k+1) = x(k') then
21                      A_patho = A_patho ∪ {(x(k'), ..., x(k))}
22                      break
23                  end if
24                  k = k + 1
25              end while
26              H = H ∪ {(x(1), ..., x(k))}
27          end for
28          do step 4 and 5
29      end for
30  end for
```

Lines 13–19 are where bullets are applied.

2.2.3.4 Step 4: identifying therapeutic bullets

To identify therapeutic bullets among the generated ones, for each (c_{targ}, c_{moda}) tested in step 3 and once the corresponding A_{patho} is obtained, the algorithm compares it with A_{physio} to check if $A_{patho} \subseteq A_{physio}$. This check ensures that all the pathological attractors are removed and that if new attractors appear then they are physiological. If this check is positive then the bullet is therapeutic and the algorithm pursues with step 5.

This step can be written in pseudocode as:

```
1 if A_patho ⊆ A_physio then
```

2 do step 5
3 **end if**

2.2.3.5 Step 5: assessing therapeutic bullets

Therapeutic bullets are qualified as either golden or silver according to their ability at making the pathological variant reaching the physiological attractors. All therapeutic bullets, being golden or silver, remove the pathological attractors without creating new ones, that is $A_{patho} \subseteq A_{physio}$. However, this does not imply that they preserve/restore the physiological attractors. A golden bullet preserves/restores all the physiological attractors: $A_{patho} = A_{physio}$ whereas a silver bullet does not: $A_{patho} \subsetneq A_{physio}$.

In this setting, golden bullets are perfect therapies whereas silver bullets are not. However, since precious things are rare and just as gold is rarer than silver, finding golden bullets is less likely than finding silver ones. Indeed, given that more constraints are required for a therapeutic bullet to be a golden one, it is more likely that the found therapeutic bullets are silver ones, except in one case: $card\ A_{physio} = 1$.

Theorem 1. *If* $card\ A_{physio} = 1$ *then all therapeutic bullets are golden.*

Proof.

$$\begin{aligned}
&(\text{therapeutic bullet}) \Rightarrow (A_{patho} \subseteq A_{physio}) && (1)\\
&(1) \Rightarrow (A_{patho} \in \mathcal{P}(A_{physio})) && (2)\\
&(card\ A_{physio} = 1) \Rightarrow (A_{physio} = \{a\}) && (3)\\
&(3) \Rightarrow (\mathcal{P}(A_{physio}) = \{\emptyset, \{a\}\}) && (4)\\
&((2) \wedge (4)) \Rightarrow ((A_{patho} = \{a\}) \vee (A_{patho} = \emptyset)) && (5)\\
&(\text{deterministic dynamical systems}) \Rightarrow (A \neq \emptyset) && (6)\\
&(6) \Rightarrow (A_{patho} \neq \emptyset) && (7)\\
&((5) \wedge (7)) \Rightarrow (A_{patho} = \{a\}) && (8)\\
&((3) \wedge (8)) \Rightarrow (A_{patho} = A_{physio}) && (9)\\
&(9) \Rightarrow (\text{therapeutic bullet is golden}) && (10)
\end{aligned}$$

□

Practically, in this setting, an organism bearing a pathological variant treated with a therapeutic bullet no longer exhibits the associated pathological phenotypes. Moreover, if the therapeutic bullet is golden then the organism exhibits the same phenotypes than its healthy counterpart. However, if the therapeutic bullet is silver then the organism fails to exhibit at least one physiological phenotype. With a silver bullet this is a matter of choice: what is the less detrimental between a silver bullet and no therapeutic bullet at all.

This step can be written in pseudocode as:

1 **if** $A_{patho} = A_{physio}$ **then**
2 $\quad golden\_set = golden\_set \cup \{(c_{targ}, c_{moda})\}$
3 **else**
4 $\quad silver\_set = silver\_set \cup \{(c_{targ}, c_{moda})\}$
5 **end if**

2.2.4 Example network

To illustrate the algorithm, it is used on a Boolean model of the mammalian cell cycle published by Faure *et al* [43]. This model is chosen for several reasons: i) a synchronous updating is performed: to date, the algorithm focuses on synchronous Boolean networks, ii) a mammalian biological system is modeled: the closer to human physiology the model is, the better it illustrates the intended applications, iii) the cell cycle is a at the heart of cancer: this gives relevancy to the example network, iv) the network comprises ten nodes: easily computable in face of its state space, and v) attractors are already computed: useful to validate the algorithm in finding them.

A graphical representation of the example network is shown in *Figure 2.1* page 35. Below are the corresponding Boolean functions where, for the sake of readability, x_i stands for $x_i(k)$ and x_{i+} stands for $x_i(k+1)$:

$$
\begin{aligned}
CycD_+ &= CycD \\
Rb_+ &= (\neg CycD \wedge \neg CycE \wedge \neg CycA \wedge \neg CycB) \vee (p27 \wedge \neg CycD \wedge \neg CycB) \\
E2F_+ &= (\neg Rb \wedge \neg CycA \wedge \neg CycB) \vee (p27 \wedge \neg Rb \wedge \neg CycB) \\
CycE_+ &= E2F \wedge \neg Rb \\
CycA_+ &= (E2F \wedge \neg Rb \wedge \neg Cdc20 \wedge \neg(Cdh1 \wedge UbcH10)) \\
&\quad \vee (CycA \wedge \neg Rb \wedge \neg Cdc20 \wedge \neg(Cdh1 \wedge UbcH10)) \\
p27_+ &= (\neg CycD \wedge \neg CycE \wedge \neg CycA \wedge \neg CycB) \\
&\quad \vee (p27 \wedge \neg(CycE \wedge CycA) \wedge \neg CycB \wedge \neg CycD) \\
Cdc20_+ &= CycB \\
Cdh1_+ &= (\neg CycA \wedge \neg CycB) \vee Cdc20 \vee (p27 \wedge \neg CycB) \\
UbcH10_+ &= \neg Cdh1 \vee (Cdh1 \wedge UbcH10 \wedge (Cdc20 \vee CycA \vee CycB)) \\
CycB_+ &= \neg Cdc20 \wedge \neg Cdh1
\end{aligned}
$$

Having the example network, two variants are needed: the physiological one and the pathological one. The physiological variant is the network as is while the pathological variant is the network plus a constitutive activation/inactivation of at least one of its nodes. For simplicity, and given the relatively small number of entities, only one is chosen: the retinoblastoma protein Rb for which a constitutive inactivation is applied. To implement this, the corresponding f_i becomes:

$$Rb(k+1) = 0$$

in $\boldsymbol{f}_{patho}$. Rb is chosen because its inactivation occurs in many cancers [107]. Therefore, a network bearing a constitutive inactivation of it should be a relevant example of a pathological variant.

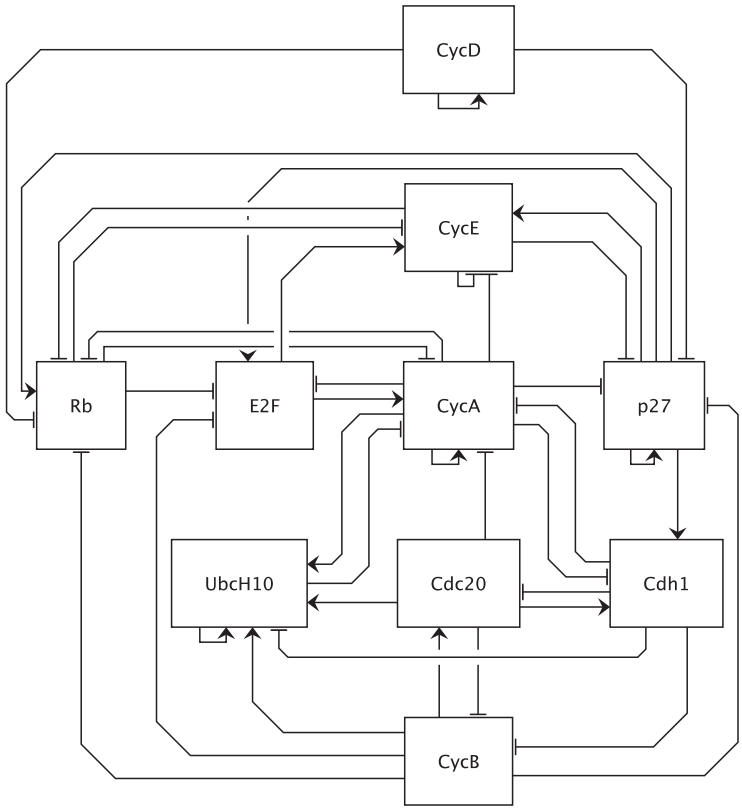

Figure 2.1 – Graphical representation of the example network adapted from [43]. CDKs (cyclin-dependent kinases) are the catalytic partners of cyclins and, in this model, are not explicitly shown since the activity of CDK-cyclin complexes essentially depends on cyclins. Furthermore, inhibition of E2F by Rb is modeled by opposing Rb to the effects of E2F on its targets. The same applies to inhibition of CycE and CycA by p27. For a complete description of the model, see [43]. CycD: CDK4/6-cyclin D complex, input of the model, initiates the cell cycle, activated by positive signals such as growth factors; CycE: CDK2-cyclin E complex; CycA: CDK2-cyclin A complex; CycB: CDK1-cyclin B complex; Rb: retinoblastoma protein, a tumor suppressor; E2F: a family of transcription factors divided into activator and repressor members, in this model E2F represents the activator members; p27: p27/Kip1, a CKI (CDK inhibitor); Cdc20: an APC (Anaphase Promoting Complex, an E3 ubiquitin ligase) activator; Cdh1: an APC activator; UbcH10: an E2 ubiquitin conjugating enzyme.

2.2.5 Case study

To illustrate the intended usage of the proposed methodology, the algorithm is used on a Boolean model of the Fanconi Anemia/Breast Cancer (FA/BRCA) pathway published by Rodriguez *et al* [27]. This model is chosen

for several reasons: i) two pathological conditions are studied: required for a case study of an *in silico* target identification, ii) the physiological and pathological variants are clearly described: required by the algorithm, iii) it is nearly three times bigger than the example network: representative of a more comprehensive biological model while remaining computationally tractable, iv) synchronous updating is used: to date, the algorithm focuses on synchronous Boolean networks, and v) attractors are already interpreted in terms of phenotypes.

The FA/BRCA pathway is dedicated to DNA repair, more precisely to interstrand cross-link (ICL) removal. As expected with any DNA repair impairment, individuals suffering from FA/BRCA pathway malfunction are subjected to increased risk of cancer, such as in Fanconi anemia, a rare genetic disorder causing bone marrow failure, congenital abnormalities and increased risk of cancer [108–110]. Rodriguez *et al* propose a Boolean model comprising the FA/BRCA pathway and three types of DNA damages commonly observed in Fanconi anemia, namely ICLs, double-strand breaks (DSBs) and DNA adducts (ADDs). It should be noted that the ICL repair process creates DSBs and ADDs before removing them, thus leaving an undamaged DNA ready for the cell cycle. For a complete description of the model, see [27]. The corresponding Boolean functions can be found in *Appendix B* page 88.

The physiological variant is the FA/BRCA pathway model as is. To it, Rodriguez *et al* propose two pathological variants, here called *patho1* and *patho2*, modeling two mutations involving genes of the FA/BRCA pathway. These mutations are observed in patients suffering from Fanconi anemia [111]. The first one involves the FANCA gene, corresponding to the $FAcore$ variable, and the second one involves the FANCD1/BRCA2 or FANCN/PALB2 gene, corresponding to the $FANCD1N$ variable. These mutations are of loss-of-function kind: to simulate them the corresponding f_i become

$$FAcore(k+1) = 0$$

for FANCA gene null mutation in $\boldsymbol{f}_{patho1}$ and

$$FANCD1N(k+1) = 0$$

for FANCD1/BRCA2 or FANCN/PALB2 gene null mutation in $\boldsymbol{f}_{patho2}$.

2.2.6 Implementation

The algorithm is implemented in Fortran compiled with GFortran[1]. The code is available on GitHub[2] at `https://github.com/arnaudporet/kali`.

2.3 Results

In this section, results produced with the algorithm on the example network are exposed to illustrate how it works. Next, results produced with the algorithm on the case study are exposed to illustrate its intended applications for target identification.

2.3.1 Results of step 1

Owing to the relatively small size of the example network, *card D* is set to *card S* = 1024. Since *card D* = *card S*, all the attractors are found. Attractors are presented as matrices where, for an attractor of length q, lines correspond to the $x_i(k)$, $k \in [\![1, q]\!]$, and columns to $\boldsymbol{x}(k)$. The algorithm returns the following attractors:

[1] `http://www.gnu.org/software/gcc/fortran/`
[2] `https://github.com/`

$$a_1 = \begin{array}{l|ccccccc} CycD & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ Rb & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ E2F & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ CycE & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ CycA & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ p27 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Cdc20 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ Cdh1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ UbcH10 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ CycB & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{array}$$

$$a_2 = \begin{array}{l|c} CycD & 0 \\ Rb & 1 \\ E2F & 0 \\ CycE & 0 \\ CycA & 0 \\ p27 & 1 \\ Cdc20 & 0 \\ Cdh1 & 1 \\ UbcH10 & 0 \\ CycB & 0 \end{array}$$

each of them attracting 50% of the $\boldsymbol{x} \in S$ under $\boldsymbol{f}_{physio}$. Then, $A_{physio} = \{a_1, a_2\}$ and corresponds to the results obtained by Faure *et al.* In terms of phenotypes, a_1 corresponds to cell cycle whereas a_2 corresponds to quiescence.

2.3.2 Results of steps 2 to 5

Results of steps 2 to 5 are grouped since only the therapeutic bullets found in step 4 and classified in step 5 are returned. The algorithm is launched with $r_{min} = 1$ and $r_{max} = 2$. Due to the relatively small size of the example network, max_{targ} and max_{moda} are set to their maximum, namely $max_{targ} = 45$ and $max_{moda} = 4$. Consequently, all the possible bullets made of 1 to 2 targets are tested. The algorithm returns the following therapeutic bullets:

$+CycD$		silver
$+CycD$	$-p27$	silver
$-CycD$	$+Rb$	silver
$+CycD$	$-Rb$	silver

where + means therapeutic activation and − means therapeutic inactivation. It should be noted that no golden bullets are found, an unsurprising result since they are rarer than silver ones.

Given these results, therapeutic activation of Rb alone, which is pathologically inactivated, is not enough to remove the pathological attractors. Indeed, as seen in the third bullet, therapeutic activation of Rb must be accompanied by therapeutic inactivation of CycD. To better illustrate what is performed to obtain these therapeutic bullets, below is A_{patho} without any bullet:

$$a_3 = \begin{array}{l|cccccccc} CycD & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Rb & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ E2F & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ CycE & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ CycA & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ p27 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ Cdc20 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ Cdh1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ UbcH10 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ CycB & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}$$

$$a_4 = \begin{array}{c|ccccccc} CycD & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ Rb & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ E2F & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ CycE & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ CycA & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ p27 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Cdc20 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ Cdh1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ UbcH10 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ CycB & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}$$

each of these two attractors attracting 50% of the $\boldsymbol{x} \in S$ under $\boldsymbol{f}_{patho}$. It should be noted that $a_4 = a_1 \in A_{physio}$: a_4 is a physiological attractor which also belongs to A_{patho}. Indeed, it is possible that the pathological variant exhibits physiological attractors: A_{patho} is not the set containing exactly all the pathological attractors, it is the attractor set of the pathological variant, so $A_{physio} \cap A_{patho} \neq \emptyset$ is possible. However, $a_3 \notin A_{physio}$: it is a pathological attractor and is what a therapeutic bullet, being golden or silver, is intended to remove.

Again to better illustrate what is performed to obtain these therapeutic bullets, below is A_{patho} under the third bullet:

$CycD$	0
Rb	1
$E2F$	0
$CycE$	0
$CycA$	0
$p27$	1
$Cdc20$	0
$Cdh1$	1
$UbcH10$	0
$CycB$	0

which is a_2. As expected for a therapeutic bullet, the pathological attractor a_3 is removed. However, the physiological attractor a_1 is not restored: the

third therapeutic bullet is silver. Consequently, with this therapeutic bullet no cell cycle occurs and the only reachable phenotype is quiescence. While disabling the cell cycle of cancer cells is beneficial, disabling the cell cycle of healthy cells is not. As mentioned above, with silver bullets this is a matter of choice.

