



HAL
open science

A logic-based modeling derived from boolean networks: adding fuzzy logic and edge tuning

Arnaud Poret, Claudio Monteiro Sousa, Jean-Pierre Boissel

► To cite this version:

Arnaud Poret, Claudio Monteiro Sousa, Jean-Pierre Boissel. A logic-based modeling derived from boolean networks: adding fuzzy logic and edge tuning. 2014. hal-01018236v2

HAL Id: hal-01018236

<https://hal.science/hal-01018236v2>

Preprint submitted on 29 Jul 2014 (v2), last revised 25 May 2015 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A logic-based modeling derived from boolean networks: adding fuzzy logic and edge tuning

Arnaud Poret^{1,2}, Claudio Monteiro Sousa^{1,2}
Jean-Pierre Boissel¹

June 29, 2014

© 2014 Arnaud Poret, Claudio Monteiro Sousa, Jean-Pierre Boissel. This document is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit <https://creativecommons.org/licenses/by/4.0/>.

Arnaud Poret (corresponding author): arnaud.poret@gmail.com
Claudio Monteiro Sousa: claudio.ms@universite-lyon.fr
Jean-Pierre Boissel: jean-pierre.boissel@novadiscovery.com

¹ Novadiscovery, 60 Avenue Rockefeller, 69008 Lyon, France, <http://www.novadiscovery.com>

² UMR CNRS 5558, 43 Boulevard du 11 Novembre 1918, 69622 Villeurbanne cedex, France, <https://lbbe.univ-lyon1.fr/>

Abstract

Quantitative modeling in biology can be difficult due to the scarcity of parameter values. An alternative is qualitative modeling since it requires few to no parameters. This article presents a qualitative modeling derived from boolean networks where fuzzy logic is used and where edges can be tuned. Fuzzy logic being continuous, its variables can be finely valued while remaining qualitative. To consider that some interactions are slower or weaker than other ones, edge states are computed to modulate in speed and strength the signal they convey. The proposed formalism is illustrated through its implementation on an example network. The simulations show that continuous results are produced, thus allowing a fine analysis, and that modulating the signal conveyed by the edges allows their tuning according to knowledge about the interaction they model. The present work is expected to bring enhancements in the ability of qualitative models to simulate biological networks.

Contents

1	Introduction	3
2	Methods	4
2.1	Basic principles	4
2.1.1	Biological networks	4
2.1.2	Boolean networks	4
2.1.3	Fuzzy logic	5
2.2	The proposed logic-based modeling	6
2.2.1	Additional features	6
2.3	Implementation	8
2.3.1	Example network	8
2.3.2	Fuzzy operators	9
2.3.3	Additional features	9
3	Results	10
3.1	Simulation 1	10
3.2	Simulation 2	11
3.3	Simulation 3	11
3.4	Simulation 4	12
3.5	Simulation 5	14
4	Conclusion	14
5	Annexes	20

1 Introduction

Quantitative modeling in systems biology allows scientists to produce formal models of the biological system of interest and then to implement it on computers [1, 2]. 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 *in vitro* and *in vivo* ones. However, the stumbling block of *in silico* experiments is that their validity depends on the knowledge used to build the model: 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 [3], KEGG [4], PharmGKB [5], Reactome [6] and TTD [7]. In addition to the complexity of integrating an increasing body of knowledge comes the inherent complexity of biological systems themselves [8]: this is where computational tools can help [9].

The interplay between traditional and computational biology is synergistic rather than competing [10]. Since *in vitro* and *in vivo* experiments produce rather factual results, they are a relatively trustworthy source of knowledge. Once these factual pieces of knowledge are obtained, computational models can help to integrate them and infer new knowledge through *in silico* experiments. This computationally obtained knowledge can be subsequently used to direct further *in vitro* or *in vivo* experiments, hence mutually potentiating the whole.

One of the main difficulties encountered when quantitatively modeling biological systems with, for example, systems of differential equations [11], is that the required parameter values are not straightforward to obtain. One solution to overcome this barrier is qualitative modeling since it requires few to no parameters [12]. Several qualitative modelings already exist and are mostly based on logic [13, 14] such as the well known boolean networks [15, 16] which are based on boolean logic [17]. 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 modeling approaches at their disposal, each with its advantages and disadvantages, depending on the pursued goals and available resources. If accurate numerical results are expected, a quantitative modeling is required. However, if tendencies and global properties are the main concerns, a qualitative modeling is entirely fitting and proved itself through several work [18–29] to cite a few.

