Accounting for Post-Transcriptional Regulation in Boolean Networks Based Regulatory Models.

Accounting for Post-Transcriptional Regulation in Boolean Networks Based Regulatory Models

Alfredo Benso[1], Stefano Di Carlo[1], Hafeez Ur Rehman[1], Gianfranco Politano[1], Alessandro Savino[1], Giovanni Squillero[1], Alessandro Vasciaveo[1], and Stefano Benedettini[2]

[1] Politecnico di Torino, Control and Comp. Engineering Department, Torino (Italy)
E-mail: <firstname>.<lastname>@polito.it,
Home page: http://www.testgroup.polito.it/index.php/bio-menu-home
[2] European Centre for Living Technology, Venice, Italy
E-mail: s.benedettini@unive.it

Abstract. Boolean Networks are emerging as a simple yet powerful formalism to model and study Gene Regulatory Networks. Nevertheless, the most widely used Boolean Network-based models do not include any post-transcriptional regulation mechanism. In this paper we discuss how the post-transcriptional regulation mechanism mediated by miRNAs can be included in a Boolean Network based model. This contribution constitutes a key preparatory step in the study of the topological and structural role of miRNAs in complex regulatory networks.

Keywords: miRNA, Gene Regulatory Networks, post-transcriptional regulation, Boolean Networks, Complex Systems, Network Analysis

1 Introduction

Remarkable advances in molecular biology enabled in the recent years to sequence the complete genome of several living organisms [18], and to identify and functionally annotate several of the thousand of proteins these genomes encode [4, 15]. Unfortunately the identification of the genome is only the first step in understanding how it works. Researchers are now switching to the next major challenge consisting in the determination of all interactions among genes, proteins and other cellular components [2]. Complex Systems Biology aims at systematically studying complex interactions among components of biological systems, by means of theoretical instruments provided by the science of complex systems [10]. In this context, the definition of models and computational methods to understand gene regulatory networks (GRN) is a primary objective.

In biological systems, decisions are reached and actions are taken by methods that are exceedingly parallel and extraordinarily integrated [12]. To understand the nature of cellular functions, it is necessary to study the behavior of genes in a holistic rather than in an individual manner because the expressions and activities of genes are not isolated or independent of each other [16]. There have been a number of attempts to model GRN, including linear models [17]

and Bayesian Networks [14]. However the model that probably received most attention is the *Boolean Network* (BN) model originally introduced by Kauffman about 30 years ago [11]. A Boolean network consists of a set of Boolean variables whose state is determined by other variables in the network. When used to model GRNs, gene expression is quantized into a Boolean value and the expression of each gene is constructed as a Boolean function of the expression of other genes. This approach is obviously a strong simplification, since in reality the expression of a gene is a continuous value and not a boolean (on/off) one. Nevertheless, Boolean Networks allow us to study high-level properties of a network like its robustness to noise, or its behavior under different initial conditions.

Recent researches suggest that several realistic biological questions may be studied by looking at this simple Boolean formalism and in particular computing and analyzing the related network attractors (i.e., a state or a set of states towards which a system, that is moving according to its dynamic, evolves over time) [13] [9]. However, most published models focus on a very high-level gene/gene or gene/protein interaction, neglecting post-transcriptional regulatory activities carried out by small non-coding RNA sequences such as micro RNA (miRNA). miRNA and non-coding RNA have demonstrated to play a central role in how the genome is regulated and how traits are passed on or eliminated by environmental and genetic factors.

In this paper we discuss how post-transcriptional regulatory interaction mediated by miRNA can be modeled by means of Boolean Networks, and how attractors of a network can be identified taking into account the peculiarity of this regulatory activity. The main contribution of the paper is therefore a more realistic representation of the cell regulatory activity that will in turn improve the exploratory power provided by the BN formalism.

2 Materials and Methods

At the molecular scale, the DNA of a specific organism encodes all the necessary information to synthesize RNAs and proteins. The interactions among genes and their products are at the basis of all vital cellular processes, like, for instance, cell metabolism, signal transduction, cell differentiation, cell fates and so on. Figure 1 shows a simple example of cellular regulatory activity in which the different actors that will be considered in our BN-based post-transcriptional model are introduced.