2.3.3 Results of the case study

With the case study, $card\ S = 268\ 435\ 456$: computing attractors from all the $\boldsymbol{x} \in S$ becomes too demanding. Indeed, it should be recalled that the algorithm computes one attractor set per bullet, namely A_{patho} under the tested bullet. Consequently, $card\ D$ is set to a more reasonable value: $card\ D = 10\ 000$. Despite that $card\ D < card\ S$, it seems sufficient for the algorithm to find all the attractors, just as Rodriguez *et al* whose the computation covers the whole state space. Below are the computed attractors:

- $A_{physio} = \{a_1\}$
- $A_{patho1} = \{a_1\}$
- $A_{patho2} = \{a_1, a_2\}$, a_1 and a_2 attracting respectively 29.5% and 70.5% of the $\boldsymbol{x} \in D$ under $\boldsymbol{f}_{patho2}$

where

$$a_1 = \begin{array}{l|ll} ICL & 0 & 0 \\ FANCM & 0 & 0 \\ FAcore & 0 & 0 \\ FANCD2I & 0 & 0 \\ MUS81 & 0 & 0 \\ FANCJBRCA1 & 0 & 0 \\ XPF & 0 & 0 \\ FAN1 & 0 & 0 \\ ADD & 0 & 0 \\ DSB & 0 & 0 \\ PCNATLS & 0 & 0 \\ MRN & 0 & 0 \\ BRCA1 & 0 & 0 \\ ssDNARPA & 0 & 0 \\ FANCD1N & 0 & 0 \\ RAD51 & 0 & 0 \\ HRR & 0 & 0 \\ USP1 & 0 & 0 \\ KU & 0 & 0 \\ DNAPK & 0 & 0 \\ NHEJ & 0 & 0 \\ ATR & 0 & 0 \\ ATM & 0 & 0 \\ p53 & 0 & 0 \\ CHK1 & 0 & 0 \\ CHK2 & 0 & 0 \\ H2AX & 0 & 0 \\ CHKREC & 0 & 1 \end{array} \qquad a_2 = \begin{array}{l|l} ICL & 0 \\ FANCM & 0 \\ FAcore & 0 \\ FANCD2I & 0 \\ MUS81 & 0 \\ FANCJBRCA1 & 1 \\ XPF & 0 \\ FAN1 & 0 \\ ADD & 0 \\ DSB & 1 \\ PCNATLS & 0 \\ MRN & 1 \\ BRCA1 & 1 \\ ssDNARPA & 1 \\ FANCD1N & 0 \\ RAD51 & 0 \\ HRR & 0 \\ USP1 & 0 \\ KU & 0 \\ DNAPK & 0 \\ NHEJ & 0 \\ ATR & 1 \\ ATM & 1 \\ p53 & 1 \\ CHK1 & 1 \\ CHK2 & 1 \\ H2AX & 1 \\ CHKREC & 0 \end{array}$$

and their biological interpretation:

- a_1: cell cycle progression
- a_2: cell cycle arrest

In physiological conditions, in case of a damaged DNA, cells repair it before performing the cell cycle, or die if repair fails. Such checkpoints enable cells to ensure genomic integrity by preventing damaged DNA to be replicated and then propagated [112, 113]. Otherwise, genetic instability may appears, potentially leading to cancer [114]. The results show that the physiological variant is able to ensure genomic integrity since its unique attractor is a_1 where $ICL = DSB = ADD = 0$: DNA damages are repaired, if any, and the cell cycle can safely occur. Interestingly, the same physiological phenotype is computed for *patho1* where $A_{patho1} = A_{physio}$. This suggests that cells bearing FANCA gene null mutation are nonetheless able to repair DNA. With *patho2*, a pathological attractor appears: a_2, where $DSB = 1$. This suggests that cells bearing FANCD1/BRCA2 or FANCN/PALB2 gene null mutation are unable to repair DSBs, explaining why a_2 corresponds to cell cycle arrest: DNA remains damaged. It should be noted that $a_1 \in A_{patho2}$, suggesting that from some $\boldsymbol{x}_0$, that is under some conditions, such cells could be able to repair DNA. However, a_1 attracts only 29.5% of the $\boldsymbol{x} \in D$ under $\boldsymbol{f}_{patho2}$, indicating that the pathological phenotype associated with a_2 is the most likely.

Altogether, according to the computed attractors and their phenotypic interpretation, and limited to the scope studied by the model of Rodriguez *et al*, FANCA gene null mutation may not induce pathological phenotypes. However, with FANCD1/BRCA2 or FANCN/PALB2 gene null mutation, two phenotypes are predicted: a physiological one and a pathological one, the latter being the most likely. Therefore, the algorithm has to operate on *patho2* to find bullets able to remove the pathological attractor a_2. By comprehensively testing all the bullets made of 1 to 3 targets, the algorithm returns the following results:

	number of all possible bullets	number of therapeutic bullets
$r = 1$	56	1 (1.786%)
$r = 2$	1 512	20 (1.323%)
$r = 3$	26 208	191 (0.729%)

all therapeutic bullets being golden since $card\ A_{physio} = 1$, as demonstrated

in the *Theorem 1* page 32. A list of the computed therapeutic bullets can be found in *Appendix C* page 90. Given that in a_1, what the pathological variant is forced to reach by means of therapeutic bullets, almost all variables are valued at 0, it is unsurprising that all targets in the computed therapeutic bullets have to be inhibited, that is set to 0.

Below is the frequency of each node in the found therapeutic bullets:

node	frequency in the found therapeutic bullets
ATM	87.736%
ICL	22.170%
$BRCA1$	18.396%
DSB	11.792%
MRN	10.377%
$FANCM$	9.906%
ADD	9.906%
$FANCJBRCA1$	9.434%
$ssDNARPA$	9.434%
$FANCD1N$	9.434%
$RAD51$	9.434%
HRR	9.434%
$USP1$	9.434%
$CHK2$	9.434%
$H2AX$	9.434%
$FAcore$	8.019%
$FANCD2I$	8.019%
$FAN1$	8.019%
$p53$	8.019%
$CHK1$	8.019%
XPF	7.547%
ATR	2.358%
$MUS81$	0.943%
$PCNATLS$	0.472%
KU	0.472%
$DNAPK$	0.472%
$NHEJ$	0.472%
$CHKREC$	0%

In this case study, DNA damages such as ICLs and DSBs are the pathological events. Unsurprisingly, the algorithm suggests them to be targeted: this is a logical consequence. However, DNA damages are not biomolecules in themselves and directly targeting them by means of drugs appears senseless.

What is relevant are the biomolecules of the FA/BRCA pathway suggested as therapeutic targets. Interestingly, ATM dominates all the other candidates, predicting it to be a pivotal therapeutic target for the *patho2* condition, namely the FA/BRCA pathway bearing FANCD1/BRCA2 or FANCN/PALB2 gene null mutation, as observed in Fanconi anemia.

2.4 Discussion

Under the assumption that attractors of dynamical systems and phenotypes of biological networks are linked when the former models the latter, the results show that the algorithm succeeds in performing the proposed *in silico* target identification. It returns therapeutic bullets for a pathological variant of the mammalian cell cycle relevant in cancer and for a pathological variant modeling Fanconi anemia. Consequently, the algorithm can be used on other synchronous Boolean models of biological networks involved in diseases for *in silico* target identification. However, both the physiological and pathological variants have to be known. This can constitute a limit of the proposed methodology since not all the pathophysiologies are known.

Target identification, whether performed *in silico* or not, is a step belonging to a wider process: drug discovery. Having demonstrated a potential target *in silico*, or even *in vitro*, is far from having a medication. Further work and many years are necessary before obtaining a drug which is effective *in vivo*. For example, and among other characteristics, such a drug has to be absorbed by the organism, has to reach its target and has to be non-toxic at therapeutic dosages. Furthermore, as with any *in silico* evidence, it should be validated through wet lab experiments: there is a bridge to cross between theory and practice. For example, targeting ATM should restore a physiological running of the FA/BRCA pathway bearing FANCD1/BRCA2 or FANCN/PALB2 gene null mutation. However, if ATM operates in other pathways, targeting it may disturb them, thus potentially creating *de novo* non-physiological conditions. Nevertheless, it is expected that the algorithm is of interest for target identification.

While finding Boolean network attractors of biological networks is not

the purpose of this work, it is a necessary step which is in itself a challenging field of computational biology. Therefore, incorporating advances made in this field could be an interesting improvement. Another possible improvement could be to extend the algorithm for asynchronous Boolean networks since such models are likely to more accurately describe the dynamics of biological systems [115,116]. In biological systems, events may be subjected to stochasticity, may not occur simultaneously or may not belong to the same time scale, three points that a synchronous updating scheme does not take into account.

Yet another possible improvement could be to use a finer logic, such as multivalued logic. One of the main limitations of Boolean models is that variables can take only two values. In reality, things are not necessarily binary and variables should be able to take more values. Multivalued logic enables it in a discrete manner where variables can take a finite number of values between 0 (false) and 1 (true). For example, one can state that Rb is partly impaired rather than totally. Such a statement is not implementable with Boolean models but is with multivalued ones such as, for example, a three-valued logic where $true = 1$, $moderate = 0.5$ and $false = 0$.

Finally, considering the basin of attraction of the pathological attractors could be an interesting extension of the criterion for selecting therapeutic bullets. In that case, the therapeutic potential of bullets could be assessed by estimating their ability at reducing the basin of the pathological attractors, as performed by Fumia *et al* with their Boolean model of cancer pathways [21]. Such a criterion enables to consider the particular case where pathological attractors are removed, that is where pathological basins are reduced to the empty set, but also the other cases where pathological basins are not necessarily reduced to the empty set. Such a less restrictive selection of therapeutic bullets would enable to consider more targeting strategies for counteracting diseases.

2.5 Additional improvements

First of all, some additional definitions should be stated:

- **physiological state space**: The state space S_{physio} of the physiological variant.
- **pathological state space**: The state space S_{patho} of the pathological variant.
- **testing state space**: The state space S_{test} of the pathological variant under the effect of a bullet.
- **physiological basin**: The basin of attraction $B_{physio,i}$ of a physiological attractor $a_{physio,i}$.
- **pathological basin**: The basin of attraction $B_{patho,i}$ of a pathological attractor $a_{patho,i}$.
- n**-bullet**: A bullet made of n targets.

Among the possible improvements mentioned in the *Discussion* section page 46, two were done after publication, namely extending the algorithm for multivalued logic and considering pathological basins for selecting therapeutic bullets.

2.5.1 Multivalued logic

2.5.1.1 Background

One of the main limitations of Boolean networks is that variables can take only two values, which can be quite simplistic. Depending on what variables model, such as activity level of enzymes or abundance of gene products, considering more than two possible levels should enable models to be more realistic. Without leaving the logic-based modeling formalism, one solution is to extend Boolean logic to multivalued logic [117]. As with Boolean logic, variables of multivalued logic are discrete, their value belonging to $[\![0;1]\!]$ where 0 means false and 1 means true. With Boolean logic, only 0 and 1 can be used to valuate variables. With multivalued logic, an arbitrary finite number h of values in $[\![0;1]\!]$ can be used. Therefore, variables of multivalued

logic can model more than only two possible levels, enabling models to be more realistic than those based on Boolean logic.

2.5.1.2 Methods

Boolean logic can be seen as a particular case of multivalued logic: it is a bivalued logic where variables take their value in $\{0, 1\}$. While Boolean operators work well in this case, multivalued logic requires suitable logical operators to be introduced. One solution is to use a mathematical formulation of the Boolean operators which also works with any multivalued logic, just as the Zadeh operators. These logical operators are a mathematical generalization of the Boolean ones proposed for fuzzy logic by its pioneer Lotfi Zadeh. Fuzzy logic, and particularly fuzzy operators, are described in the *Fuzzy operators* section page 65 since they constitute one of the key points of the second work of this thesis. For now, what is important is that the Zadeh operators are logical operators compliant with multivalued logic, including the Boolean one, and are the ones used in this work for extending the algorithm to multivalued logic. Their mathematical formulation is as follow:

$$\begin{aligned} AND(x, y) &= min(x, y) \\ OR(x, y) &= max(x, y) \\ NOT(x) &= 1 - x \end{aligned}$$

With a h-valued logic, $card\ S = h^n$. If $h = 2$ then this is the Boolean case, where $card\ S$ already raises computational difficulties. With an arbitrary $h > 2$, $card\ S$ raises even more computational difficulties. The same applies to the testable bullets since there are h^r possible c_{moda} and then $(n! \cdot h^r)/(r! \cdot (n-r)!)$ possible bullets. To illustrate how the algorithm works with a multivalued logic without overloading it, a 3-valued logic is used with $\{0, 0.5, 1\}$ as domain of value: $x_i(k) \in \{0, 0.5, 1\}$. 0 and 1 have the same meaning as in Boolean logic, namely false and true respectively. 0.5 is an intermediate truth degree which can be seen as an intermediate level of activity

or abundance, depending on what is modeled. Consequently, $S = \{0, 0.5, 1\}^n$ implying $\boldsymbol{x}_0, \boldsymbol{x}(k) \in \{0, 0.5, 1\}^n$, $D \subseteq \{0, 0.5, 1\}^n$ and $moda_i \in \{0, 0.5, 1\}$. Moreover, the Boolean operators of the f_i are replaced by the Zadeh operators. This results in the following minor changes in the pseudocode of the algorithm described in *Appendix A* page 85:

line	Boolean logic	h-valued logic
2	$card\ D = min(card\ D, 2^n)$	$card\ D = min(card\ D, h^n)$
29	$max^r_{moda} = min(max_{moda}, 2^r)$	$max^r_{moda} = min(max_{moda}, h^r)$

How the algorithm works with this 3-valued logic is illustrated with the example network, whose the logical functions become:

$$
\begin{aligned}
CycD_+ &= CycD \\
Rb_+ &= max(min(1-CycD, 1-CycE, 1-CycA, 1-CycB), \\
&\quad min(p27, 1-CycD, 1-CycB)) \\
E2F_+ &= max(min(1-Rb, 1-CycA, 1-CycB), min(p27, 1-Rb, 1-CycB)) \\
CycE_+ &= min(E2F, 1-Rb) \\
CycA_+ &= max(min(E2F, 1-Rb, 1-Cdc20, 1-min(Cdh1, UbcH10)), \\
&\quad min(CycA, 1-Rb, 1-Cdc20, 1-min(Cdh1, UbcH10))) \\
p27_+ &= max(min(1-CycD, 1-CycE, 1-CycA, 1-CycB), \\
&\quad min(p27, 1-min(CycE, CycA), 1-CycB, 1-CycD)) \\
Cdc20_+ &= CycB \\
Cdh1_+ &= max(min(1-CycA, 1-CycB), Cdc20, min(p27, 1-CycB)) \\
UbcH10_+ &= max(1-Cdh1, min(Cdh1, UbcH10, max(Cdc20, CycA, CycB))) \\
CycB_+ &= min(1-Cdc20, 1-Cdh1)
\end{aligned}
$$

which is $\boldsymbol{f}_{physio}$. For $\boldsymbol{f}_{patho}$, owing to this 3-valued logic, a constitutive but partial inactivation of Rb is simulated. Its corresponding f_i becomes:

$$Rb_+ = 0.5$$

in $\boldsymbol{f}_{patho}$.

2.5.1.3 Results

With the example network modeled by this 3-valued logic, $card\ S = 59\ 049$, which remains computationally tractable. Therefore, $card\ D = card\ S$: all the attractors are found. With the physiological variant, the algorithm returns:

$$A_{physio} = \{a_{physio1}, a_{physio2}, a_{physio3}, a_{physio4}, a_{physio5}, a_{physio6}\}$$

where

$$a_{physio1} = \begin{array}{l|l} CycD & 0 \\ Rb & 0.5 \\ E2F & 0.5 \\ CycE & 0.5 \\ CycA & 0.5 \\ p27 & 0.5 \\ Cdc20 & 0.5 \\ Cdh1 & 0.5 \\ UbcH10 & 0.5 \\ CycB & 0.5 \end{array} \qquad a_{physio2} = \begin{array}{l|l} CycD & 0 \\ Rb & 1 \\ E2F & 0 \\ CycE & 0 \\ CycA & 0 \\ p27 & 1 \\ Cdc20 & 0 \\ Cdh1 & 1 \\ UbcH10 & 0 \\ CycB & 0 \end{array}$$

$$a_{physio3} = \begin{array}{l|l} CycD & 0.5 \\ Rb & 0.5 \\ E2F & 0.5 \\ CycE & 0.5 \\ CycA & 0.5 \\ p27 & 0.5 \\ Cdc20 & 0.5 \\ Cdh1 & 0.5 \\ UbcH10 & 0.5 \\ CycB & 0.5 \end{array} \qquad a_{physio4} = \begin{array}{l|l} CycD & 1 \\ Rb & 0 \\ E2F & 0.5 \\ CycE & 0.5 \\ CycA & 0.5 \\ p27 & 0 \\ Cdc20 & 0.5 \\ Cdh1 & 0.5 \\ UbcH10 & 0.5 \\ CycB & 0.5 \end{array}$$

$$a_{physio5} = \begin{array}{l|ll} CycD & 0 & 0 \\ Rb & 0.5 & 1 \\ E2F & 0 & 0.5 \\ CycE & 0 & 0 \\ CycA & 0 & 0 \\ p27 & 0.5 & 1 \\ Cdc20 & 0.5 & 0 \\ Cdh1 & 0.5 & 1 \\ UbcH10 & 0.5 & 0.5 \\ CycB & 0 & 0.5 \end{array}$$

$$a_{physio6} = \begin{array}{l|lllllll} CycD & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ Rb & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ E2F & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ CycE & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ CycA & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ p27 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ Cdc20 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ Cdh1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ UbcH10 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ CycB & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{array}$$

with their corresponding basin of attraction:

a_i	B_i (in % of $card\ S_{physio}$)
$a_{physio1}$	9.9%
$a_{physio2}$	20.1%
$a_{physio3}$	33.3%
$a_{physio4}$	24.5%
$a_{physio5}$	3.4%
$a_{physio6}$	8.8%

It should be noted that $a_{physio2}$ and $a_{physio6}$ are the two physiological attractors found in the Boolean case. Indeed, since $\{0,1\} \subset \{0,0.5,1\}$ and since the Zadeh operators also work with Boolean logic, Boolean logic is included

in this three-valued logic. This means that results obtainable with the former are also obtainable with the latter. With the pathological variant, where Rb is constitutively but partially inactivated, the algorithm returns:

$$A_{patho} = \{a_{physio1}, a_{physio3}, a_{patho1}\}$$

where

$$a_{patho1} = \begin{array}{l|l} CycD & 1 \\ Rb & 0.5 \\ E2F & 0.5 \\ CycE & 0.5 \\ CycA & 0.5 \\ p27 & 0 \\ Cdc20 & 0.5 \\ Cdh1 & 0.5 \\ UbcH10 & 0.5 \\ CycB & 0.5 \end{array}$$

with their corresponding basin of attraction:

a_i	B_i (in % of $card\ S_{patho}$)
$a_{physio1}$	33.3%
$a_{physio3}$	33.3%
a_{patho1}	33.3%

Only $a_{physio1}$ and $a_{physio3}$ remain, while a_{patho1} appears and is what therapeutic bullets have to remove from S_{test}.

As in the Boolean case, the algorithm is launched with $r_{min} = 1$ and $r_{max} = 2$. max_{targ} and max_{moda} are set to their maximum, namely $max_{targ} = 45$ and $max_{moda} = 9$: all the $1, 2$-bullets are tested. The algorithm returns the following therapeutic bullets:

$CycD[0]$		silver
$CycD[0.5]$		silver
$CycD[0]$	$Rb[0.5]$	silver
$CycD[0.5]$	$Rb[0.5]$	silver
$CycD[1]$	$Rb[0]$	silver
$CycD[0]$	$E2F[0.5]$	silver
$CycD[0.5]$	$E2F[0.5]$	silver
$CycD[0]$	$CycE[0.5]$	silver
$CycD[0.5]$	$CycE[0.5]$	silver
$CycD[0]$	$CycA[0.5]$	silver
$CycD[0.5]$	$CycA[0.5]$	silver
$CycD[0]$	$p27[0.5]$	silver
$CycD[0.5]$	$p27[0.5]$	silver
$CycD[0]$	$Cdc20[0.5]$	silver
$CycD[0.5]$	$Cdc20[0.5]$	silver
$CycD[0]$	$Cdh1[0.5]$	silver
$CycD[0.5]$	$Cdh1[0.5]$	silver
$CycD[0]$	$UbcH10[0.5]$	silver
$CycD[0.5]$	$UbcH10[0.5]$	silver
$CycD[0]$	$CycB[0.5]$	silver
$CycD[0.5]$	$CycB[0.5]$	silver

where $X[y]$ means that the node $X \in V$ has to be set to the value $y \in \{0, 0.5, 1\}$. For example, the third therapeutic bullet is made of the targets CycD and Rb whose the value has to be set to 0 and 0.5 respectively. As in the Boolean case, it should be noted that no golden bullets are found, an unsurprising result since they are rarer than silver ones.