The present work starts from boolean networks, pioneered in biology by Kauffman [30], Glass [31], Ostrander [32] and Thomas [33]. However, boolean logic is replaced with fuzzy logic [34, 35]. The basic principles remain the same: given a biological network [36, 37], entities are modeled by variables and interactions by functions of these variables allowing their value to be updated at each iteration of the simulation.

Fuzzy logic is continuous and allows its variables to be valued at any real between 0 and 1, that is to consider all the possible truth degrees between the absolutely true and the definitively false. This is the basis of possibility theory [38] which considers this continuum of truth degrees as possibilities. The results provided by fuzzy logic, while remaining qualitative, can be finer than those provided by the boolean one where variables can only be valued at 0 or 1, namely at true or false. 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 [39]. However, in some other cases where things are not necessarily binary, such as in signaling pathways where enzymes can be more or less active, fuzzy logic can be an interesting choice.

In addition to the use of fuzzy logic to qualitatively model biological networks, some additional features are added in order to capture more aspects of their possible behaviors. These additional features concern the edges of the network, that is the interactions taking place between the entities composing it. In this case, edges are seen as conveyors of signals corresponding to the influences exerted by the entities of the network onto other ones. This signal, together with its modulation, are taken into account: edges can be tuned. To do so, edge states are computed, in addition to node states, and the signal they convey can be slowed or weakened. The result is a logic-based qualitative modeling intended to bring a fine qualitative quantification of biological network behaviors.

Talking about qualitative quantification can appear somewhat contradictory but is common in thinking processes which are, needless to say, 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, or to state that this enzyme is active is moderately true: quantification is expressed but only with perceptions and tendencies. Indeed, qualitative quantification is expressed by words rather than measurements, hence its qualitative nature, and is characteristic of fuzzy logic [40, 41].

2 Methods

This section briefly introduces some basic principles and then describes how the proposed logic-based modeling is built. An example network to illustrate it together with its implementation are also described.

2.1 Basic principles

The proposed logic-based modeling is introduced starting from boolean networks. This section briefly introduces i) biological networks, what is modeled, ii) boolean networks, the starting point and iii) fuzzy logic, an added value.

2.1.1 Biological networks

A network can be seen as a graph $G = (V, E)$ where $V = \{v_1, \dots, v_n\}$ is the set of size n containing exactly all the nodes v_i of the network and where $E = \{(v_{i,1}, v_{j,1}), \dots, (v_{i,m}, v_{j,m})\} \subseteq V^2$ is the set of size m containing exactly all the edges (v_i, v_j) of the network [42, 43]. In practice, nodes represent things and edges represent binary relations $R \subseteq V^2$ involving these things: $v_i R v_j$. For example, in gene regulatory networks, nodes represent gene products and edges represent gene expression modulations.

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 $b_i \in \llbracket 0, n \rrbracket$ inputs $x_{i,1}, \dots, x_{i,b_i}$. The variables which are not input of x_i have no direct influence on it. In the case where $b_i = 0$, x_i is a parameter, or an input, and does not depend on other variables. At each iteration $k \in \llbracket k_0, k_{end} \rrbracket$ of the computation, the value $x_i(k) \in \{0, 1\}$ of each x_i is updated to the value $x_i(k+1)$ thanks to a boolean function f_i and to the values $x_{i,1}(k), \dots, x_{i,b_i}(k)$ of its inputs:

```

1 for  $k \in \llbracket k_0, k_{end} - 1 \rrbracket$  do
2    $x_1(k+1) = f_1(x_{1,1}(k), \dots, x_{1,b_1}(k))$ 
3   ...
4    $x_n(k+1) = f_n(x_{n,1}(k), \dots, x_{n,b_n}(k))$ 
5 end for

```

which can be written in a more concise form:

```

1 for  $k \in \llbracket k_0, k_{end} - 1 \rrbracket$  do
2    $\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k))$ 
3 end for

```

where $\mathbf{f} = (f_1, \dots, f_n)$ is the boolean transition function and $\mathbf{x} = (x_1, \dots, x_n)$ is the state vector. In the particular case where $k = k_0$, $\mathbf{x}(k_0) = \mathbf{x}_0$ is the initial state. If the values of all the x_i are updated simultaneously at each k then the network is synchronous, otherwise it is asynchronous.