In the example, two genes, G1 and G2 are responsible for the biosyntheses of two mRNA molecules (mRNA1 and mRNA2) that convey the genetic information required to be translated into proteins P1 and P2. P1 represents an outcome of this specific sample regulatory activity whereas P2 works as an upstream promoter of gene G3 (i.e., G2 is a transcription factor of gene G3) thus giving shape to a simple GRN. RNAs are not the only molecules produced during gene transcription. miRNAs are often produced during transcription as reported in our sample GRN where G1 also produces a microRNA molecule named miRNA1. miRNAs are short non-coding RNA sequences that act as post-transcriptional

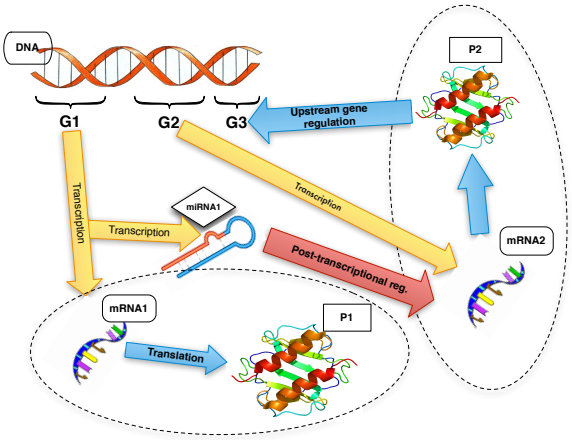

Fig. 1: Modeling of post-transcriptional mechanisms.

regulators. They bind to complementary sequences on target mRNAs, usually resulting in translational repression or target degradation. In our specific example miRNA1 acts as a post-transcriptional repressor of mRNA1 which results in a translation repression of P2 and therefore in an inhibition effect on gene G3.

Even if simple, the proposed example allows us to visualize all the actors required to introduce post-transcriptional regulation into GNR models. Figure 2 shows an extended BN representing the proposed regulatory example.

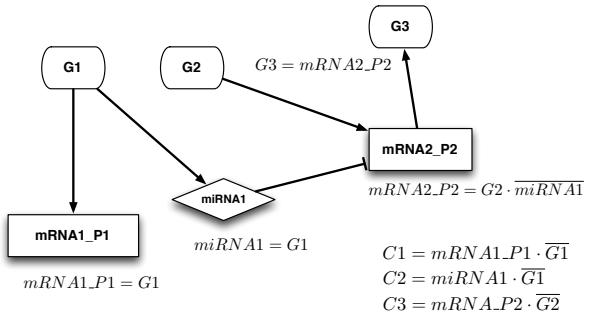

Fig. 2: Boolean network model.

For simplicity, the translation process that leads to the protein production starting from the related mRNA molecule has not been explicitly modeled. Three types of nodes are therefore used to model the GRN: (1) *genes*, (2)

mRNA_Protein pairs, and (3) *miRNA*. This is not a limitation, since, if necessary, the proposed approach could be easily extended to explicit the translation process.

The post-transcriptional regulation model is obtained by carefully designing the state function of each transcriptional product targeted by a miRNA (i.e., the *mRNA2_P1* node in the previous example). Post-transcriptional regulation acts at mRNA level, hence considering the final protein production it has higher priority compared to gene expression activity. In terms of boolean network, it can be modeled by placing the miRNA expression status in Boolean AND with the mRNA expression function.

Modeling the miRNA activity into the mRNA node update function of the BN is however not enough to properly model post-transcriptional activity, especially when the simulation of the dynamics of the BN is used to search for network attractors. If we define the state of the system as the value each node of the BN assumes at a given time, the search algorithm of network attractors traces a trajectory in the state space starting from an initial state and flowing through a set of intermediate states until an attractor is identified. This is done by simulating the dynamics of the system.