2.5.1.4 Discussion

The algorithm is now extended for multivalued logic, which includes the Boolean one. This means that the previous strictly Boolean version of the algorithm is included in this new one. Moreover, allowing variables to take an arbitrary finite number of values should enable to more accurately model bio-

logical processes and produce more fine-tuned therapeutic bullets. However, this accuracy and fine-tuning are at the cost of an increased computational requirement. Indeed, in this work, the computational requirement essentially depends on the cardinality of the state space, which itself depends on the size of the model and the used multivalued logic. Therefore, the size of the model and the used multivalued logic should be balanced: the smaller the model is, the more variables should be finely valued. For example, for a fine therapeutic investigation, the model should only contain the essential and specific pieces of the pathophysiology of interest, modeled by a finely valued logic. On the other hand, for a gross therapeutic investigation, an exhaustive model could be used but modeled by a coarse-grained logic, such as the Boolean one. Finally, it should be noted that the ultimate multivalued logic is the infinitely valued one, which is fuzzy logic. With fuzzy logic, the whole $[0; 1] \subset \mathbb{R}$ is used to valuate variables, which should bring the best accuracy for the qualitative modeling formalism. Using fuzzy logic to qualitatively model biological networks is addressed in the second work of this thesis.

2.5.2 Therapeutic bullet assessment

2.5.2.1 Background

Till now, the algorithm requires therapeutic bullets to remove all the pathological attractors from the pathological state space, so that the pathological variant no longer exhibits pathological phenotypes. This criterion for selecting therapeutic bullets can appear somewhat drastic since it is all or nothing. A less strict criterion should enable to consider more targeting strategies, and then more possibilities for counteracting diseases. Certainly, a less restrictive criterion could bring less "powerful" therapeutic bullets, but being too demanding potentially leads to no results and loss of nonetheless interesting findings.

The therapeutic potential of bullets could be assessed by estimating their ability at reducing the cardinality of the pathological basins. This is a more permissive criterion since therapeutic bullets no longer have to necessarily

remove the pathological attractors. Reducing the cardinality of a pathological basin renders the corresponding pathological attractor less reachable, and then the associated pathological phenotype less likely. This new criterion includes the previous one: removing an attractor means reducing its basin of attraction to the empty set. Therefore, therapeutic bullets obtainable with the previous criterion are also obtainable with this new one.

2.5.2.2 Methods

To implement this new criterion for selecting therapeutic bullets, the algorithm considers a bullet as therapeutic if it increases $card \bigcup B_{physio,i}$ in S_{test} without creating *de novo* attractors. Since the attractors are either physiological or pathological, increasing $card \bigcup B_{physio,i}$ is equivalent to decreasing $card \bigcup B_{patho,i}$. The goal of this new criterion is to increase the physiological part of S_{test}, which is equivalent to decreasing its pathological part. Consequently, a pathological variant treated by such a therapeutic bullet tends to, but not necessarily reaches, an overall physiological behavior. However, as with the previous criterion, it does not ensure that all the $a_{physio,i}$ are preserved/restored. *A fortiori*, it does not ensure that the $B_{physio,i}$ in S_{test} are as in S_{physio}. This means that it does not ensure that the reachability of the $a_{physio,i}$ is preserved/restored. Nevertheless, as with the previous criterion, this is a matter of choice between a therapeutic bullet or not. To assist this choice and better visualize the effects of therapeutic bullets, the $card\ B_{physio,i}$ and $card\ B_{patho,i}$ in S_{test} are computed.

Implementing this new criterion for selecting therapeutic bullets is a major change. Therefore, the pseudocode of the algorithm presented in *Appendix A* page 85 is rewritten and structured into three modules:

- the *compute_A* function, which computes A_{physio} or A_{patho}, depending on which of the $\boldsymbol{f}_{physio}$ or $\boldsymbol{f}_{patho}$ is passed
- the *compute_cover* function, which for two attractor sets A_1 and A_2 computes the covering of S_2 by $\bigcup B_{1,i}$, expressed in percents of $card\ S_2$
- the *compute_T* function, which computes a set T of therapeutic bullets

Below is the corresponding pseudocode:

function $A = compute\_A(\boldsymbol{f}, c_{targ}, c_{moda}, D, V)$

```
1  A = {}
2  for x_0 ∈ D do
3      k = 1
4      x(k) = x_0
5      while true do
6          x(k+1) = f(x(k))
7          for targ_i ∈ c_targ do
8              for v_j ∈ V do
9                  if v_j = targ_i then
10                     x_j(k+1) = moda_i
11                 end if
12             end for
13         end for
14         if ∃k' ∈ [[1, k]]: x(k+1) = x(k') then
15             a_i.seq = (x(k'), ..., x(k))
16             if ∃a_j ∈ A: a_i.seq = a_j.seq then
17                 a_j.freq = a_j.freq + 1
18             else
19                 a_i.freq = 1
20                 A = A ∪ {a_i}
21             end if
22             break
23         end if
24         k = k + 1
25     end while
26 end for
27 for a ∈ A do
28     a.freq = a.freq · 100/card D
29 end for
30 return A
```

end function

For A_{physio} and A_{patho}, which are computed without bullet, the empty bullet $((), ())$ has to be passed. The a_i are represented as structures composed of two fields: $a_i.seq$, which is the sequence of a_i (line 15), and $a_i.freq$, which is the corresponding *card* B_i, expressed in percent of *card* D. To compute $a_i.freq$, the algorithm counts the number of times a_i is reached (line 19 if this is the first time a_i is reached, line 17 otherwise) and then, once all the $x_0 \in D$ are computed, translates $a_i.freq$ in percent of *card* D (line 28).

function $y = compute\_cover(A_1, A_2)$

```
1  cover = 0
2  for a_1 ∈ A_1 do
3      if ∃a_2 ∈ A_2 : a_1.seq = a_2.seq then
4          cover = cover + a_2.freq
5      end if
6  end for
7  return cover
```

end function

If a_1 also belongs to A_2 (line 3) then the cardinality of its basin in S_2 is used to compute the covering of S_2 by $\bigcup B_{1,i}$ (line 4).

function $T = compute\_T(\boldsymbol{f}_{physio}, \boldsymbol{f}_{patho}, r_{min}, r_{max}, max_{targ}, max_{moda}, max_D, h, V)$

```
1  n = card V
2  D = {}
3  while card D < max_D do
4      generate x_0 ∉ D
5      D = D ∪ {x_0}
6  end while
7  A_physio = compute_A(f_physio, (), (), D, V)
8  A_patho = compute_A(f_patho, (), (), D, V)
9  T = {}
10 cover_patho = compute_cover(A_physio, A_patho)
11 for r ∈ [[r_min, r_max]] do
```

12 $C_{targ} = \{\}$
13 $C_{moda} = \{\}$
14 **while** *card* $C_{targ} < min(max_{targ}, n!/(r! \cdot (n-r)!))$ **do**
15 generate $c_{targ} \notin C_{targ}$
16 $C_{targ} = C_{targ} \cup \{c_{targ}\}$
17 **end while**
18 **while** *card* $C_{moda} < min(max_{moda}, h^r)$ **do**
19 generate $c_{moda} \notin C_{moda}$
20 $C_{moda} = C_{moda} \cup \{c_{moda}\}$
21 **end while**
22 **for** $c_{targ} \in C_{targ}$ **do**
23 **for** $c_{moda} \in C_{moda}$ **do**
24 $A_{test} = compute\_A(\boldsymbol{f}_{patho}, c_{targ}, c_{moda}, D, V)$
25 **if** $A_{test} \subseteq A_{physio} \cup A_{patho}$ **then**
26 $cover_{test} = compute\_cover(A_{physio}, A_{test})$
27 **if** $cover_{test} > cover_{patho}$ **then**
28 $T = T \cup \{(c_{targ}, c_{moda})\}$
29 **end if**
30 **end if**
31 **end for**
32 **end for**
33 **end for**
34 **return** T
end function

max_D is the desired *card* D and h is the cardinality of the domain of value, which depends on the used multivalued logic. A_{physio} and A_{patho} are computed without bullet, so the empty bullet $((),())$ is passed to *compute_A* (lines 7 and 8). $cover_{patho}$ is the covering of S_{patho} by $\bigcup B_{physio,i}$ (line 10) and $cover_{test}$ is the covering of S_{test} by $\bigcup B_{physio,i}$ (line 26). A_{test} is the pathological attractor set under the effect of the tested bullet (line 24). A therapeutic bullet has to avoid the appearance of *de novo* attractors (line 25) and has to increase the covering of S_{test} by $\bigcup B_{physio,i}$ (line 27).

2.5.2.3 Results

This new criterion for selecting therapeutic bullets is illustrated on the case study modeled by Boolean logic: $h = 2$. Since *patho1* does not produce pathological attractors, only *patho2* is computed. As previously, wholly computing S is too demanding. Therefore, D is intended to have a reasonable cardinality: $max_D = 100\ 000$. All the $1,2$-bullets are tested: $r_{min} = 1$, $r_{max} = 2$, $max_{targ} = 378$ and $max_{moda} = 4$. However, their therapeutic potential is no longer expressed as golden or silver but by their gain. It is displayed as follow: $x\% \rightarrow y\%$ where $card \bigcup B_{physio,i} = x\%$ in S_{patho} and $y\%$ in S_{test}. Consequently, in order to increase the physiological part of S_{test}, a therapeutic bullet has to make $y > x$. The $card\ B_{physio,i}$ and $card\ B_{patho,i}$ in S_{test} are also computed and expressed in percent of $card\ S_{test}$. The algorithm returns 59 therapeutic bullets whose the list can be found in *Appendix D* page 97.

A therapeutic bullet as defined by the previous criterion, that is which removes all the $a_{patho,i}$ from S_{test}, makes *de facto* $card \bigcup B_{physio,i} = 100\%$ in S_{test}. As already mentioned, the previous criterion is included in this new one: therapeutic bullets obtainable with the former are also obtainable with the latter. This can be checked by noting that the $1,2$-therapeutic bullets found with the previous criterion are also found with this new one.

With this case study, $A_{physio} = \{a_{physio1}\}$, so $\bigcup B_{physio,i} = B_{physio1}$. Therefore, in this particular case where $card\ A_{physio} = 1$, therapeutic bullets have to increase $card\ B_{physio1}$ in S_{test}. It should be recalled that $card\ B_{physio1} = 29.4\%$ in S_{patho}, so therapeutic bullets have to make $card\ B_{physio1} > 29.4\%$ in S_{test}. For example, below are the computed 1-therapeutic bullets:

bullet	gain			$B_{physio1}$	B_{patho1}
$-FANCM$	29.4%	$\rightarrow$	44.6%	44.6%	55.4%
$-FANCD2I$	29.4%	$\rightarrow$	30.4%	30.4%	69.6%
$-XPF$	29.4%	$\rightarrow$	46.2%	46.2%	53.8%
$-FAN1$	29.4%	$\rightarrow$	32.9%	32.9%	67.1%
$-ATM$	29.4%	$\rightarrow$	100%	100%	0%

$-ATM$ is a therapeutic bullet also found with the previous criterion since

it removes all the $a_{patho,i}$, namely a_{patho1}, from S_{test}. However, the other four therapeutic bullets are only obtainable with this new criterion since they do not remove a_{patho1} from S_{test}. Nevertheless, as therapeutic bullets, they increase $card\ B_{physio1}$ in S_{test}. This highlight the ability of this new criterion to unravel more therapeutic bullets of varying therapeutic potential, thus opening the way for more targeting strategies of varying theoretical efficacy. Of course, therapeutic bullets of poor potential are also unraveled, such as $-FANCD2I$ which only increases $card\ B_{physio1}$ from 29.4% in S_{patho} to 30.4% in S_{test}. However, *in silico* tools should not restrict their predictions to only those exhibiting a high theoretical potency since predicted does not necessarily mean true. Indeed, a prediction of apparently poor interest can reveal itself of great interest in practice, and *vice versa*.

2.5.2.4 Discussion

The algorithm now uses a new criterion for selecting therapeutic bullets which brings a wider range of targeting strategies of varying predicted efficacy. Moreover, no information is lost from the previous criterion since results obtainable with the previous one are also obtainable with this new one. This new criterion is based on a more permissive assumption stating that reducing the reachability of pathological attractors is therapeutic. For an *in silico* tool such as this algorithm, a more permissive assumption is important since theoretical findings have to outlive the bottleneck separating prediction to reality. Indeed, results predicted *in silico* have to be validated *in vitro* and/ or *in vivo*. Therefore, requiring only perfect predictions such as therapeutic bullets removing all the pathological attractors could left insufficient results after validation. All the more so that a prediction of apparently poor interest could reveal itself as an insight of great interest and *vice versa*, hence the necessity of obtaining a wide range of theoretical findings.

This new criterion for selecting therapeutic bullets also brings a finer assessment of their potential since all the percentages between $card \bigcup B_{physio,i}$ in S_{patho} and 100% are considered. With the previous criterion, the only therapeutic potential is $card \bigcup B_{physio,i} = 100\%$ in S_{test}, thus reducing the

assessment to therapeutic or not. However, things are not necessarily black or white but rather a continuum of gray nuances, so the assessment of therapeutic potentials should be nuanced too, just as it is now.

Chapter 3

Enhancing Boolean networks with fuzzy operators and edge tuning

This chapter describes the second work of this thesis, mostly as prepublished.

3.1 Background

This work is an extension of the Boolean network formalism aimed at enhancing it. The basic principles remain the same: given a biological network, entities are modeled by variables and their interactions by functions allowing their value to be updated at each iteration of the simulation. However, Boolean operators are replaced by the operators of fuzzy logic [118, 119], allowing variables to be valued at any real number between 0 and 1, that is to consider all the possible truth degrees between the absolutely true and the definitively false. Therefore, results obtainable with fuzzy operators, while remaining qualitative, can be finer than those obtainable with Boolean operators. In some cases, the ON/OFF nature of Boolean logic is a relevant choice, as for example with gene regulatory networks where gene expression level can be approximated by Boolean states. However, in some other cases where things are not necessarily binary, such as in signaling pathways where enzymes can be more or less active, using fuzzy operators can be an interesting choice.

In addition of using fuzzy operators, some additional features are introduced in order to capture more behavioral aspects of biological networks. These additional features concern the edges of the network, which are seen as conveyors of signals corresponding to influences exerted by entities of the network onto other ones. This signal, together with its modulation, are taken into account so that edges can be tuned. To do so, edge states are computed and the signal they convey can be slowed or weakened. This results in a qualitative modeling approach intended to bring a fine qualitative quantification of biological network behaviors.

Talking about a qualitative quantification can appear somewhat contradictory but is common in thinking processes, which are at the basis of any scientific reasoning. Simple examples of such qualitative quantification could be to state that an enzyme is more active than another one, or to state that an enzyme is moderately active: quantification is expressed by perceptions and tendencies. Indeed, qualitative quantification is expressed by words rather than measurements, hence its qualitative nature, and is characteristic of fuzzy logic [120, 121].

Fuzzy logic-based modeling is a promising approach successfully developed in several works [122–127]. However, this work is not fuzzy logic-based: there are no fuzzy sets, no membership functions, no degrees of membership and no fuzzy inference systems. Only the operators are taken from fuzzy logic to replace the Boolean ones, the goal being to enhance the Boolean network formalism by extending it to a continuous form and by adding edge tuning.

3.2 Methods

This section introduces fuzzy operators and then describes how the proposed logic-based modeling is built. An example to illustrate it, together with its implementation, are also described.

3.2.1 Fuzzy operators

The main difference between Boolean and fuzzy logic is that the former is discrete, that is valued in $[\![0;1]\!] \subset \mathbb{N}$, whereas the latter is continuous, that is valued in $[0;1] \subset \mathbb{R}$. Fuzzy logic can be seen as a generalization of Boolean logic, implying that the fuzzy counterparts of the Boolean operators have to behave like them on $[\![0;1]\!]$ but have to be defined on $[0;1]$. The generalization of the Boolean AND operator is the *t-norm*, the generalization of the Boolean OR operator is the *s-norm* and the generalization of the Boolean NOT operator is the *complement*:

$$\begin{aligned} \textit{t-norm}&: [0;1]^2 \to [0;1] : (x,y) \mapsto \textit{t-norm}(x,y) \\ \textit{s-norm}&: [0;1]^2 \to [0;1] : (x,y) \mapsto \textit{s-norm}(x,y) \\ \textit{complement}&: [0;1] \to [0;1] : x \mapsto \textit{complement}(x) \end{aligned}$$

where $x, y \in [0;1]$. There exist different mathematical formulations of the *t-norm*, *s-norm* and *complement*, all fulfilling the rules of Boolean algebra [128] but defined on $[0;1]$. For convenience, both the Boolean and fuzzy operators can be named AND, OR and NOT, the context specifying which of them is referred to.

Due to the ability of fuzzy operators to be continuous, variables can take their value in $[0;1]$. Therefore, they can be equal to 1 (true), 0 (false) or all the other real numbers of $[0;1]$ (more or less true): all the truth degrees between true and false are considered. This could be more realistic in a world where things are not necessarily binary. For example, a Boolean model of a signaling pathway allows enzymes to be ON or OFF and nothing between. However, one can expect that an enzyme is allowed to be in an intermediate activity level, an expectation not implementable with Boolean models but which is with fuzzy ones. Whatever the truth degrees represent, using fuzzy operators enables to consider all the intermediate levels of what is modeled without leaving the qualitative modeling formalism.