2.1.3 Fuzzy logic

The main difference between boolean and fuzzy logic is that the first one is discrete, that is valued in $\llbracket 0; 1 \rrbracket \subset \mathbb{N}$, whereas the second one is continuous, that is valued in $[0; 1] \subset \mathbb{R}$. Fuzzy logic can be seen as an extension, a generalization, of the boolean one. To do so, the fuzzy counterparts of the boolean operators have to behave like them on $\llbracket 0; 1 \rrbracket$ but have to be defined on $[0; 1]$. With fuzzy logic, 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*:

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

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 [44] 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 the variables of fuzzy logic to take their value in $[0; 1]$ they can be equal to 1 (*true*), 0 (*false*) or all the other reals 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 of the variables represent, dealing with fuzzy logic enables to consider all the intermediate levels/states of what is modeled without leaving the qualitative modeling framework.

2.2 The proposed logic-based modeling

To build the proposed logic-based modeling from boolean networks, the boolean operators *AND*, *OR* and *NOT* have to be replaced with 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 $[0; 1]$:

$$f_i : [0; 1]^n \rightarrow [0; 1] : \mathbf{x} \mapsto f_i(\mathbf{x})$$

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

$$\mathbf{f} : [0; 1]^n \rightarrow [0; 1]^n : \mathbf{x} \mapsto \mathbf{f}(\mathbf{x})$$

Algorithmically, the computation remains the same as the one of boolean networks.

2.2.1 Additional features

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

Edge computation In the same way as for 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 the present work, this is made explicit in order to modulate the signal conveyed by the edges.

As a consequence, 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 in order to tune the signal conveyed by the edges. 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 accepts the value $e_{ij}(k)$ of e_{ij} itself as argument to compute its state $e_{ij}(k+1)$ at the next iteration. 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} :

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

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

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

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

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

On the basis of the computation of boolean networks described above, the computation becomes:

```

1 for  $k \in \llbracket k_0, k_{end} - 1 \rrbracket$  do
2   ...
3    $e_{ij}(k+1) = f_{ij}^{edge}(x_i(k), e_{ij}(k))$ 
4   ...
5    $x_i(k+1) = f_i^{node}(\dots, e_{ij}(k), \dots)$ 
6   ...
7 end for
```

which can be written in a more concise form:

```

1 for  $k \in \llbracket k_0, k_{end} - 1 \rrbracket$  do
2    $\mathbf{e}(k+1) = \mathbf{f}^{edge}(\mathbf{x}(k), \mathbf{e}(k))$ 
3    $\mathbf{x}(k+1) = \mathbf{f}^{node}(\mathbf{e}(k))$ 
4 end for
```

Edge reactivity The additional feature *edge reactivity* is implemented by a 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 high reactive edge has a p_{ij} close to 1 while a low reactive edge has a p_{ij} close to 0.

Biologically, *edge reactivity* can model the fact that some biological interactions can be slower, or of higher inertia, than other ones. For example, an edge representing a gene expression activation of a gene product by a transcription factor should have a lower p_{ij} than an edge representing an activating phosphorylation of an enzyme by another enzyme. Indeed, gene expression is a more complex mechanism than an activating phosphorylation and then takes more time to be accomplished.

Edge weakening The additional feature *edge weakening* is implemented by a 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 the signal $x_i(k)$ conveyed by e_{ij} is weakened: a strong edge has a q_{ij} close to 1 while a weak edge has a q_{ij} close to 0.

Biologically, *edge weakening* can model the fact that some biological interactions can be weaker than other ones. For example, an edge representing an activation of a receptor by a partial agonist should have a lower q_{ij} than an edge representing an activation of a receptor by a full agonist.

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} :

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

2.3 Implementation

To illustrate the proposed logic-based modeling at work, it is implemented on an example network with GNU Octave¹. The code is available on GitHub² at <https://github.com/arnaudporet/kali-sim> under a BSD 3-Clause License³.

It should be noted that, in the present 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 simply to visualize sequences of events linked by causal connections without true time quantification, a sort of qualitative time. To do so, $k_0 = 1$ and $k_{end} = 50$: 50 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 : $e_{ij}(k_0) = x_i(k_0)$.