3.2.2 The proposed logic-based modeling

First of all, it should be mentioned that a distinction is made between quantitative and qualitative parameters, this distinction residing in what parameters translate. A quantitative parameter translates a quantification obtained by experimental measurements whereas a qualitative parameter translates a perception by means of truth degrees. For example, regarding the velocity of a biochemical reaction, "slow" could be expressed by the truth degree 0.2 whereas "fast" by 0.8: this is the truth degree of the statement "This biochemical reaction is fast.". Unlike an experimental quantification which is *de facto* objective, a perception is subjective, so the same applies to its associated truth degree. Incorporating qualitative parameters should not yield the scarcity of parameter values encountered in quantitative modeling since qualitative information is relatively easy to obtain.

To build the proposed logic-based modeling from Boolean networks, the Boolean operators AND, OR and NOT have to be replaced by the fuzzy operators *t-norm*, *s-norm* and *complement*. Furthermore, the initial states $x_i(k_0)$ of the x_i have to belong to $[0;1]$. As a consequence, the value of the x_i belongs to $[0;1]$: $x_i(k) \in [0;1]$, the f_i become functions from $[0;1]^n$ to $[0;1]$:

$$f_i \colon [0;1]^n \to [0;1] \colon \boldsymbol{x} \mapsto f_i(\boldsymbol{x})$$

the value of $\boldsymbol{x}$ and $\boldsymbol{x}_0$ belongs to $[0;1]^n$: $\boldsymbol{x}(k), \boldsymbol{x}_0 \in [0;1]^n$ and $\boldsymbol{f}$ becomes a function from $[0;1]^n$ onto itself:

$$\boldsymbol{f} \colon [0;1]^n \to [0;1]^n \colon \boldsymbol{x} \mapsto \boldsymbol{f}(\boldsymbol{x})$$

Finally, additional features are added in order to capture more behavioral aspects of biological networks. These features concern the edges and are presented separately for the sake of clarity before being integrated all together.

3.2.2.1 Edge computation

As with node states, edge states are computed. For convenience, edges can be notated e_{ij} instead of (x_i, x_j). An edge e_{ij} is seen as a channel conveying the signal sent by its source x_i to its target x_j which uses it to compute its state thanks to f_j. Practically, e_{ij} conveys the value $x_i(k)$ of x_i to x_j and then f_j uses it to compute $x_j(k+1)$. This is implicitly done in Boolean networks where $x_j(k+1) = f_j(\dots, x_i(k), \dots)$ but, in this work, this is made explicit in order to modulate the signal conveyed by the edges. Consequently, the f_j no longer directly accept the $x_i(k)$ as arguments but accept the $e_{ij}(k)$. Since e_{ij} conveys $x_i(k)$, its value $e_{ij}(k)$ should be $x_i(k)$, but this is where additional features are added. Indeed, a function f_{ij}^{edge} is attributed to each e_{ij}:

$$e_{ij}(k+1) = f_{ij}^{edge}(x_i(k), e_{ij}(k))$$

It should be noted that, in addition to the value $x_i(k)$ of the source x_i, f_{ij}^{edge} also takes as argument the value $e_{ij}(k)$ of e_{ij} itself. This is required for the additional feature *edge reactivity* described below. As mentioned above, the f_j have now to accept the $e_{ij}(k)$ instead of the $x_i(k)$. For convenience, the f_j are renamed f_j^{node}:

$$x_j(k+1) = f_j^{node}(\boldsymbol{e}(k))$$

where $\boldsymbol{e} = (\dots, e_{ij}, \dots)$ is the counterpart of $\boldsymbol{x} = (\dots, x_i, \dots)$, namely the state vector of the edges, its value at the iteration k being $\boldsymbol{e}(k) = (\dots, e_{ij}(k), \dots)$. Consequently, $\boldsymbol{f}$ becomes $\boldsymbol{f}^{node} = (\dots, f_i^{node}, \dots)$:

$$\boldsymbol{x}(k+1) = \boldsymbol{f}^{node}(\boldsymbol{e}(k))$$

and its counterpart the transition function of the edges $\boldsymbol{f}^{edge} = (\dots, f_{ij}^{edge}, \dots)$ is introduced:

$$\boldsymbol{e}(k+1) = \boldsymbol{f}^{edge}(\boldsymbol{x}(k), \boldsymbol{e}(k))$$

On the basis of the updating scheme of synchronous Boolean networks, the computation becomes:

1 **for** $k \in [\![k_0, k_{end} - 1]\!]$ **do**
2 $\quad \ldots$
3 $\quad e_{ij}(k+1) = f_{ij}^{edge}(x_i(k), e_{ij}(k))$
4 $\quad \ldots$
5 $\quad x_i(k+1) = f_i^{node}(\ldots, e_{ij}(k), \ldots)$
6 $\quad \ldots$
7 **end for**

which can be written in a more concise form:

1 **for** $k \in [\![k_0, k_{end} - 1]\!]$ **do**
2 $\quad \boldsymbol{e}(k+1) = \boldsymbol{f}^{edge}(\boldsymbol{x}(k), \boldsymbol{e}(k))$
3 $\quad \boldsymbol{x}(k+1) = \boldsymbol{f}^{node}(\boldsymbol{e}(k))$
4 **end for**

3.2.2.2 Edge reactivity

The additional feature *edge reactivity* is implemented by a qualitative parameter $p_{ij} \in [0; 1]$ attributed to each e_{ij}. p_{ij} is the portion of the signal conveyed by e_{ij} which is updated at each k, namely the portion of the value $e_{ij}(k)$ which is updated to $x_i(k)$:

$$e_{ij}(k+1) = (1 - p_{ij}) \cdot e_{ij}(k) + p_{ij} \cdot x_i(k)$$

The higher p_{ij} is, the higher is the portion of $e_{ij}(k)$ which is updated: a highly reactive edge has a p_{ij} close to 1 whereas a faintly reactive edge has a p_{ij} close to 0. Biologically, *edge reactivity* can take into account that some biological interactions can be slower, or of higher inertia, than other ones. For example, an edge modeling a gene expression activation of a gene product

by a transcription factor should have a lower p_{ij} than an edge modeling an activating phosphorylation of an enzyme by another one. Indeed, gene expression is a complex mechanism involving several steps and then takes more time to be accomplished and terminated than a phosphorylation.

3.2.2.3 Edge weakening

The additional feature *edge weakening* is implemented by a qualitative parameter $q_{ij} \in [0;1]$ attributed to each e_{ij}. q_{ij} is a weakening coefficient applied at each k to the signal conveyed by e_{ij}, that is to $x_i(k)$:

$$e_{ij}(k+1) = q_{ij} \cdot x_i(k)$$

The higher q_{ij} is, the lower is the weakening of the signal $x_i(k)$ conveyed by e_{ij}: a strong edge has a q_{ij} close to 1 whereas a weak edge has a q_{ij} close to 0. Biologically, *edge weakening* can take into account that some biological interactions can be weaker than other ones. For example, given a receptor, an edge modeling its activation by a partial agonist should have a lower q_{ij} than an edge modeling its activation by a full agonist.

3.2.2.4 Combining the all

Edge reactivity and *edge weakening* are described separately for the sake of clarity but are both computed at each iteration:

$$e_{ij}(k+1) = (1 - p_{ij}) \cdot e_{ij}(k) + p_{ij} \cdot q_{ij} \cdot x_i(k)$$

hence the mathematical formulation of the f_{ij}^{edge}:

$$f_{ij}^{edge}(x_i, e_{ij}) = (1 - p_{ij}) \cdot e_{ij} + p_{ij} \cdot q_{ij} \cdot x_i$$

3.2.3 Implementation

In this work, k is not the time, it only represents the iterations performed during a run. Although quantifying time through k is possible, here the goal is to visualize sequences of events linked by causal connections without time quantification. To do so, $k_0 = 1$ and $k_{end} = 50$: 49 iterations are performed during a run. Furthermore, the initial state $e_{ij}(k_0)$ of each e_{ij} is assumed to be equal to the initial state $x_i(k_0)$ of its source x_i: $e_{ij}(k_0) = x_i(k_0)$. To illustrate the proposed logic-based modeling, it is implemented on an example with GNU Octave[1]. The code is available on GitHub[2] at `https://github.com/arnaudporet/smoosim`.

3.2.3.1 Example

The used example is a tiny sample of the epidermal growth factor receptor signaling pathway [129] adapted from [18]. It is chosen for its simplicity so that it can be mentally computed in order to easily judge the produced results. A digital electronic representation is shown in *Figure 3.1* page 72. Below are the corresponding Boolean functions where AND, NOT and OR stand for the Boolean operators:

$$
\begin{aligned}
EGF(k+1) &= \text{input set manually} \\
HRG(k+1) &= \text{input set manually} \\
EGFR(k+1) &= OR(EGF(k), HRG(k)) \\
PI3K(k+1) &= AND(EGFR(k), NOT(ERK(k))) \\
AKT(k+1) &= PI3K(k) \\
Raf(k+1) &= OR(EGFR(k), AKT(k)) \\
ERK(k+1) &= Raf(k)
\end{aligned}
$$

By applying the above-described methodology, below are the obtained f_{ij}^{edge} and f_i^{node} where AND, NOT and OR stand for the fuzzy operators:

[1] `http://www.gnu.org/software/octave/`
[2] `https://github.com/`

$$
\begin{aligned}
(EGF, EGFR)(k+1) &= (1 - p_{EGF,EGFR}) \cdot (EGF, EGFR)(k) \\
&\quad + p_{EGF,EGFR} \cdot q_{EGF,EGFR} \cdot EGF(k) \\
(HRG, EGFR)(k+1) &= (1 - p_{HRG,EGFR}) \cdot (HRG, EGFR)(k) \\
&\quad + p_{HRG,EGFR} \cdot q_{HRG,EGFR} \cdot HRG(k) \\
(EGFR, PI3K)(k+1) &= (1 - p_{EGFR,PI3K}) \cdot (EGFR, PI3K)(k) \\
&\quad + p_{EGFR,PI3K} \cdot q_{EGFR,PI3K} \cdot EGFR(k) \\
(ERK, PI3K)(k+1) &= (1 - p_{ERK,PI3K}) \cdot (ERK, PI3K)(k) \\
&\quad + p_{ERK,PI3K} \cdot q_{ERK,PI3K} \cdot ERK(k) \\
(PI3K, AKT)(k+1) &= (1 - p_{PI3K,AKT}) \cdot (PI3K, AKT)(k) \\
&\quad + p_{PI3K,AKT} \cdot q_{PI3K,AKT} \cdot PI3K(k) \\
(EGFR, Raf)(k+1) &= (1 - p_{EGFR,Raf}) \cdot (EGFR, Raf)(k) \\
&\quad + p_{EGFR,Raf} \cdot q_{EGFR,Raf} \cdot EGFR(k) \\
(AKT, Raf)(k+1) &= (1 - p_{AKT,Raf}) \cdot (AKT, Raf)(k) \\
&\quad + p_{AKT,Raf} \cdot q_{AKT,Raf} \cdot AKT(k) \\
(Raf, ERK)(k+1) &= (1 - p_{Raf,ERK}) \cdot (Raf, ERK)(k) \\
&\quad + p_{Raf,ERK} \cdot q_{Raf,ERK} \cdot Raf(k)
\end{aligned}
$$

$$
\begin{aligned}
EGF(k+1) &= \text{input set manually} \\
HRG(k+1) &= \text{input set manually} \\
EGFR(k+1) &= OR((EGF, EGFR)(k), (HRG, EGFR)(k)) \\
PI3K(k+1) &= AND((EGFR, PI3K)(k), NOT((ERK, PI3K)(k))) \\
AKT(k+1) &= (PI3K, AKT)(k) \\
Raf(k+1) &= OR((EGFR, Raf)(k), (AKT, Raf)(k)) \\
ERK(k+1) &= (Raf, ERK)(k)
\end{aligned}
$$

It should be noted that f_{EGF}^{node} and f_{HRG}^{node} do not accept any $e_{ij}(k)$ as argument. This is because they are associated with the two inputs EGF and HRG of the network and are consequently set manually.

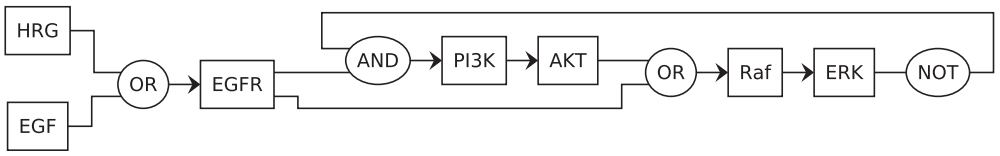

Figure 3.1 – Digital electronic representation of the example. Nodes are rectangles whereas logical gates are ellipses. This digraph should be read from left to right. For example, the node $PI3K$ is an input of the node AKT and the node ERK, due to a feedback loop, is an input of the node $PI3K$. Logical gates are not nodes and, as such, edges only pass through them. For example, the edge $(ERK, PI3K)$ passes through a NOT and AND gate whereas the edge (Raf, ERK) does not pass through any logical gate.

3.2.3.2 Fuzzy operators

As mentioned above, there exist different mathematical formulations of the fuzzy operators, all fulfilling the rules of Boolean algebra but defined on $[0; 1]$. In this work, the algebraic formulation is used:

$$\begin{aligned} AND(x, y) &= x \cdot y \\ OR(x, y) &= x + y - x \cdot y \\ NOT(x) &= 1 - x \end{aligned}$$

which is one of the most simple and convenient.

3.2.3.3 Additional features

Since $p_{ij} \in [0; 1]$, its value can be set to any real number of $[0; 1]$. However, p_{ij} is a qualitative parameter and rather than requiring to precisely valuate it as in quantitative models, its value is randomly picked in specified intervals of $[0; 1]$ from a uniform distribution. By the way, this random selection introduces a little of a rudimentary stochasticity, although introducing randomness is not the purpose of this work. To do so, $[0; 1]$ is split into intervals of truth degrees reflecting various edge reactivities:

instantaneous	$p_{ij} = 1$
faster	$p_{ij} \in [0.75; 1]$
fast	$p_{ij} \in [0.5; 0.75]$
slow	$p_{ij} \in [0.25; 0.5]$
slower	$p_{ij} \in [0; 0.25]$
down	$p_{ij} = 0$

plus the entire interval $[0; 1]$ in case of an undetermined *edge reactivity*. For example, $p_{ij} = fast$ means that the value of p_{ij} is randomly picked in $[0.5; 0.75]$ from a uniform distribution. This random selection occurs before each run and, once selected, the value of p_{ij} remains the same during the run. To better approach the behavior of the modeled biological network, replicates are made: r runs are performed and the results are superposed. In this work, $r = 10$. $q_{ij}, x_i(k_0) \in [0; 1]$ are subjected to the same replication with the following splits of $[0; 1]$:

strong	$q_{ij} = 1$
weak	$q_{ij} \in [0.75; 1]$
weaker	$q_{ij} \in [0.5; 0.75]$
faint	$q_{ij} \in [0.25; 0.5]$
fainter	$q_{ij} \in [0; 0.25]$
down	$q_{ij} = 0$

and

full	$x_i(k_0) = 1$
much more	$x_i(k_0) \in [0.75; 1]$
much	$x_i(k_0) \in [0.5; 0.75]$
few	$x_i(k_0) \in [0.25; 0.5]$
fewer	$x_i(k_0) \in [0; 0.25]$
none	$x_i(k_0) = 0$

plus the entire interval $[0; 1]$ in case of an undetermined *edge weakening* or initial state.

3.3 Results

In this section, results obtained with the example through five simulations are presented. Although the obtained curves are continuous due to the use of fuzzy operators, they are not quantitative. As qualitative results, rather than looking for numerical values, one can say, for example, that $PI3K$ is totally inhibited or that ERK is partly activated, two simple examples of qualitative quantification expressed by words and perceptions.

3.3.1 Simulation 1

EGF and HRG are the two inputs of the example and, since both can activate $EGFR$, one is sufficient to initiate the signaling cascade. It is assumed that, at the resting state, both the inputs are down: $\forall k,\ EGF(k) = HRG(k) = none$. However, at $k_{EGF} = k_{end}/10$, EGF is activated: $\forall k > k_{EGF},\ EGF(k) = full$. Therefore, f_{EGF}^{node} and f_{HRG}^{node} become:

$$EGF(k+1) = \begin{cases} full & \text{if } k \geq k_{EGF} \\ none & \text{if } k < k_{EGF} \end{cases}$$
$$HRG(k+1) = none$$

The network being assumed to be at the resting state, $\boldsymbol{x}_0 = (\dots, none, \dots)$. The p_{ij} are set to $fast$ and the q_{ij} to $strong$. The corresponding results are shown in *Figure 3.2* page 75. As expected, before EGF activation, the network is at rest: the signaling cascade is not active. However, once EGF activated, the signaling cascade activates. This ultimately activates ERK, hence the subsequent inactivation of $PI3K$ despite sustained $EGFR$ activity. Since AKT is activated by $PI3K$, it also deactivates.

3.3.2 Simulation 2

In addition to the inputs described in simulation 1, a perturbation is introduced. It consists in disabling the inhibitory effect of ERK on $PI3K$, that

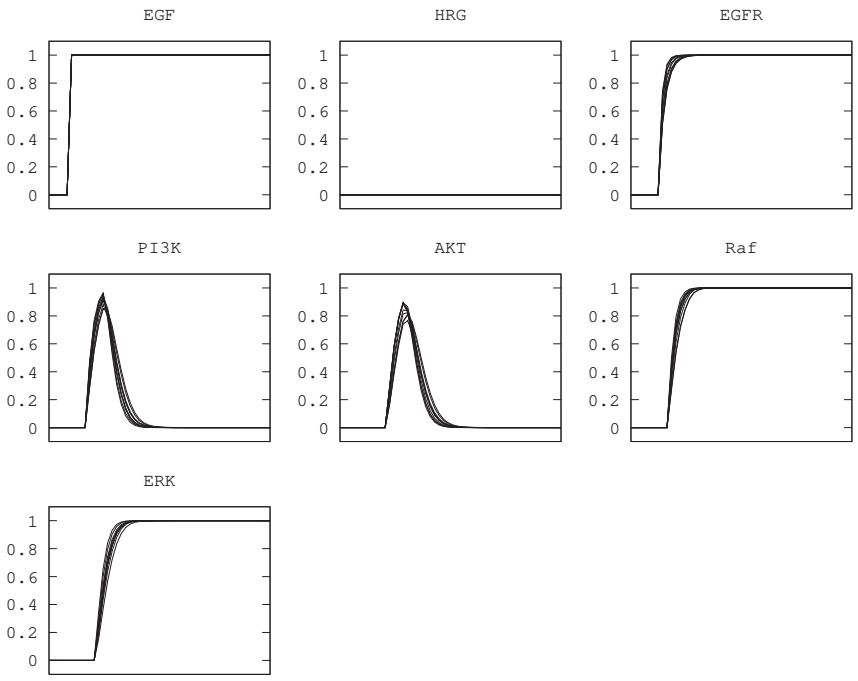

Figure 3.2 – Activation of the signaling cascade by EGF and subsequent inhibition of $PI3K$ by ERK.

is in disabling the edge $(ERK, PI3K)$. It points out an advantage of computing edge states: disturbing a node disturbs all its effects while selectively disturbing edges prevents this. To implement this perturbation, the parameter values are as in simulation 1, except $q_{ERK,PI3K}$ which is set to $weaker$. With $q_{ERK,PI3K} = weaker$, the signal conveyed by the edge $(ERK, PI3K)$ is weakened throughout this simulation. The corresponding results are shown in *Figure 3.3* page 76. As expected, weakening the edge $(ERK, PI3K)$ results in a weakened inhibition of $PI3K$ by ERK: ERK does not totally inhibit $PI3K$.

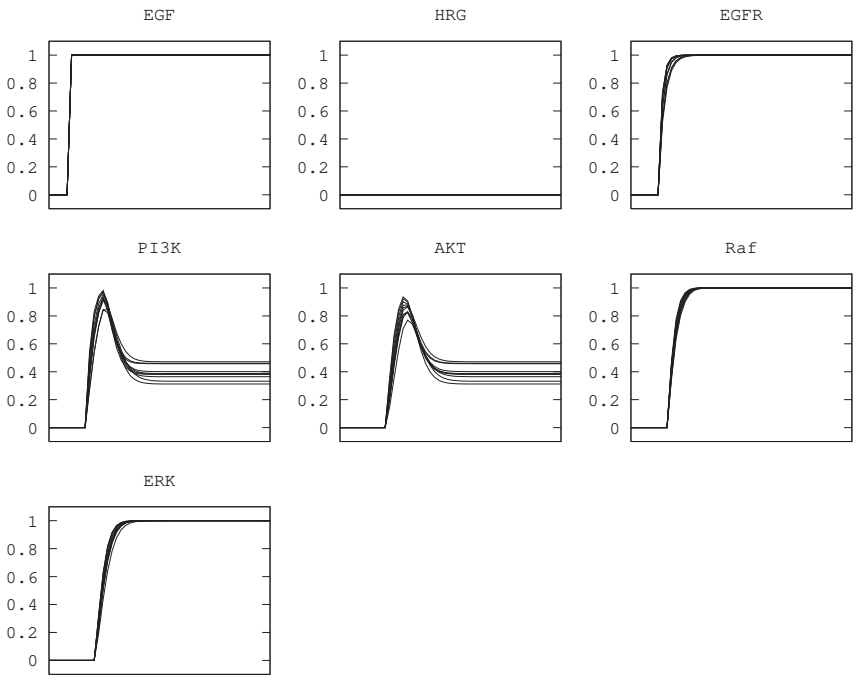

Figure 3.3 – Weakening the inhibitory effect of ERK on $PI3K$.

3.3.3 Simulation 3

A perturbation is again applied to the edge $(ERK, PI3K)$. However, in this simulation the perturbation concerns its reactivity, namely $p_{ERK,PI3K}$, which is set to *slower*. The other parameter values are as in simulation 1. With $p_{ERK,PI3K} = slower$, the signal conveyed by the edge $(ERK, PI3K)$ is slowed throughout this simulation. The corresponding results are shown in *Figure 3.4* page 77. As expected, slowing the edge $(ERK, PI3K)$ results in a slowed inhibition of $PI3K$ by ERK: although ERK totally inhibits $PI3K$, it does it slower than in simulation 1 where $p_{ERK,PI3K} = fast$.

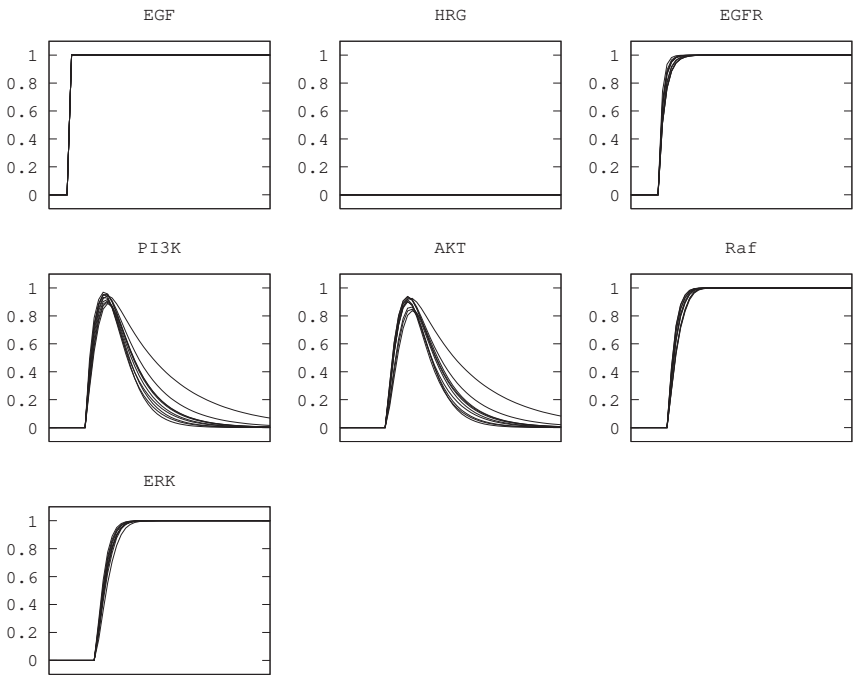

Figure 3.4 – Slowing the inhibitory effect of ERK on $PI3K$.

3.3.4 Simulation 4

In this simulation, no perturbations are applied and the parameter values are as in simulation 1. However, rather than totally activating EGF, it is set to few. Therefore, f^{node}_{EGF} and f^{node}_{HRG} become:

$$EGF(k+1) = \begin{cases} few & \text{if } k \geq k_{EGF} \\ none & \text{if } k < k_{EGF} \end{cases}$$
$$HRG(k+1) = none$$

The corresponding results are shown in *Figure 3.5* page 78. As expected, the activation of EGF is not total and the same applies to the entire signaling cascade. For example, $PI3K$ does not totally activate since $EGFR$ does

not. Furthermore, $PI3K$ is not totally inhibited by ERK since ERK itself does not totally activate.

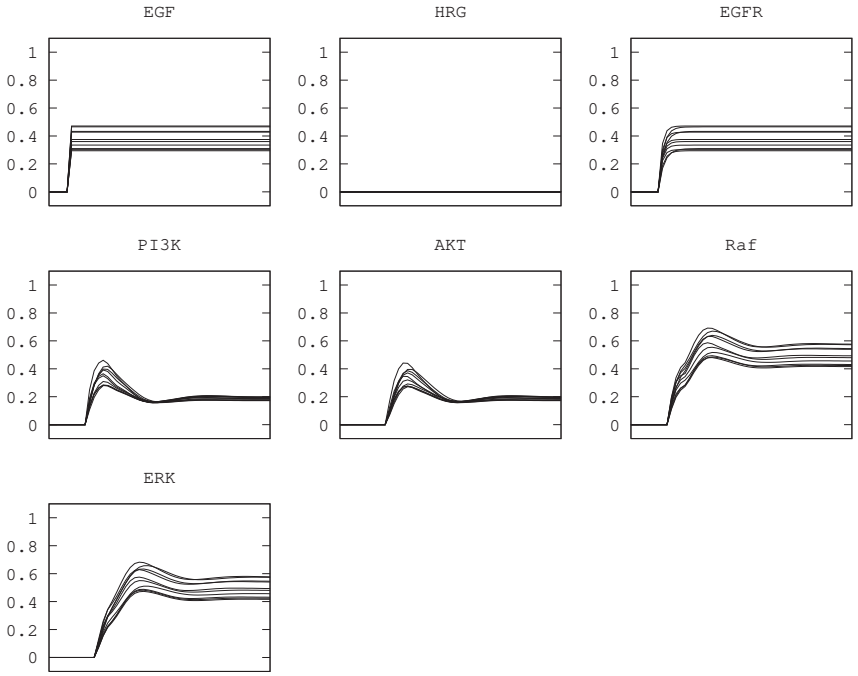

Figure 3.5 – Consequences on the signaling cascade of a partial activation of EGF.

3.3.5 Simulation 5

In this simulation, both EGF and HRG are set to few. Therefore, f_{EGF}^{node} and f_{HRG}^{node} become:

$$EGF(k+1) = \begin{cases} few & \text{if } k \geq k_{EGF} \\ none & \text{if } k < k_{EGF} \end{cases}$$

$$HRG(k+1) = \begin{cases} few & \text{if } k \geq k_{HRG} \\ none & \text{if } k < k_{HRG} \end{cases}$$

with $k_{HRG} = k_{EGF}$, the other parameter values being as in simulation 1. The corresponding results are shown in *Figure 3.6* page 79. It points out that the effect of EGF and HRG on $EGFR$ is cumulative due to an OR gate. Indeed, although both EGF and HRG are set to few, cumulating their effect on $EGFR$ makes the signaling cascade more active than in simulation 4 where only EGF is set to few.

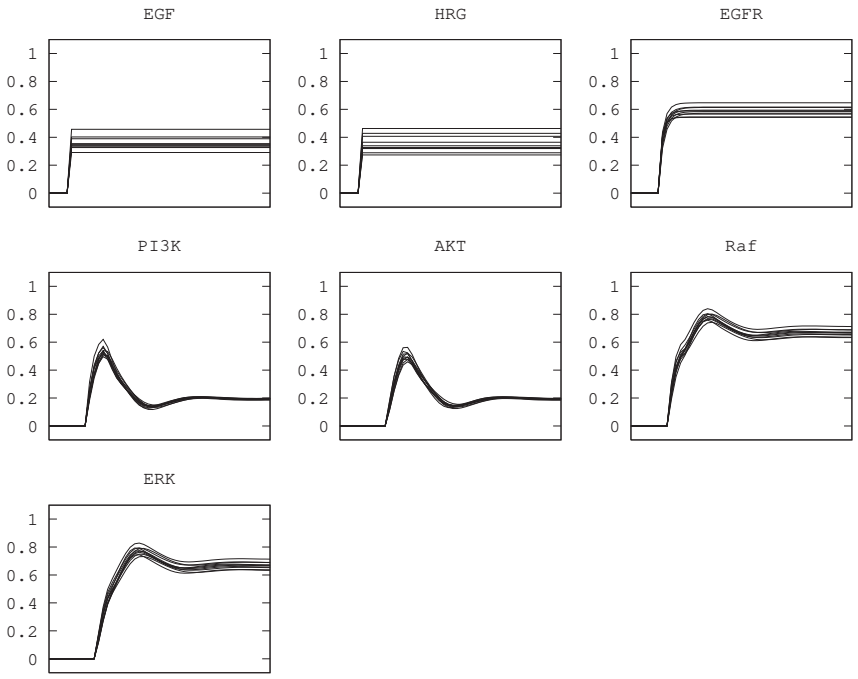

Figure 3.6 – Cumulative effect of partly activated EGF and HRG on the signaling cascade.

3.4 Discussion

Owing to the use of fuzzy operators, the simulations performed with the example show that the proposed logic-based modeling is able to produce continuous results while remaining qualitative. This allows qualitative variables to be more finely valued than with discrete approaches, such as Boolean

networks, by taking into account all the possible levels of what is modeled. Moreover, thanks to the additional features *edge reactivity* and *edge weakening* attributed to each edge, it is possible to tune in speed and strength the interactions taking place in the modeled biological network. This is expected to take into account that some interactions can be weaker or slower relative to other ones and therefore to be more realistic in their qualitative modeling.

A little of stochasticity on the two additional features *edge reactivity* and *edge weakening* is also realized through the random selection of their value in specified intervals followed by replication and superposition of the produced results. This stochasticity, although very rudimentary, constitutes a line of improvement which should yield more realism since events taking place in biological systems are themselves subjected to stochasticity [130, 131]. Another improvement could be to apply information theory [132] on the signal conveyed by the edges, as previously introduced for cell signaling [133, 134]. This improvement should enable to better model how the edges of a network convey the information, particularly how they preserve its fidelity against noise from its sender, such as a receptor, to its receiver, such as a transcription factor. Altogether, starting from Boolean networks and still founded on their basic principles, this work is expected to bring a fine qualitative quantification of the behavior of biological networks.

A qualitative quantification remains qualitative and should not be confused with a true quantification which involves experimental measurements, values and units [135]. The qualitative quantification proposed in this work has the goal of bringing enhancements in the ability of qualitative models to simulate the behavior of biological networks. One of the main goals, and advantages, of qualitative modeling remains to propose an alternative to, but not a replacement of, quantitative approaches when the frequently encountered scarcity in quantitative information makes the work unreasonably or unnecessarily difficult.

It is also possible to use qualitative and quantitative approaches in combination. For example, qualitative modeling can be used to explore global properties and then quantitative modeling can be used to focus on particular aspects. Knowing the difficulty of quantitative modeling in systems biology,

this two-steps approach could make modeling more efficient by highlighting where to deploy quantitative approaches. Qualitative and quantitative approaches can also be merged into hybrid models [136–138] which attempt to exploit the advantages of these two approaches in one. Hybrid models, or semi-quantitative models, can be good compromises between the convenience of qualitative modeling and the accuracy of quantitative modeling.

Chapter 4

Conclusion

This thesis devoted to the qualitative modeling of biological networks for therapeutic innovation brings two works. The first one is an algorithm for *in silico* target identification in Boolean models of pathologically disturbed biological networks while the second one aims at improving the Boolean network formalism in modeling the dynamics of biological networks.

The algorithm for *in silico* target identification shows that it finds, in Boolean models of pathologically disturbed biological networks, combinations of targets able to push pathological behaviors toward physiological ones. It is intended to be of use in the early steps of target identification by providing an efficient way to identify candidate targets prior to costly wet lab experiments. However, this algorithm produces *in silico* results and has to be considered as such: mathematical models approximate reality without reproducing it, qualitative modeling is not quantitative, and theory must meet practice. Consequently, it should be used in combination with wet lab experiments in a synergistic manner aimed at improving the efficiency of the overall target identification process by performing prior screenings of candidate targets. This is why the criterion for selecting therapeutic bullets is softened: with a too strict criterion, the risk of highlighting too few candidate targets and to miss some interesting ones is too hight. It should be noted that this algorithm fits into the encompassing field investigating how to control biological systems, a field with tremendous applications in biomedicine. Several

endeavors based on qualitative modeling approaches have been made in this way [139–144], demonstrating its utility in investigating how to take control over pathologically disturbed biological systems.

The second work proposes a qualitative modeling approach derived from Boolean networks. It adds the possibility to tune the edges of the network according to knowledge about the modeled interactions. Furthermore, by using fuzzy operators, it allows continuous simulations to be performed. These extensions should enable to incorporate more knowledge, notably about biological processes, and to obtain more accurate results. In exchange, it requires the parameters controlling how the signal flows in the edges to be valued. These parameters are intended to be qualitative, that is parameters whose the valuation is knowledge-based, by opposition to quantitative parameters whose valuation is data-based. In other words, qualitative parameters translate qualitative information, an information which should be easier to obtain than the quantitative one. Indeed, quantitative models require their parameters to be valued by data obtained through experimental measurements. However, due to experimental limitations, such measurements can be challenging. Qualitative information is easier to obtain but at the cost of being qualitative, as its name indicates. This is the well-known trade-off between what is wished and what is obtainable.

Two improvements were done on the first work after its publication, namely handling multivalued logic and softening the selection of therapeutic bullets. These two improvements are relatively minor since they do not change the computational principles of the algorithm, namely computing dynamics of discrete dynamical systems. The second work proposes a continuous qualitative modeling, thus requiring to move from discrete to continuous dynamical systems. The continuous can be seen as the ultimate generalization of the discrete and is surely more accurate but brings some challenging differences. For example, with discrete dynamical systems of reasonable size, it is possible to compute the whole state space and then to find all the attractors. For discrete dynamical systems of bigger size, owing to the finite cardinality of their state space, it is at least possible to quantify which portion of it is computed and then to estimate the likeliness of finding

all the attractors. With continuous dynamical systems, the state space has an infinite cardinality. Therefore, estimating the likeliness of finding all the attractors is far less straightforward, not to mention the certainty of finding all the attractors.

Continuous dynamical systems are mostly modeled by differential equations for which advanced solvers are available, such as LSODE (the Livermore Solver for Ordinary Differential Equations) [145]. The second work introduces continuous dynamical systems made of logical equations, for which advanced solvers do not seem to exist. However, mathematically speaking, it is likely that these continuous logical equations are differential equations thought and built in a different way. Consequently, it would be possible to mathematically express them as differential equations and then to use available computational tools aimed at analyzing continuous dynamical systems. This is a way to pursue the second work and to incorporate it into the first one in order to build a version of it based on continuous dynamical systems.

Appendix A

The algorithm in one block of pseudocode.