2.3.1 Example network

The example network is a tiny sample of the epidermal growth factor receptor signaling pathway [45] adapted from [13]. A digital electronic like representation of it is shown in figure 1 page 9. This example network is chosen for its simplicity: simple enough to be mentally computed in order to easily judge the produced results. The boolean equations of the example network plus the corresponding f_{ij}^{edge} and f_i^{node} are listed in annexes page 20.

¹<https://www.gnu.org/software/octave/>

²<https://github.com/>

³https://raw.githubusercontent.com/arnaudporet/kali-sim/master/BSD_3-Clause

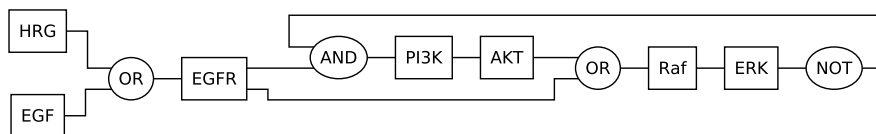


Figure 1: Digital electronic like representation of the example network. Nodes are rectangles whereas logical gates are ellipses. The graph should be read from left to right. For example, the node *PI3K* is an input of the node *AKT* and the node *ERK*, due to a feedback loop, is an input of the node *PI3K*. Logical gates are not nodes: as such, edges only, and eventually, pass through them. For example, the edge (*ERK*, *PI3K*) passes through a *NOT* gate then an *AND* gate whereas the edge (*Raf*, *ERK*) does not pass through any logical gate.

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 the present work, the algebraic formulation is used:

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

one of the most simple and convenient mathematical formulations of the fuzzy operators.

2.3.3 Additional features

$p_{ij} \in [0; 1]$: its value can be set to any infinitely accurate real of $[0; 1]$. But p_{ij} is a parameter belonging to qualitative modeling, particularly suitable to escape the scarcity of parameter values frequently encountered in biological quantitative modeling. Rather than requiring to precisely valueate p_{ij} as with quantitative modeling, its value is randomly picked in specified intervals of $[0; 1]$ along a uniform distribution introducing, at the same time, a little of a rudimentary stochasticity. To do so, $[0; 1]$ is split into intervals reflecting possible 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 $[0.5; 0.75]$ along a uniform distribution. Of course, one can split $[0; 1]$ in a different way.

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: $r \in \mathbb{N}^*$ runs are performed and the results are superposed. In the present 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	$p_{ij} = 1$
weak	$p_{ij} \in [0.75; 1]$
weaker	$p_{ij} \in [0.5; 0.75]$
faint	$p_{ij} \in [0.25; 0.5]$
fainter	$p_{ij} \in [0; 0.25]$
down	$p_{ij} = 0$

and

full	$p_{ij} = 1$
much more	$p_{ij} \in [0.75; 1]$
much	$p_{ij} \in [0.5; 0.75]$
few	$p_{ij} \in [0.25; 0.5]$
fewer	$p_{ij} \in [0; 0.25]$
none	$p_{ij} = 0$

plus the entire interval $[0; 1]$ in case of an undetermined *edge weakening*/initial state. Again, one can split $[0; 1]$ in a different way.

3 Results

This section presents results produced with the implementation of the example network through five simulations. Although the obtained curves are continuous due to the use of fuzzy logic, 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.

3.1 Simulation 1

EGF and *HRG* are the two inputs of the example network 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 is assumed to be at the resting state: $\mathbf{x}_0 = (\dots, none, \dots)$. The p_{ij} are set to *fast* and the q_{ij} are set to *strong*. The results are shown in figure 2 page 11. As expected, before *EGF* activation, the network is at rest: the signaling cascade is not active. However, once *EGF* activated, the signaling

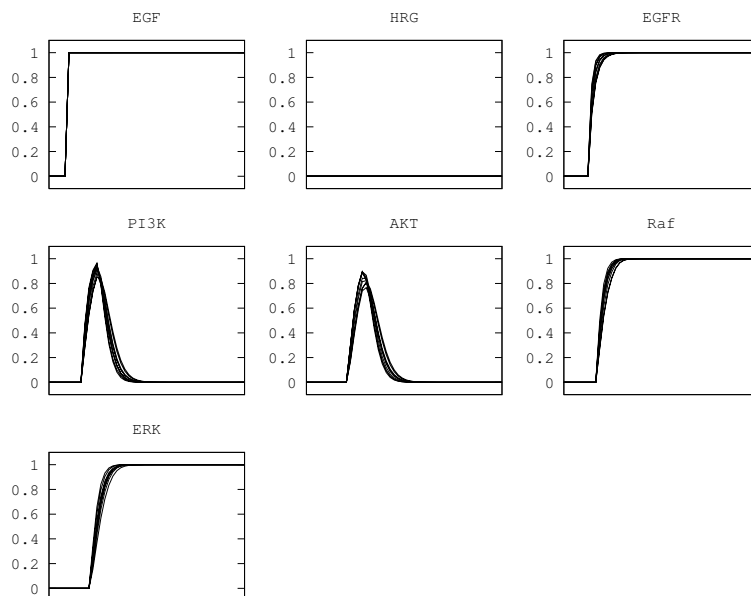


Figure 2: Activation of the signaling cascade by *EGF* and subsequent inhibition of *PI3K* by *ERK*.

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.

3.2 Simulation 2

In addition to the inputs, a perturbation is applied to the network. It consists in disabling the inhibitory effect of *ERK* on *PI3K*, that is in disabling the edge (*ERK*, *PI3K*). It points out an advantage of computing edge states: disturbing a node disturbs all its effects in the network while selectively disturbing edges prevents this.

The parameter values are the same as in simulation 1 except $q_{ERK,PI3K}$ which is set to *weaker*. With $q_{ERK,PI3K} = \textit{weaker}$ the signal conveyed by the edge (*ERK*, *PI3K*) is weakened throughout this simulation. The results are shown in figure 3 page 12. As expected, the weakening of the edge (*ERK*, *PI3K*) results in a weakened inhibition of *PI3K* by *ERK*: *ERK* does not totally inhibit *PI3K*.

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 remain the same as in simulation

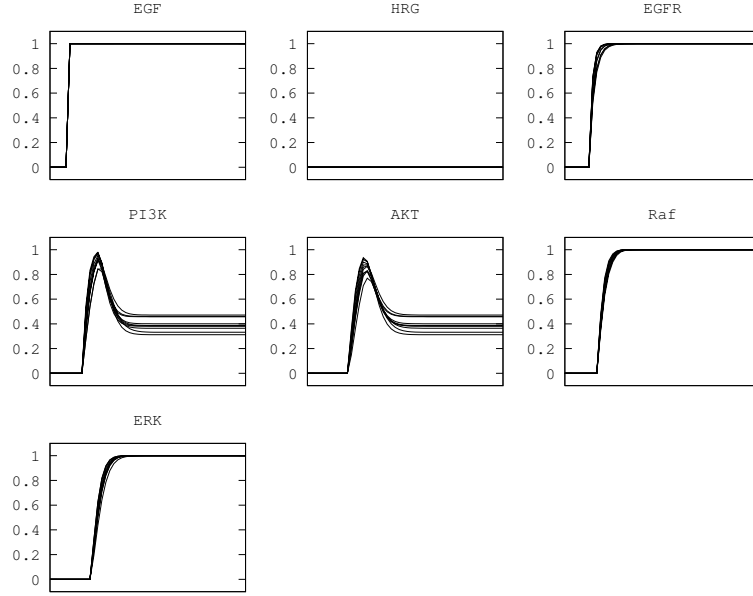


Figure 3: Weakening the inhibitory effect of *ERK* on *PI3K*.

1. With $p_{ERK,PI3K} = \textit{slower}$ the signal conveyed by the edge (*ERK*, *PI3K*) is slowed throughout this simulation. The results are shown in figure 4 page 13. As expected, the slowing of 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 $p_{ERK,PI3K} = \textit{fast}$.

3.4 Simulation 4

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

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

$$HRG(k+1) = none$$

The results are shown in figure 5 page 13. As expected, the activation of *EGF* is not total and therefore the same applies to the 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.