1 **prompt** *card* D
2 $card\ D = min(card\ D, 2^n)$
3 generate $D \subseteq S$
4 $H = \{\}$
5 $A_{physio} = \{\}$
6 **for** $x_0 \in D$ **do**
7 $\quad k = 1$
8 $\quad \boldsymbol{x}(k) = x_0$
9 $\quad$ **while true do**
10 $\quad\quad$ **if** $\exists w \in H : \boldsymbol{x}(k) \in w$ **then**
11 $\quad\quad\quad$ **break**
12 $\quad\quad$ **end if**
13 $\quad\quad \boldsymbol{x}(k+1) = \boldsymbol{f}_{physio}(\boldsymbol{x}(k))$
14 $\quad\quad$ **if** $\exists k' \in [\![1, k]\!] : \boldsymbol{x}(k+1) = \boldsymbol{x}(k')$ **then**
15 $\quad\quad\quad A_{physio} = A_{physio} \cup \{(\boldsymbol{x}(k'), \ldots, \boldsymbol{x}(k))\}$
16 $\quad\quad\quad$ **break**
17 $\quad\quad$ **end if**
18 $\quad\quad k = k + 1$
19 $\quad$ **end while**
20 $\quad H = H \cup \{(\boldsymbol{x}(1), \ldots, \boldsymbol{x}(k))\}$
21 **end for**
22 **return** A_{physio}

23 **prompt** $r_{min}, r_{max}, max_{targ}, max_{moda}$
24 $r_{max} = min(r_{max}, n)$
25 $golden\_set = \{\}$
26 $silver\_set = \{\}$
27 **for** $r \in [\![r_{min}, r_{max}]\!]$ **do**
28 $\quad max^r_{targ} = min(max_{targ}, n!/(r! \cdot (n-r)!))$
29 $\quad max^r_{moda} = min(max_{moda}, 2^r)$
30 $\quad C_{targ} = \{\}$
31 $\quad C_{moda} = \{\}$
32 $\quad$ **while** $card\ C_{targ} < max^r_{targ}$ **do**
33 $\quad\quad$ generate $c_{targ} \notin C_{targ}$
34 $\quad\quad C_{targ} = C_{targ} \cup \{c_{targ}\}$
35 $\quad$ **end while**
36 $\quad$ **while** $card\ C_{moda} < max^r_{moda}$ **do**
37 $\quad\quad$ generate $c_{moda} \notin C_{moda}$
38 $\quad\quad C_{moda} = C_{moda} \cup \{c_{moda}\}$
39 $\quad$ **end while**
40 $\quad$ **for** $c_{targ} \in C_{targ}$ **do**
41 $\quad\quad$ **for** $c_{moda} \in C_{moda}$ **do**
42 $\quad\quad\quad H = \{\}$
43 $\quad\quad\quad A_{patho} = \{\}$
44 $\quad\quad\quad$ **for** $x_0 \in D$ **do**
45 $\quad\quad\quad\quad k = 1$
46 $\quad\quad\quad\quad \boldsymbol{x}(k) = x_0$
47 $\quad\quad\quad\quad$ **while true do**
48 $\quad\quad\quad\quad\quad$ **if** $\exists w \in H : \boldsymbol{x}(k) \in w$ **then**
49 $\quad\quad\quad\quad\quad\quad$ **break**
50 $\quad\quad\quad\quad\quad$ **end if**
51 $\quad\quad\quad\quad\quad \boldsymbol{x}(k+1) = \boldsymbol{f}_{patho}(\boldsymbol{x}(k))$
52 $\quad\quad\quad\quad\quad$ **for** $targ_i \in c_{targ}$ **do**
53 $\quad\quad\quad\quad\quad\quad$ **for** $v_j \in V$ **do**
54 $\quad\quad\quad\quad\quad\quad\quad$ **if** $v_j = targ_i$ **then**
55 $\quad\quad\quad\quad\quad\quad\quad\quad x_j(k+1) = moda_i$

```
                        end if
                    end for
                end for
                if ∃k′ ∈ ⟦1, k⟧: x(k + 1) = x(k′) then
                    A_patho = A_patho ∪ {(x(k′), . . . , x(k))}
                    break
                end if
                k = k + 1
            end while
            H = H ∪ {(x(1), . . . , x(k))}
        end for
        if A_patho ⊆ A_physio then
            if A_patho = A_physio then
                golden_set = golden_set ∪ {(c_targ, c_moda)}
            else
                silver_set = silver_set ∪ {(c_targ, c_moda)}
            end if
        end if
    end for
end for
end for
return golden_set, silver_set
```

Appendix B

Boolean functions of the case study where, for the sake of readability, x_i stands for $x_i(k)$ and x_{i+} stands for $x_i(k+1)$:

$$
\begin{aligned}
ICL_+ &= ICL \wedge \neg DSB \\
FANCM_+ &= ICL \wedge \neg CHKREC \\
FAcore_+ &= FANCM \wedge (ATR \vee ATM) \wedge \neg CHKREC \\
FANCD2I_+ &= FAcore \wedge ((ATM \vee ATR) \vee (H2AX \wedge DSB)) \wedge \neg USP1 \\
MUS81_+ &= ICL \\
FANCJBRCA1_+ &= (ICL \vee ssDNARPA) \wedge (ATM \vee ATR) \\
XPF_+ &= (MUS81 \wedge p53 \wedge \neg(FAcore \wedge FANCD2I \wedge FAN1)) \\
&\quad \vee (MUS81 \wedge \neg FANCM) \\
FAN1_+ &= MUS81 \wedge FANCD2I \\
ADD_+ &= (ADD \vee (MUS81 \wedge (FAN1 \vee XPF))) \wedge \neg PCNATLS \\
DSB_+ &= (DSB \vee FAN1 \vee XPF) \wedge \neg(NHEJ \vee HRR) \\
PCNATLS_+ &= (ADD \vee (ADD \wedge FAcore)) \wedge \neg(USP1 \vee FAN1) \\
MRN_+ &= DSB \wedge ATM \wedge \neg((KU \wedge FANCD2I) \vee RAD51 \vee CHKREC) \\
BRCA1_+ &= DSB \wedge (ATM \vee CHK2 \vee ATR) \wedge \neg CHKREC \\
ssDNARPA_+ &= DSB \wedge ((FANCD2I \wedge FANCJBRCA1) \vee MRN) \\
&\quad \wedge \neg(RAD51 \vee KU)
\end{aligned}
$$

$$
\begin{aligned}
FANCD1N_{+} &= (FANCD2I \wedge ssDNARPA \wedge \neg CHKREC) \\
&\quad \vee (ssDNARPA \wedge BRCA1) \\
RAD51_{+} &= ssDNARPA \wedge FANCD1N \wedge \neg CHKREC \\
HRR_{+} &= DSB \wedge RAD51 \wedge FANCD1N \wedge BRCA1 \wedge \neg CHKREC \\
USP1_{+} &= ((FANCD1N \wedge FANCD2I) \vee PCNATLS) \wedge \neg FANCM \\
KU_{+} &= DSB \wedge \neg(MRN \vee FANCD2I \vee CHKREC) \\
DNAPK_{+} &= (DSB \wedge KU) \wedge \neg CHKREC \\
NHEJ_{+} &= ((DSB \wedge DNAPK \wedge KU) \wedge \neg(ATM \wedge ATR)) \\
&\quad \vee (\neg((FANCJBRCA1 \wedge ssDNARPA) \vee CHKREC) \\
&\quad \wedge DSB \wedge DNAPK \wedge XPF) \\
ATR_{+} &= (ssDNARPA \vee FANCM \vee ATM) \wedge \neg CHKREC \\
ATM_{+} &= (ATR \vee DSB) \wedge \neg CHKREC \\
p53_{+} &= (((ATM \wedge CHK2) \vee (ATR \wedge CHK1)) \vee DNAPK) \wedge \neg CHKREC \\
CHK1_{+} &= (ATM \vee ATR \vee DNAPK) \wedge \neg CHKREC \\
CHK2_{+} &= (ATM \vee ATR \vee DNAPK) \wedge \neg CHKREC \\
H2AX_{+} &= DSB \wedge (ATM \vee ATR \vee DNAPK) \wedge \neg CHKREC \\
CHKREC_{+} &= ((PCNATLS \vee NHEJ \vee HRR) \wedge \neg DSB) \\
&\quad \vee ((\neg ADD) \wedge (\neg ICL) \wedge (\neg DSB) \wedge \neg CHKREC)
\end{aligned}
$$

Appendix C

Therapeutic bullets found for the case study.

$-ATM$		golden
$-ATM$	$-CHK2$	golden
$-HRR$	$-ATM$	golden
$-ssDNARPA$	$-ATM$	golden
$-BRCA1$	$-ATM$	golden
$-MRN$	$-ATM$	golden
$-FAN1$	$-ATM$	golden
$-ICL$	$-DSB$	golden
$-FAcore$	$-ATM$	golden
$-USP1$	$-ATM$	golden
$-ATM$	$-H2AX$	golden
$-ADD$	$-ATM$	golden
$-RAD51$	$-ATM$	golden
$-XPF$	$-ATM$	golden
$-FANCM$	$-ATM$	golden
$-FANCD1N$	$-ATM$	golden
$-ATM$	$-CHK1$	golden
$-ICL$	$-ATM$	golden
$-ATM$	$-p53$	golden
$-FANCJBRCA1$	$-ATM$	golden

$-FANCD2I$	$-ATM$		golden
$-ICL$	$-FANCD1N$	$-ATM$	golden
$-ICL$	$-FAcore$	$-DSB$	golden
$-BRCA1$	$-USP1$	$-ATM$	golden
$-BRCA1$	$-ssDNARPA$	$-ATM$	golden
$-BRCA1$	$-ATM$	$-CHK1$	golden
$-ADD$	$-ATM$	$-H2AX$	golden
$-FAN1$	$-MRN$	$-ATM$	golden
$-ATM$	$-CHK2$	$-H2AX$	golden
$-ICL$	$-DSB$	$-MRN$	golden
$-XPF$	$-MRN$	$-ATM$	golden
$-FAcore$	$-FANCD2I$	$-ATM$	golden
$-FANCM$	$-ATM$	$-CHK2$	golden
$-RAD51$	$-ATM$	$-p53$	golden
$-ICL$	$-ssDNARPA$	$-ATM$	golden
$-FANCM$	$-ATR$	$-ATM$	golden
$-RAD51$	$-ATM$	$-H2AX$	golden
$-ADD$	$-FANCD1N$	$-ATM$	golden
$-ICL$	$-USP1$	$-ATM$	golden
$-FANCM$	$-MRN$	$-ATR$	golden
$-MRN$	$-USP1$	$-ATM$	golden
$-FAN1$	$-HRR$	$-ATM$	golden
$-BRCA1$	$-ATM$	$-H2AX$	golden
$-FANCJBRCA1$	$-ADD$	$-ATM$	golden
$-MRN$	$-ssDNARPA$	$-ATM$	golden
$-FAcore$	$-ssDNARPA$	$-ATM$	golden
$-FAcore$	$-FANCD1N$	$-ATM$	golden
$-FANCD2I$	$-BRCA1$	$-ATM$	golden
$-ADD$	$-MRN$	$-ATM$	golden
$-ATM$	$-p53$	$-CHK2$	golden
$-RAD51$	$-ATM$	$-CHK2$	golden
$-FANCM$	$-ATM$	$-H2AX$	golden
$-ADD$	$-PCNATLS$	$-ATM$	golden
$-FANCJBRCA1$	$-ATM$	$-p53$	golden
$-FANCM$	$-MRN$	$-ATM$	golden

$-FANCJBRCA1$	$-ATM$	$-CHK2$	golden
$-FANCD2I$	$-USP1$	$-ATM$	golden
$-ADD$	$-ATM$	$-CHK2$	golden
$-FANCD2I$	$-FANCD1N$	$-ATM$	golden
$-MRN$	$-HRR$	$-ATM$	golden
$-ICL$	$-DSB$	$-USP1$	golden
$-FAN1$	$-FANCD1N$	$-ATM$	golden
$-FAN1$	$-ATM$	$-H2AX$	golden
$-FANCJBRCA1$	$-FAN1$	$-ATM$	golden
$-ssDNARPA$	$-ATM$	$-H2AX$	golden
$-ATM$	$-CHK1$	$-CHK2$	golden
$-ADD$	$-HRR$	$-ATM$	golden
$-ATM$	$-p53$	$-CHK1$	golden
$-FAcore$	$-ATM$	$-H2AX$	golden
$-FANCD2I$	$-ATM$	$-CHK2$	golden
$-FAN1$	$-RAD51$	$-ATM$	golden
$-FANCD2I$	$-RAD51$	$-ATM$	golden
$-FANCJBRCA1$	$-XPF$	$-ATM$	golden
$-ICL$	$-FANCJBRCA1$	$-DSB$	golden
$-ssDNARPA$	$-HRR$	$-ATM$	golden
$-MRN$	$-BRCA1$	$-ATM$	golden
$-FANCM$	$-FAN1$	$-ATM$	golden
$-ssDNARPA$	$-ATM$	$-p53$	golden
$-FAN1$	$-ATM$	$-CHK2$	golden
$-FANCD2I$	$-ssDNARPA$	$-ATM$	golden
$-FANCD2I$	$-FAN1$	$-ATM$	golden
$-XPF$	$-HRR$	$-ATM$	golden
$-FAN1$	$-BRCA1$	$-ATM$	golden
$-ADD$	$-ATM$	$-CHK1$	golden
$-FAcore$	$-HRR$	$-ATM$	golden
$-XPF$	$-ATM$	$-CHK1$	golden
$-ADD$	$-BRCA1$	$-ATM$	golden
$-ICL$	$-FAN1$	$-DSB$	golden
$-ADD$	$-ATM$	$-p53$	golden
$-ICL$	$-MUS81$	$-ATM$	golden

$-FAcore$	$-RAD51$	$-ATM$	golden
$-ATM$	$-CHK1$	$-H2AX$	golden
$-ICL$	$-MRN$	$-ATM$	golden
$-ssDNARPA$	$-ATM$	$-CHK2$	golden
$-XPF$	$-RAD51$	$-ATM$	golden
$-FANCM$	$-ATM$	$-CHK1$	golden
$-ICL$	$-DSB$	$-KU$	golden
$-ICL$	$-MRN$	$-ATR$	golden
$-ssDNARPA$	$-RAD51$	$-ATM$	golden
$-FANCJBRCA1$	$-ssDNARPA$	$-ATM$	golden
$-XPF$	$-ATM$	$-p53$	golden
$-FAcore$	$-MRN$	$-ATM$	golden
$-HRR$	$-ATM$	$-H2AX$	golden
$-HRR$	$-ATM$	$-p53$	golden
$-FANCJBRCA1$	$-FANCD1N$	$-ATM$	golden
$-FANCM$	$-ADD$	$-ATM$	golden
$-FAcore$	$-ATM$	$-CHK2$	golden
$-ICL$	$-ATM$	$-CHK1$	golden
$-MRN$	$-FANCD1N$	$-ATM$	golden
$-ADD$	$-ssDNARPA$	$-ATM$	golden
$-MRN$	$-RAD51$	$-ATM$	golden
$-FANCD1N$	$-ATM$	$-p53$	golden
$-FANCD1N$	$-RAD51$	$-ATM$	golden
$-BRCA1$	$-ATM$	$-CHK2$	golden
$-ADD$	$-RAD51$	$-ATM$	golden
$-ICL$	$-DSB$	$-FANCD1N$	golden
$-ICL$	$-RAD51$	$-ATM$	golden
$-ICL$	$-ATM$	$-CHK2$	golden
$-FANCD1N$	$-ATM$	$-H2AX$	golden
$-MRN$	$-ATM$	$-H2AX$	golden
$-FAcore$	$-FAN1$	$-ATM$	golden
$-ICL$	$-XPF$	$-ATM$	golden
$-FANCD2I$	$-ADD$	$-ATM$	golden
$-FANCD2I$	$-ATM$	$-H2AX$	golden
$-ICL$	$-ATR$	$-ATM$	golden

$-FANCM$	$-HRR$	$-ATM$	golden
$-USP1$	$-ATM$	$-H2AX$	golden
$-ICL$	$-DSB$	$-RAD51$	golden
$-ICL$	$-ATM$	$-H2AX$	golden
$-FANCD1N$	$-USP1$	$-ATM$	golden
$-FANCM$	$-FANCD2I$	$-ATM$	golden
$-FANCD2I$	$-MRN$	$-ATM$	golden
$-FAcore$	$-ADD$	$-ATM$	golden
$-ICL$	$-FAcore$	$-ATM$	golden
$-FANCM$	$-ssDNARPA$	$-ATM$	golden
$-XPF$	$-ATM$	$-H2AX$	golden
$-FAcore$	$-USP1$	$-ATM$	golden
$-HRR$	$-ATM$	$-CHK1$	golden
$-BRCA1$	$-RAD51$	$-ATM$	golden
$-FAN1$	$-ADD$	$-ATM$	golden
$-FANCJBRCA1$	$-MRN$	$-ATM$	golden
$-FANCM$	$-USP1$	$-ATM$	golden
$-FANCJBRCA1$	$-ATM$	$-H2AX$	golden
$-FANCM$	$-FAcore$	$-ATM$	golden
$-HRR$	$-USP1$	$-ATM$	golden
$-ICL$	$-FANCM$	$-ATM$	golden
$-ICL$	$-DSB$	$-ssDNARPA$	golden
$-FAN1$	$-USP1$	$-ATM$	golden
$-FANCM$	$-FANCJBRCA1$	$-ATM$	golden
$-ssDNARPA$	$-ATM$	$-CHK1$	golden
$-FAcore$	$-FANCJBRCA1$	$-ATM$	golden
$-FANCD2I$	$-HRR$	$-ATM$	golden
$-FANCD2I$	$-FANCJBRCA1$	$-ATM$	golden
$-XPF$	$-ssDNARPA$	$-ATM$	golden
$-USP1$	$-ATM$	$-CHK1$	golden
$-ICL$	$-DSB$	$-ATM$	golden
$-ICL$	$-ADD$	$-DSB$	golden
$-USP1$	$-ATM$	$-CHK2$	golden
$-XPF$	$-BRCA1$	$-ATM$	golden
$-RAD51$	$-ATM$	$-CHK1$	golden

$-FANCD1N$	$-ATM$	$-CHK2$	golden
$-RAD51$	$-HRR$	$-ATM$	golden
$-ICL$	$-ATM$	$-p53$	golden
$-ICL$	$-DSB$	$-DNAPK$	golden
$-FANCM$	$-FANCD1N$	$-ATM$	golden
$-BRCA1$	$-FANCD1N$	$-ATM$	golden
$-ICL$	$-HRR$	$-ATM$	golden
$-FANCJBRCA1$	$-HRR$	$-ATM$	golden
$-USP1$	$-ATM$	$-p53$	golden
$-XPF$	$-ATM$	$-CHK2$	golden
$-ICL$	$-DSB$	$-CHK2$	golden
$-ICL$	$-XPF$	$-DSB$	golden
$-ssDNARPA$	$-FANCD1N$	$-ATM$	golden
$-FANCJBRCA1$	$-RAD51$	$-ATM$	golden
$-ICL$	$-DSB$	$-ATR$	golden
$-HRR$	$-ATM$	$-CHK2$	golden
$-ADD$	$-USP1$	$-ATM$	golden
$-FANCM$	$-RAD51$	$-ATM$	golden
$-FANCJBRCA1$	$-ATM$	$-CHK1$	golden
$-FANCM$	$-ATM$	$-p53$	golden
$-XPF$	$-FANCD1N$	$-ATM$	golden
$-FAcore$	$-BRCA1$	$-ATM$	golden
$-ICL$	$-DSB$	$-NHEJ$	golden
$-BRCA1$	$-ATM$	$-p53$	golden
$-BRCA1$	$-HRR$	$-ATM$	golden
$-FANCJBRCA1$	$-USP1$	$-ATM$	golden
$-ssDNARPA$	$-USP1$	$-ATM$	golden
$-ICL$	$-DSB$	$-H2AX$	golden
$-FANCM$	$-BRCA1$	$-ATM$	golden
$-MRN$	$-ATM$	$-CHK1$	golden
$-ICL$	$-FANCJBRCA1$	$-ATM$	golden
$-FANCD1N$	$-ATM$	$-CHK1$	golden
$-ICL$	$-DSB$	$-BRCA1$	golden
$-MRN$	$-ATM$	$-CHK2$	golden
$-FANCJBRCA1$	$-BRCA1$	$-ATM$	golden

$-FAN1$	$-ssDNARPA$	$-ATM$	golden
$-MRN$	$-ATM$	$-p53$	golden
$-FANCD1N$	$-HRR$	$-ATM$	golden
$-ICL$	$-MUS81$	$-DSB$	golden
$-ICL$	$-DSB$	$-p53$	golden
$-XPF$	$-USP1$	$-ATM$	golden
$-XPF$	$-ADD$	$-ATM$	golden
$-ATM$	$-p53$	$-H2AX$	golden
$-ICL$	$-FANCM$	$-DSB$	golden
$-ICL$	$-DSB$	$-HRR$	golden
$-ICL$	$-BRCA1$	$-ATM$	golden
$-RAD51$	$-USP1$	$-ATM$	golden
$-ICL$	$-FAN1$	$-ATM$	golden
$-ICL$	$-ADD$	$-ATM$	golden
$-ICL$	$-DSB$	$-CHK1$	golden
$-ICL$	$-FANCD2I$	$-DSB$	golden
$-ICL$	$-FANCD2I$	$-ATM$	golden

Appendix D

Therapeutic bullets found for the case study using the new criterion.