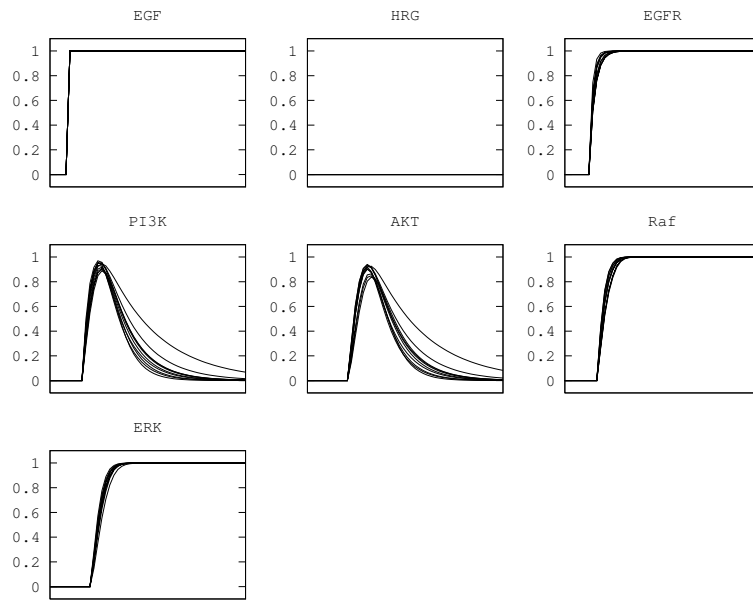


Figure 4: Slowing the inhibitory effect of *ERK* on *PI3K*.

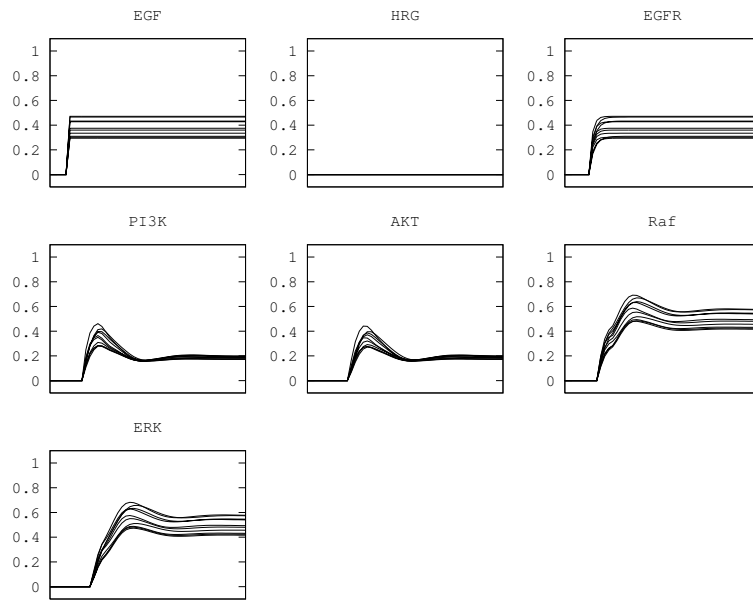


Figure 5: Consequences on the signaling cascade of a partial activation of *EGF*.

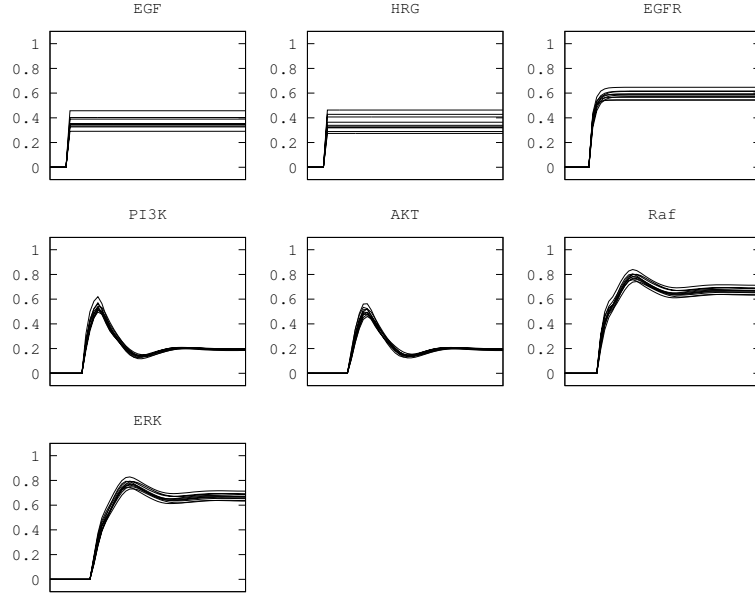


Figure 6: Cumulative effect of partly activated *EGF* and *HRG* on the signaling cascade.

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 remaining the same as in simulation 1. The results are shown in figure 6 page 14. 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*.

4 Conclusion

The simulations performed with the implementation of the example network show that the proposed logic-based modeling produces continuous results while remaining qualitative. The continuous nature of the produced results is due to the use of fuzzy logic and allows qualitative variables to be more finely valued

than with discrete qualitative modelings such as boolean networks. Moreover, thanks to the additional features *edge reactivity* and *edge weakening* attributed to each edge of the network, 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, in a given biological system, some interactions can be weaker or slower relative to other ones and therefore be more realistic in their qualitative modeling. A little of stochasticity on the two qualitative parameters *edge reactivity* and *edge weakening* is also realized through a random selection of their value in specified intervals followed by replication and superposition of the produced results. This stochasticity, although very rudimentary, should yield some more realism since events taking place in biological systems can themselves be subjected to stochasticity [46, 47]. It probably represents one of the most relevant lines of further improvements which could be done on the present work. Another relevant line of further improvement could be to apply information theory [48] to the signal conveyed by the edges as previously introduced for cell signaling [49, 50]. Altogether, starting from boolean networks and still founded on their basic principles, the present work is expected to bring a fine qualitative quantification of biological network behaviors.

A qualitative quantification remains qualitative and should not be confused with a true quantification which involves measurements, values and units [51]. The qualitative quantification proposed by the present work has the goal to bring some enhancements in the ability of qualitative models to simulate biological network behaviors. One of the main goals, and advantages, of qualitative modeling remains to propose an alternative to, but not a replacement of, quantitative modeling when the frequently encountered parameter value scarcity in systems biology makes the work unreasonably, or unnecessarily, difficult. The choice between a qualitative and quantitative modeling is up to the modeler, depending on its needs and resources.

It is also possible to use qualitative and quantitative modeling in combination. For example, qualitative modeling can be used to explore the global properties of the biological system of interest and then quantitative modeling can be used to focus on particular aspects. Knowing the relative difficulty of quantitative modeling in systems biology, this two-steps approach could make modeling more efficient by highlighting where to deploy quantitative modeling. Qualitative and quantitative modeling can also be merged into hybrid models [52, 53] which attempt to exploit the advantages of these two modelings in one. Hybrid models, or semi-quantitative models, can be good compromises between the convenience of qualitative modeling and the accuracy of quantitative modeling.

References

- [1] Denis Noble. Biological computation. *eLS*, 2002.
- [2] Ralf Hofestädt and Sven Thelen. Quantitative modeling of biochemical networks. *In Silico Biology*, 1(1):39–53, 1998.
- [3] 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.
- [4] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.
- [5] 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.
- [6] 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.
- [7] Xin Chen, Zhi Liang Ji, and Yu Zong Chen. Ttd: therapeutic target database. *Nucleic acids research*, 30(1):412–415, 2002.
- [8] Hiroaki Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–1664, 2002.
- [9] Hiroaki Kitano. Computational systems biology. *Nature*, 420(6912):206–210, 2002.
- [10] 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.
- [11] 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.
- [12] 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.
- [13] 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.
- [14] Steven Watterson, Stephen Marshall, and Peter Ghazal. Logic models of pathway biology. *Drug discovery today*, 13(9):447–456, 2008.
- [15] Stefan Bornholdt. Boolean network models of cellular regulation: prospects and limitations. *Journal of the Royal Society Interface*, 5(Suppl 1):S85–S94, 2008.

- [16] Assieh Saadatpour and Réka Albert. Boolean modeling of biological regulatory networks: a methodology tutorial. *Methods*, 2012.
- [17] George Boole. The mathematical analysis of logic, being an essay towards a calculus of deductive reasoning, originally published in cambridge by macmillan, barclay, & macmillan. *Reprinted in Oxford by Basil Blackwell*, page 175, 1847.
- [18] Herman F Fumiã and Marcelo L Martins. Boolean network model for cancer pathways: Predicting carcinogenesis and targeted therapy outcomes. *PLoS one*, 8(7):e69008, 2013.
- [19] Adrien Fauré, Aurélien 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.
- [20] Luis Mendoza. A network model for the control of the differentiation process in th cells. *Biosystems*, 84(2):101–114, 2006.
- [21] 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.
- [22] Aurélien 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.
- [23] Gwenael Kervizic and Laurent Corcos. Dynamical modeling of the cholesterol regulatory pathway with boolean networks. *BMC systems biology*, 2(1):99, 2008.
- [24] Luca Grieco, Laurence Calzone, Isabelle Bernard-Pierrot, François Radvanyi, Brigitte Kahn-Perlès, and Denis Thieffry. Integrative modelling of the influence of mapk network on cancer cell fate decision. *PLoS computational biology*, 9(10):e1003286, 2013.
- [25] Özgür Sahin, Holger Fröhlich, Christian Löbke, 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.
- [26] Ranran Zhang, Mithun Vinod Shah, Jun Yang, Susan B Nyland, Xin Liu, Jong K Yun, Réka 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.
- [27] 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.