bullet		gain			$B_{physio1}$	B_{patho1}
$-FANCM$		29.4%	$\rightarrow$	44.6%	44.6%	55.4%
$-FANCD2I$		29.4%	$\rightarrow$	30.4%	30.4%	69.6%
$-XPF$		29.4%	$\rightarrow$	46.2%	46.2%	53.8%
$-FAN1$		29.4%	$\rightarrow$	32.9%	32.9%	67.1%
$-ATM$		29.4%	$\rightarrow$	100%	100%	0%
$-ICL$	$-FANCD2I$	29.4%	$\rightarrow$	30.9%	30.9%	69.1%
$-ICL$	$-MUS81$	29.4%	$\rightarrow$	53%	53%	47%
$-ICL$	$-XPF$	29.4%	$\rightarrow$	58.6%	58.6%	41.4%
$-ICL$	$-FAN1$	29.4%	$\rightarrow$	33.9%	33.9%	66.1%
$-ICL$	$-DSB$	29.4%	$\rightarrow$	100%	100%	0%
$-ICL$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-FANCM$	$-FAcore$	29.4%	$\rightarrow$	45.8%	45.8%	54.2%
$-FANCM$	$-FANCD2I$	29.4%	$\rightarrow$	46.3%	46.3%	53.7%
$-FANCM$	$-FAN1$	29.4%	$\rightarrow$	47.3%	47.3%	52.7%
$-FANCM$	$-ADD$	29.4%	$\rightarrow$	47.3%	47.3%	52.7%
$-FANCM$	$-FANCD1N$	29.4%	$\rightarrow$	44.6%	44.6%	55.4%
$-FANCM$	$-RAD51$	29.4%	$\rightarrow$	44.6%	44.6%	55.4%
$-FANCM$	$-HRR$	29.4%	$\rightarrow$	44.1%	44.1%	55.9%
$-FANCM$	$-USP1$	29.4%	$\rightarrow$	44.3%	44.3%	55.7%
$-FANCM$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%

bullet		gain			$B_{physio1}$	B_{patho1}
$-FAcore$	$-FANCD2I$	29.4%	$\rightarrow$	30.4%	30.4%	69.6%
$-FAcore$	$-FAN1$	29.4%	$\rightarrow$	33%	33%	67%
$-FAcore$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-FANCD2I$	$-FAN1$	29.4%	$\rightarrow$	33.2%	33.2%	66.8%
$-FANCD2I$	$-ADD$	29.4%	$\rightarrow$	30.5%	30.5%	69.5%
$-FANCD2I$	$-FANCD1N$	29.4%	$\rightarrow$	30.4%	30.4%	69.6%
$-FANCD2I$	$-RAD51$	29.4%	$\rightarrow$	30.4%	30.4%	69.6%
$-FANCD2I$	$-USP1$	29.4%	$\rightarrow$	30.4%	30.4%	69.6%
$-FANCD2I$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-FANCJBRCA1$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-XPF$	$-ADD$	29.4%	$\rightarrow$	46.2%	46.2%	53.8%
$-XPF$	$-FANCD1N$	29.4%	$\rightarrow$	46.2%	46.2%	53.8%
$-XPF$	$-RAD51$	29.4%	$\rightarrow$	46.2%	46.2%	53.8%
$-XPF$	$-HRR$	29.4%	$\rightarrow$	45.3%	45.3%	54.7%
$-XPF$	$-USP1$	29.4%	$\rightarrow$	46.2%	46.2%	53.8%
$-XPF$	$-KU$	29.4%	$\rightarrow$	46.1%	46.1%	53.9%
$-XPF$	$-DNAPK$	29.4%	$\rightarrow$	46.1%	46.1%	53.9%
$-XPF$	$-NHEJ$	29.4%	$\rightarrow$	41.6%	41.6%	58.4%
$-XPF$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-FAN1$	$-ADD$	29.4%	$\rightarrow$	32.9%	32.9%	67.1%
$-FAN1$	$-FANCD1N$	29.4%	$\rightarrow$	32.9%	32.9%	67.1%
$-FAN1$	$-RAD51$	29.4%	$\rightarrow$	32.9%	32.9%	67.1%
$-FAN1$	$-HRR$	29.4%	$\rightarrow$	32.2%	32.2%	67.8%
$-FAN1$	$-USP1$	29.4%	$\rightarrow$	32.9%	32.9%	67.1%
$-FAN1$	$-KU$	29.4%	$\rightarrow$	31.7%	31.7%	68.2%
$-FAN1$	$-DNAPK$	29.4%	$\rightarrow$	31%	31%	69%
$-FAN1$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-ADD$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-MRN$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-BRCA1$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-ssDNARPA$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-FANCD1N$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-RAD51$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-HRR$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%
$-USP1$	$-ATM$	29.4%	$\rightarrow$	100%	100%	0%

bullet		gain			$B_{physio1}$	B_{patho1}
$-ATM$	$-p53$	29.4%	$\rightarrow$	100%	100%	0%
$-ATM$	$-CHK1$	29.4%	$\rightarrow$	100%	100%	0%
$-ATM$	$-CHK2$	29.4%	$\rightarrow$	100%	100%	0%
$-ATM$	$-H2AX$	29.4%	$\rightarrow$	100%	100%	0%

Bibliography

[1] Arnaud Poret and Jean-Pierre Boissel. An in silico target identification using boolean network attractors: avoiding pathological phenotypes. *Comptes rendus biologies*, 337(12):661–678, 2014.

[2] Arnaud Poret and Jean-Pierre Boissel. Therapeutic target discovery using boolean network attractors: avoiding pathological phenotypes. HAL: `https://hal.archives-ouvertes.fr/hal-01024788`, arXiv: `http://arxiv.org/abs/1407.4374`, 2014.

[3] Arnaud Poret, Claudio Monteiro Sousa, and Jean-Pierre Boissel. Enhancing boolean networks with fuzzy operators and edge tuning. HAL: `https://hal.archives-ouvertes.fr/hal-01018236`, arXiv: `http://arxiv.org/abs/1407.1135`, 2014.

[4] Raymond Cheong and Andre Levchenko. Wires in the soup: quantitative models of cell signaling. *Trends in cell biology*, 18(3):112–118, 2008.

[5] Alex Mogilner, Roy Wollman, and Wallace F Marshall. Quantitative modeling in cell biology: what is it good for? *Developmental cell*, 11(3):279–287, 2006.

[6] David S Wishart, Craig Knox, An Chi Guo, Dean Cheng, Savita Shrivastava, Dan Tzur, Bijaya Gautam, and Murtaza Hassanali. Drugbank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic Acids Research*, 36(suppl 1):D901–D906, 2008.

[7] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

[8] M Whirl-Carrillo, EM McDonagh, JM Hebert, L Gong, K Sangkuhl, CF Thorn, RB Altman, and Teri E Klein. Pharmacogenomics knowledge for personalized medicine. *Clinical Pharmacology & Therapeutics*, 92(4):414–417, 2012.

[9] David Croft, Gavin O'Kelly, Guanming Wu, Robin Haw, Marc Gillespie, Lisa Matthews, Michael Caudy, Phani Garapati, Gopal Gopinath, Bijay Jassal, et al. Reactome: a database of reactions, pathways and biological processes. *Nucleic acids research*, 39(suppl 1):D691–D697, 2011.

[10] Xin Chen, Zhi Liang Ji, and Yu Zong Chen. Ttd: therapeutic target database. *Nucleic Acids Research*, 30(1):412–415, 2002.

[11] Hiroaki Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–1664, 2002.

[12] Jean-Pierre Boissel, Michel Cucherat, Patrice Nony, Francois Gueyffier, and Francois-Henri Boissel. Modeling and medical product r& d. In *International pharmaceutical product registration*, pages 762–773. Informa Healthcare, 2009.

[13] Jean-Pierre Boissel, Benjamin Ribba, Emmanuel Grenier, Guillemette Chapuisat, and Marie-Aimee Dronne. Modelling methodology in physiopathology. *Progress in biophysics and molecular biology*, 97(1):28–39, 2008.

[14] Hiroaki Kitano. Computational systems biology. *Nature*, 420(6912):206–210, 2002.

[15] Barbara Di Ventura, Caroline Lemerle, Konstantinos Michalodimitrakis, and Luis Serrano. From in vivo to in silico biology and back. *Nature*, 443(7111):527–533, 2006.

[16] M Ilea, M Turnea, M Rotariu, et al. Ordinary differential equations with applications in molecular biology. *Revista medico-chirurgicala a Societatii de Medici si Naturalisti din Iasi*, 116(1):347–352, 2011.

[17] Michelle L Wynn, Nikita Consul, Sofia D Merajver, and Santiago Schnell. Logic-based models in systems biology: a predictive and parameter-free network analysis method. *Integrative Biology*, 4(11):1323–1337, 2012.

[18] Melody K Morris, Julio Saez-Rodriguez, Peter K Sorger, and Douglas A Lauffenburger. Logic-based models for the analysis of cell signaling networks. *Biochemistry*, 49(15):3216–3224, 2010.

[19] Steven Watterson, Stephen Marshall, and Peter Ghazal. Logic models of pathway biology. *Drug discovery today*, 13(9):447–456, 2008.

[20] Silvia Von der Heyde, Christian Bender, Frauke Henjes, Johanna Sontag, Ulrike Korf, and Tim Beissbarth. Boolean erbb network reconstructions and perturbation simulations reveal individual drug response in different breast cancer cell lines. *BMC systems biology*, 8(1):75, 2014.

[21] Herman F Fumia and Marcelo L Martins. Boolean network model for cancer pathways: predicting carcinogenesis and targeted therapy outcomes. *PLOS ONE*, 8(7):e69008, 2013.

[22] Luca Grieco, Laurence Calzone, Isabelle Bernard-Pierrot, Francois Radvanyi, Brigitte Kahn-Perles, and Denis Thieffry. Integrative modelling of the influence of mapk network on cancer cell fate decision. *PLoS computational biology*, 9(10):e1003286, 2013.

[23] Oscar Andres Moreno-Ramos, Maria Claudia Lattig, and Andres Fernando Gonzalez Barrios. Modeling of the hypothalamic-pituitary-adrenal axis-mediated interaction between the serotonin regulation pathway and the stress response using a boolean approximation: a novel study of depression. *Theor. Biol. Med. Model*, 10:59, 2013.

[24] Hedi Peterson, Raed Abu Dawud, Abhishek Garg, Ying Wang, Jaak Vilo, Ioannis Xenarios, and James Adjaye. Qualitative modeling identifies il-11 as a novel regulator in maintaining self-renewal in human pluripotent stem cells. *Frontiers in physiology*, 4:303, 2013.

[25] Tomas Tokar, Zdenko Turcan, and Jozef Ulicny. Boolean network-based model of the bcl-2 family mediated momp regulation. *Theor Biol Med Model*, 10:40, 2013.

[26] Ozgur E Akman, Steven Watterson, Andrew Parton, Nigel Binns, Andrew J Millar, and Peter Ghazal. Digital clocks: simple boolean models can quantitatively describe circadian systems. *Journal of The Royal Society Interface*, page rsif20120080, 2012.

[27] Alfredo Rodriguez, David Sosa, Leda Torres, Bertha Molina, Sara Frias, and Luis Mendoza. A boolean network model of the fa/brca pathway. *Bioinformatics*, 28(6):858–866, 2012.

[28] Amit Singh, Juliana M Nascimento, Silke Kowar, Hauke Busch, and Melanie Boerries. Boolean approach to signalling pathway modelling in hgf-induced keratinocyte migration. *Bioinformatics*, 28(18):i495–i501, 2012.

[29] G Bhardwaj, CP Wells, R Albert, DB van Rossum, and RL Patterson. Exploring phospholipase c-coupled ca2+ signalling networks using boolean modelling. *IET systems biology*, 5(3):174–184, 2011.

[30] Chu C Kang, Yung J Chuang, Kai C Tung, Chun C Chao, Chuan Y Tang, Shih C Peng, and David SH Wong. A genetic algorithm-based boolean delay model of intracellular signal transduction in inflammation. *BMC bioinformatics*, 12(Suppl 1):S17, 2011.

[31] Amy L Bauer, Trachette L Jackson, Yi Jiang, and Thimo Rohlf. Receptor cross-talk in angiogenesis: mapping environmental cues to cell phenotype using a stochastic, boolean signaling network model. *Journal of theoretical biology*, 264(3):838–846, 2010.

[32] Aurelien Naldi, Jorge Carneiro, Claudine Chaouiya, and Denis Thieffry. Diversity and plasticity of th cell types predicted from regulatory network modelling. *PLoS computational biology*, 6(9):e1000912, 2010.

[33] Juilee Thakar and Reka Albert. Boolean models of within-host immune interactions. *Current opinion in microbiology*, 13(3):377–381, 2010.

[34] Hao Ge and Min Qian. Boolean network approach to negative feedback loops of the p53 pathways: synchronized dynamics and stochastic limit cycles. *Journal of Computational Biology*, 16(1):119–132, 2009.

[35] Zhongxing Mai and Haiyan Liu. Boolean network-based analysis of the apoptosis network: irreversible apoptosis and stable surviving. *Journal of Theoretical Biology*, 259(4):760–769, 2009.

[36] Ozgur Sahin, Holger Frohlich, Christian Lobke, Ulrike Korf, Sara Burmester, Meher Majety, Jens Mattern, Ingo Schupp, Claudine Chaouiya, Denis Thieffry, et al. Modeling erbb receptor-regulated g1/s transition to find novel targets for de novo trastuzumab resistance. *BMC Systems Biology*, 3(1):1, 2009.

[37] Regina Samaga, Julio Saez-Rodriguez, Leonidas G Alexopoulos, Peter K Sorger, and Steffen Klamt. The logic of egfr/erbb signaling: theoretical properties and analysis of high-throughput data. *PLoS computational biology*, 5(8):e1000438, 2009.

[38] Rebekka Schlatter, Kathrin Schmich, Ima Avalos Vizcarra, Peter Scheurich, Thomas Sauter, Christoph Borner, Michael Ederer, Irmgard Merfort, and Oliver Sawodny. On/off and beyond-a boolean model of apoptosis. *PLoS computational biology*, 5(12):e1000595, 2009.

[39] Maria I Davidich and Stefan Bornholdt. Boolean network model predicts cell cycle sequence of fission yeast. *PLOS ONE*, 3(2):e1672, 2008.

[40] Gwenael Kervizic and Laurent Corcos. Dynamical modeling of the cholesterol regulatory pathway with boolean networks. *BMC systems biology*, 2(1):99, 2008.

[41] Ranran Zhang, Mithun Vinod Shah, Jun Yang, Susan B Nyland, Xin Liu, Jong K Yun, Reka Albert, and Thomas P Loughran. Network model of survival signaling in large granular lymphocyte leukemia. *Proceedings of the National Academy of Sciences*, 105(42):16308–16313, 2008.

[42] Simone Gupta, Siddharth S Bisht, Ritushree Kukreti, Sanjeev Jain, and Samir K Brahmachari. Boolean network analysis of a neurotransmitter signaling pathway. *Journal of theoretical biology*, 244(3):463–469, 2007.

[43] Adrien Faure, Aurelien Naldi, Claudine Chaouiya, and Denis Thieffry. Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. *Bioinformatics*, 22(14):e124–e131, 2006.

[44] Luis Mendoza. A network model for the control of the differentiation process in th cells. *Biosystems*, 84(2):101–114, 2006.

[45] Sui Huang and Donald E Ingber. Shape-dependent control of cell growth, differentiation, and apoptosis: switching between attractors in cell regulatory networks. *Experimental Cell Research*, 261(1):91–103, 2000.

[46] Xiaowei Zhu, Mark Gerstein, and Michael Snyder. Getting connected: analysis and principles of biological networks. *Genes & Development*, 21(9):1010–1024, 2007.

[47] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

[48] Ilia Nikolaevich Bronshtein, Konstantin A Semendyayev, Gerhard Musiol, and Heiner Muehlig. Algorithms of graph theory. In *Handbook of Mathematics*, chapter 5, pages 348–359. Springer, 2007.

[49] Wolfgang Huber, Vincent J Carey, Li Long, Seth Falcon, and Robert Gentleman. Graphs in molecular biology. *BMC Bioinformatics*, 8(Suppl 6):S8, 2007.

[50] Oliver Mason and Mark Verwoerd. Graph theory and networks in biology. *Systems Biology, IET*, 1(2):89–119, 2007.

[51] Avi Ma'ayan. Insights into the organization of biochemical regulatory networks using graph theory analyses. *Journal of Biological Chemistry*, 284(9):5451–5455, 2009.