- [28] 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.
- [29] 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.
- [30] Stuart A Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3):437–467, 1969.
- [31] Leon Glass. Classification of biological networks by their qualitative dynamics. *Journal of Theoretical Biology*, 54(1):85–107, 1975.
- [32] Lee E Ostrander and Joe E Brand. Functional boolean models for systems with continuous variables. *Journal of theoretical biology*, 38(3):665–675, 1973.
- [33] René Thomas. Boolean formalization of genetic control circuits. *Journal of theoretical biology*, 42(3):563–585, 1973.
- [34] Lotfi Asker Zadeh. Fuzzy logic. *Computer*, 21(4):83–93, 1988.
- [35] Il'ia Nikolaevich Bronshtein, Konstantin A Semendyayev, Gerhard Musiol, and Heiner Muehlig. Fuzzy logic. In *Handbook of mathematics*, chapter 5, pages 360–378. Springer, 2007.
- [36] Xiaowei Zhu, Mark Gerstein, and Michael Snyder. Getting connected: analysis and principles of biological networks. *Genes & development*, 21(9):1010–1024, 2007.
- [37] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.
- [38] Lotfi A Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 100:9–34, 1999.
- [39] Yufei Xiao. A tutorial on analysis and simulation of boolean gene regulatory network models. *Current genomics*, 10(7):511–525, 2009.
- [40] Lotfi A Zadeh. Fuzzy logic = computing with words. *Fuzzy Systems, IEEE Transactions on*, 4(2):103–111, 1996.
- [41] Lotfi A Zadeh. From computing with numbers to computing with words. from manipulation of measurements to manipulation of perceptions. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 46(1):105–119, 1999.
- [42] Wolfgang Huber, Vincent J Carey, Li Long, Seth Falcon, and Robert Gentleman. Graphs in molecular biology. *BMC bioinformatics*, 8(Suppl 6):S8, 2007.

- [43] Il'ia 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.
- [44] Il'ia 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.
- [45] 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.
- [46] Marcello Buiatti and Giuseppe Longo. Randomness and multilevel interactions in biology. *Theory in Biosciences*, 132(3):139–158, 2013.
- [47] Mukhtar Ullah and Olaf Wolkenhauer. Stochastic approaches in systems biology. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 2(4):385–397, 2010.
- [48] CE Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.
- [49] Christian Waltermann and Edda Klipp. Information theory based approaches to cellular signaling. *Biochimica et Biophysica Acta (BBA)-General Subjects*, 1810(10):924–932, 2011.
- [50] Alex Rhee, Raymond Cheong, and Andre Levchenko. The application of information theory to biochemical signaling systems. *Physical biology*, 9(4):045011, 2012.
- [51] Walter Benenson and Horst Stöcker. Measurements and measurement errors. In *Handbook of physics*, chapter 30, pages 1097–1113. Springer, 2002.
- [52] 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.
- [53] Regina Samaga and Steffen Klamt. Modeling approaches for qualitative and semi-quantitative analysis of cellular signaling networks. *Cell Commun Signal*, 11:43, 2013.

5 Annexes

The boolean equations of the example network. *AND*, *NOT* and *OR* stand for the boolean operators.

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

The f_{ij}^{edge} built from the boolean equations of the example network. *AND*, *NOT* and *OR* stand for the fuzzy operators.

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

The f_i^{node} built from the boolean equations of the example network. *AND*, *NOT* and *OR* stand for the fuzzy operators.

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

It should be noted that f_{EGF}^{node} and f_{HRG}^{node} do not accept any $e_{ij}(k)$ as argument. This is because f_{EGF}^{node} and f_{HRG}^{node} are associated to the inputs of the network: having themselves no inputs, they are set by the modeler through their f^{node} .