[52] Jill H Larkin and Herbert A Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive science*, 11(1):65–100, 1987.

[53] Thomas C Ings, Jose M Montoya, Jordi Bascompte, Nico Bluthgen, Lee Brown, Carsten F Dormann, Francois Edwards, David Figueroa, Ute Jacob, J Iwan Jones, et al. Review: ecological networks–beyond food webs. *Journal of Animal Ecology*, 78(1):253–269, 2009.

[54] Oswald J Schmitz and Andrew P Beckerman. Food webs. *eLS*, 2007.

[55] Yufei Xiao. A tutorial on analysis and simulation of boolean gene regulatory network models. *Current Genomics*, 10(7):511, 2009.

[56] John C Rockett and David J Dix. Gene expression networks. *eLS*, 2006.

[57] Assieh Saadatpour and Reka Albert. Boolean modeling of biological regulatory networks: a methodology tutorial. *Methods*, 2012.

[58] Stuart A Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3):437–467, 1969.

[59] Leon Glass. Classification of biological networks by their qualitative dynamics. *Journal of Theoretical Biology*, 54(1):85–107, 1975.

[60] Lee E Ostrander and Joe E Brand. Functional boolean models for systems with continuous variables. *Journal of theoretical biology*, 38(3):665–675, 1973.

[61] Rene Thomas. Boolean formalization of genetic control circuits. *Journal of theoretical biology*, 42(3):563–585, 1973.

[62] Reka Albert and Juilee Thakar. Boolean modeling: a logic-based dynamic approach for understanding signaling and regulatory networks and for making useful predictions. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 6(5):353–369, 2014.

[63] Rui-Sheng Wang, Assieh Saadatpour, and Reka Albert. Boolean modeling in systems biology: an overview of methodology and applications. *Physical biology*, 9(5):055001, 2012.

[64] Istvan Albert, Juilee Thakar, Song Li, Ranran Zhang, and Reka Albert. Boolean network simulations for life scientists. *Source code for biology and medicine*, 3(1):1–8, 2008.

[65] Stefan Bornholdt. Boolean network models of cellular regulation: prospects and limitations. *Journal of the Royal Society Interface*, 5(Suppl 1):S85–S94, 2008.

[66] Sui Huang. Genomics, complexity and drug discovery: insights from boolean network models of cellular regulation. *Pharmacogenomics*, 2(3):203–222, 2001.

[67] J Leclerco and Jacques Emile Dumont. Boolean analysis of cell regulation networks. *Journal of theoretical biology*, 104(4):507–534, 1983.

[68] George Boole. The mathematical analysis of logic, being an essay towards a calculus of deductive logic, 1847.

[69] Abhishek Garg, Alessandro Di Cara, Ioannis Xenarios, Luis Mendoza, and Giovanni De Micheli. Synchronous versus asynchronous modeling of gene regulatory networks. *Bioinformatics*, 24(17):1917–1925, 2008.

[70] Jonathan Knowles and Gianni Gromo. Target selection in drug discovery. *Nature Reviews Drug Discovery*, 2(1):63–69, 2003.

[71] Mark A Lindsay. Target discovery. *Nature Reviews Drug Discovery*, 2(10):831–838, 2003.

[72] Peter Imming, Christian Sinning, and Achim Meyer. Drugs, their targets and the nature and number of drug targets. *Nature Reviews Drug discovery*, 5(10):821–834, 2006.

[73] Andrew Anighoro, Jurgen Bajorath, and Giulio Rastelli. Polypharmacology: challenges and opportunities in drug discovery. *Journal of medicinal chemistry*, 57(19):7874–7887, 2014.

[74] Grant R Zimmermann, Joseph Lehar, and Curtis T Keith. Multi-target therapeutics: when the whole is greater than the sum of the parts. *Drug Discovery Today*, 12(1):34–42, 2007.

[75] Jackson B Gibbs. Mechanism-based target identification and drug discovery in cancer research. *Science*, 287(5460):1969–1973, 2000.

[76] KI Kaitin. Deconstructing the drug development process: the new face of innovation. *Clinical Pharmacology and Therapeutics*, 87(3):356, 2010.

[77] Jing Tang and Tero Aittokallio. Network pharmacology strategies toward multi-target anticancer therapies: from computational models to experimental design principles. *Current pharmaceutical design*, 20(1):20, 2014.

[78] Lora Mak, Sonia Liggi, Lu Tan, Kanthida Kusonmano, Judith M Rollinger, Alexios Koutsoukas, Robert C Glen, and Johannes Kirchmair. Anti-cancer drug development: computational strategies to identify and target proteins involved in cancer metabolism. *Current pharmaceutical design*, 19(4):532–577, 2013.

[79] Malabika Sarker, Carolyn Talcott, and Amit K Galande. In silico systems biology approaches for the identification of antimicrobial targets. In *In Silico Models for Drug Discovery*, pages 13–30. Springer, 2013.

[80] Lixia Yao. In silico search for drug targets of natural compounds. *Current pharmaceutical biotechnology*, 13(9):1632–1639, 2012.

[81] Praveen M Bahadduri, James E Polli, Peter W Swaan, and Sean Ekins. Targeting drug transporters–combining in silico and in vitro approaches to predict in vivo. In *Membrane Transporters in Drug Discovery and Development*, pages 65–103. Springer, 2010.

[82] Xiao Hua Ma, Zhe Shi, Chunyan Tan, Yuyang Jiang, Mei Lin Go, Boon Chuan Low, and Yu Zong Chen. In silico approaches to multitarget drug discovery. *Pharmaceutical research*, 27(5):739–749, 2010.

[83] Todor Vujasinovic, Andre Sinisa Zampera, Pascale Jackers, Despina Sanoudou, and Antoine Depaulis. In silico dynamic molecular interaction networks for the discovery of new therapeutic targets. *Current pharmaceutical design*, 16(20):2241–2251, 2010.

[84] Nagasuma Chandra. Computational systems approach for drug target discovery. *Expert opinion on drug discovery*, 4(12):1221–1236, 2009.

[85] Nadia Saidani, Delphine Grando, Helene Valadie, Olivier Bastien, and Eric Marechal. Potential and limits of in silico target discovery–case study of the search for new antimalarial chemotherapeutic targets. *Infection, Genetics and Evolution*, 9(3):359–367, 2009.

[86] Ulrik B Nielsen and Birgit Schoeberl. Using computational modeling to drive the development of targeted therapeutics. *IDrugs*, 8(10):822–826, 2005.

[87] D Malcolm Duckworth and Philippe Sanseau. In silico identification of novel therapeutic targets. *Drug discovery today*, 7(11):S64–S69, 2002.

[88] Denis Noble, Jeremy Levin, and William Scott. Biological simulations in drug discovery. *Drug Discovery Today*, 4(1):10–16, 1999.

[89] Nicolas Carels, Tatiana Tilli, and Jack A Tuszynski. A computational strategy to select optimized protein targets for drug development toward the control of cancer diseases. *PloS one*, 10(1):e0115054, 2015.

[90] Costa Frangou, Ying-Wei Li, He Shen, Nuo Yang, Kayla E Wilson, Maxime Blijlevens, Jin Guo, Norma J Nowak, and Jianmin Zhang. Molecular profiling and computational network analysis of taz-mediated mammary tumorigenesis identifies actionable therapeutic targets. *Oncotarget*, 5(23):12166–12176, 2014.

[91] Tan Li and Guanyu Wang. Computer-aided targeting of the pi3k/akt/mtor pathway: toxicity reduction and therapeutic opportunities. *International journal of molecular sciences*, 15(10):18856–18891, 2014.

[92] Daniel Nicklas and Leonor Saiz. In silico identification of potential therapeutic targets in the tgf-β signal transduction pathway. *Molecular BioSystems*, 10(3):537–548, 2014.

[93] Nageswara Rao Reddy Neelapu and T Pavani. Identification of novel drug targets in hpb38, hpp12, hpg27, hpshi470, hpsjm180 strains of helicobacter pylori: an in silico approach for therapeutic intervention. *Current drug targets*, 14(5):601–611, 2013.

[94] Bilachi S Ravindranath, Venkatappa Krishnamurthy, Venkatarangaiah Krishna, and Kahale Bommaiah Lingaiah Vasudevanayaka. In silico analyses of metabolic pathway and protein interaction network for identification of next gen therapeutic targets in chlamydophila pneumoniae. *Bioinformation*, 9(12):605, 2013.

[95] Wu-Lung R Yang, Yu-En Lee, Ming-Huang Chen, Kun-Mao Chao, and Chi-Ying F Huang. In silico drug screening and potential target identification for hepatocellular carcinoma using support vector machines based on drug screening result. *Gene*, 518(1):201–208, 2013.

[96] Sergio Iadevaia, Yiling Lu, Fabiana C Morales, Gordon B Mills, and Prahlad T Ram. Identification of optimal drug combinations targeting

cellular networks: integrating phospho-proteomics and computational network analysis. *Cancer research*, 70(17):6704–6714, 2010.

[97] F Betul Kavun Ozbayraktar and Kutlu O Ulgen. Drug target identification in sphingolipid metabolism by computational systems biology tools: metabolic control analysis and metabolic pathway analysis. *Journal of biomedical informatics*, 43(4):537–549, 2010.

[98] ON Koborova, DA Filimonov, AV Zakharov, AA Lagunin, SM Ivanov, A Kel, and VV Poroikov. In silico method for identification of promising anticancer drug targets. *SAR and QSAR in Environmental Research*, 20(7-8):755–766, 2009.

[99] Madhukar S Dasika, Anthony Burgard, and Costas D Maranas. A computational framework for the topological analysis and targeted disruption of signal transduction networks. *Biophysical journal*, 91(1):382–398, 2006.

[100] Pau Creixell, Erwin M Schoof, Janine T Erler, and Rune Linding. Navigating cancer network attractors for tumor-specific therapy. *Nature Biotechnology*, 30(9):842–848, 2012.

[101] Keith Baverstock. A comparison of two cell regulatory models entailing high dimensional attractors representing phenotype. *Progress in Biophysics and Molecular Biology*, 106(2):443–449, 2011.

[102] Wensheng Guo, Guowu Yang, Wei Wu, Lei He, and Mingyu Sun. A parallel attractor finding algorithm based on boolean satisfiability for genetic regulatory networks. *PloS one*, 9(4):e94258, 2014.

[103] Nikolaos Berntenis and Martin Ebeling. Detection of attractors of large boolean networks via exhaustive enumeration of appropriate subspaces of the state space. *BMC bioinformatics*, 14(1):361, 2013.

[104] D Zheng, G Yang, X Li, Z Wang, and WNN Hung. An efficient algorithm for finding attractors in synchronous boolean networks with

biochemical applications. *Genetics and Molecular Research: GMR*, 12(4):4656, 2013.

[105] Elena Dubrova and Maxim Teslenko. A sat-based algorithm for finding attractors in synchronous boolean networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 8(5):1393–1399, 2011.

[106] Ferhat Ay, Fei Xu, and Tamer Kahveci. Scalable steady state analysis of boolean biological regulatory networks. *PloS one*, 4(12):e7992, 2009.

[107] Charles J Sherr and Frank McCormick. The rb and p53 pathways in cancer. *Cancer Cell*, 2(2):103–112, 2002.

[108] Robert S Schwartz and Alan D D'Andrea. Susceptibility pathways in fanconi's anemia and breast cancer. *New England Journal of Medicine*, 362(20):1909–1919, 2010.

[109] Arleen D Auerbach. Fanconi anemia and its diagnosis. *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*, 668(1):4–10, 2009.

[110] Johan P de Winter and Hans Joenje. The genetic and molecular basis of fanconi anemia. *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*, 668(1):11–19, 2009.

[111] Kornelia Neveling, Daniela Endt, Holger Hoehn, and Detlev Schindler. Genotype-phenotype correlations in fanconi anemia. *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*, 668(1):73–91, 2009.

[112] Jiri Bartek and Jiri Lukas. Dna damage checkpoints: from initiation to recovery or adaptation. *Current opinion in cell biology*, 19(2):238–245, 2007.

[113] Kazuhiro Ishikawa, Hideshi Ishii, and Toshiyuki Saito. Dna damage-dependent cell cycle checkpoints and genomic stability. *DNA and cell biology*, 25(7):406–411, 2006.

[114] Makoto Nakanishi, Midori Shimada, and Hiroyuki Niida. Genetic instability in cancer cells by impaired cell cycle checkpoints. *Cancer science*, 97(10):984–989, 2006.

[115] Peican Zhu and Jie Han. Asynchronous stochastic boolean networks as gene network models. *Journal of Computational Biology*, 21(10):771–783, 2014.

[116] Jinghang Liang and Jie Han. Stochastic boolean networks: an efficient approach to modeling gene regulatory networks. *BMC systems biology*, 6(1):113, 2012.

[117] Nicholas Rescher. *Many-valued logic*. Springer, 1968.

[118] Ilia Nikolaevich Bronshtein, Konstantin A Semendyayev, Gerhard Musiol, and Heiner Muehlig. Fuzzy logic. In *Handbook of mathematics*, chapter 5, pages 360–378. Springer, 2007.

[119] Lotfi Asker Zadeh. Fuzzy logic. *Computer*, 21(4):83–93, 1988.

[120] Lotfi A Zadeh. From computing with numbers to computing with words. *Annals of the New York Academy of Sciences*, 929(1):221–252, 2001.

[121] Lotfi A Zadeh. Fuzzy logic = computing with words. *Fuzzy Systems, IEEE Transactions on*, 4(2):103–111, 1996.

[122] Melody K Morris, Julio Saez-Rodriguez, David C Clarke, Peter K Sorger, and Douglas A Lauffenburger. Training signaling pathway maps to biochemical data with constrained fuzzy logic: quantitative analysis of liver cell responses to inflammatory stimuli. *PLoS computational biology*, 7(3):e1001099, 2011.

[123] Bree B Aldridge, Julio Saez-Rodriguez, Jeremy L Muhlich, Peter K Sorger, and Douglas A Lauffenburger. Fuzzy logic analysis of kinase pathway crosstalk in tnf/egf/insulin-induced signaling. *PLoS computational biology*, 5(4):e1000340, 2009.

[124] Zuyi Huang and Juergen Hahn. Fuzzy modeling of signal transduction networks. *Chemical Engineering Science*, 64(9):2044–2056, 2009.

[125] Ezequiel Franco-Lara and Dirk Weuster-Botz. Application of fuzzy-logic models for metabolic control analysis. *Journal of theoretical biology*, 245(3):391–399, 2007.

[126] Pan Du, Jian Gong, Eve Syrkin Wurtele, and Julie A Dickerson. Modeling gene expression networks using fuzzy logic. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 35(6):1351–1359, 2005.

[127] Bogju Lee, John Yen, Linyu Yang, and James C Liao. Incorporating qualitative knowledge in enzyme kinetic models using fuzzy logic. *Biotechnology and bioengineering*, 62(6):722–729, 1999.

[128] Ilia Nikolaevich Bronshtein, Konstantin A Semendyayev, Gerhard Musiol, and Heiner Muehlig. Boolean algebras and switch algebra. In *Handbook of mathematics*, chapter 5, pages 342–347. Springer, 2007.

[129] Kanae Oda, Yukiko Matsuoka, Akira Funahashi, and Hiroaki Kitano. A comprehensive pathway map of epidermal growth factor receptor signaling. *Molecular systems biology*, 1(1), 2005.

[130] Marcello Buiatti and Giuseppe Longo. Randomness and multilevel interactions in biology. *Theory in Biosciences*, 132(3):139–158, 2013.

[131] Mukhtar Ullah and Olaf Wolkenhauer. Stochastic approaches in systems biology. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 2(4):385–397, 2010.

[132] CE Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[133] Alex Rhee, Raymond Cheong, and Andre Levchenko. The application of information theory to biochemical signaling systems. *Physical biology*, 9(4):045011, 2012.

[134] Christian Waltermann and Edda Klipp. Information theory based approaches to cellular signaling. *Biochimica et Biophysica Acta (BBA)-General Subjects*, 1810(10):924–932, 2011.

[135] Walter Benenson and Horst Stocker. Measurements and measurement errors. In *Handbook of physics*, chapter 30, pages 1097–1113. Springer, 2002.

[136] Faiz M Khan, Ulf Schmitz, Svetoslav Nikolov, David Engelmann, Brigitte M Putzer, Olaf Wolkenhauer, and Julio Vera. Hybrid modeling of the crosstalk between signaling and transcriptional networks using ordinary differential equations and multi-valued logic. *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, 1844(1):289–298, 2014.

[137] Regina Samaga and Steffen Klamt. Modeling approaches for qualitative and semi-quantitative analysis of cellular signaling networks. *Cell Commun Signal*, 11:43, 2013.

[138] Leon Glass and Stuart A Kauffman. The logical analysis of continuous, non-linear biochemical control networks. *Journal of theoretical Biology*, 39(1):103–129, 1973.

[139] Colin Campbell and Reka Albert. Stabilization of perturbed boolean network attractors through compensatory interactions. *BMC systems biology*, 8(1):53, 2014.

[140] Yushan Qiu, Takeyuki Tamura, Wai-Ki Ching, and Tatsuya Akutsu. On control of singleton attractors in multiple boolean networks: integer programming-based method. *BMC systems biology*, 8(Suppl 1):S7, 2014.

[141] Sriganesh Srihari, Venkatesh Raman, Hon Wai Leong, and Mark A Ragan. Evolution and controllability of cancer networks: a boolean perspective. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 11(1):83–94, 2014.

[142] Xi Chen, Tatsuya Akutsu, and Wai-Ki Ching. Finding optimal control policy in probabilistic boolean networks with hard constraints by using integer programming and dynamic programming. *International Journal of Data Mining and Bioinformatics*, 7(3):1–22, 2013.

[143] Koichi Kobayashi and Kunihiko Hiraishi. Optimal control of gene regulatory networks with effectiveness of multiple drugs: A boolean network approach. *BioMed research international*, 2013, 2013.

[144] Koichi Kobayashi and Kunihiko Hiraishi. Symbolic approach to verification and control of deterministic/probabilistic boolean networks. *IET systems biology*, 6(6):215–222, 2012.

[145] Alan C Hindmarsh. Odepack, a systematized collection of ode solvers. *IMACS transactions on scientific computation*, 1:55–64, 1983.