However, in gene regulatory networks not all states are biologically acceptable. While the dynamics of the network, if well designed, avoid to move the system into an illegal state, if such a state is selected as an initial condition in the attractor search process, biological meaningless results could be inferred. As an example, let us consider G1 and the related protein *mRNA1_P1*. A protein can be expressed only if the related gene has been expressed thus triggering the transcription/translation process. In this example, any state in which *mRNA1_P1* is equal to 1 (expressed) while G1 is equal to 0 (not expressed) is actually an illegal state and should be discarded.

We therefore propose to complement the description of the BN with a set of conditions identifying illegal states of the networks. In Figure 2 these conditions are represented by the three Boolean equations C1, C2, and C3. Every time a state of the network is considered, the three conditions must be evaluated. The state is considered legal if all conditions return zero, illegal otherwise.

There are a number of software applications for experimenting with BNs [5, 1, 8]. Some of them are too narrow in scope, inefficient or difficult to integrate. Since our primary need was to integrate the BN representation, we found the Boolean Network Toolkit (BNT), presented in [3], very easy to customize because of the C++ open implementation of its core engine. Resorting to the BTN, we implemented an attractor search procedure based on the extended BN representation proposed in Figure 1. In the BNT core, a boolean network is a direct graph implemented using adjacent lists supported by the BOOST C++ Library, [7]. Each node in this structure contains different data such as the node name, type (e.g., gene, protein or miRNA), and other parameters useful to characterize the node, also from a graphical point of view. The graphical framework chosen to represent the network is Cytoscape [6]. Algorithm 1 shows the overall attractor search process.

Given the high number of states a complex BN may assume, attractors are searched by selecting a limited number of candidate initial conditions, whose number is identified by the *probes* parameter. The attractor search procedure is an iterative process involving *probes* iteration. For each iteration, a random initial state is selected, the validity of the state against the transcriptional and post-transcriptional constraints is verified, and if the state is admissible, the network dynamics are simulated to search for an attractor on the selected trajectory.

```
Data: the Gene Regulatory Network
Result: a Set which contains the attractors
input  : probes the number of initial condition to test
forall the probes do
    σ ← generateRandomState;
    isValid ← isValidBooleanState(σ,C);
    if isValid = true then
        anAttractor ← findAttractor(σ);
        grnAttractorSet.add(anAttractor);
    else
        continue;
    end
end
```

Algorithm 1: Algorithm for searching for a GRN attractor.

To check whether a state is valid or not, the isValidBooleanState function (see Algorithm 2) is used. This function receives in input the state to analyze and the set of constraints established for the network. Each constraint represents a boolean function that is evaluated in the target state. If at least one of the constraints is true, it means the state is not valid and the initial state is discarded. Otherwise, a true value is returned.

3 Results and Discussion

Figure 3 shows the four attractors found for the simple network of Figure 2. Gray nodes are active or expressed, while white nodes are silenced. It is easy to see that all attractors fulfill the "post-transcriptional" constraints. Therefore, they are all valid network states. In this phase, besides adding the post-transcriptional regulation mechanism to the GRN model, we concentrated in making the toolset as flexible and computationally fast as possible. To better profile the scalability of the toolset we applied the attractors search algorithm on a set of artificially generated GRNs that include post-transcriptional regulation. The challenge here was also to considerably increase the manageable network size in order to be able, eventually, to analyze real regulatory networks of significant size. Experiments

Result: if the state is admissible or not
input : X the state to check
input : stateConstraintSet the collection of constraints for the given network
output: true if the state being checked is admissible
output: false otherwise

notValid ← **false**;
foreach anInvalidSchema *in* stateConstraintSet **do**
  notValid ← isValid *OR* anInvalidSchema (X) ;
end
if *notValid* = **true then**
  return **false** ;
else
  return **true** ;
end

Algorithm 2: Algorithm for checking the validity of an initial state.

were performed on a 8-core workstation featuring multithreading programming on three different network types:

- *dense networks*: each node has an average in/out degree (number of input/output edges) equal to 25;
- *sparse networks*: each node has an average in/out degree equal to 5;
- *scale-invariant networks*: each node has an average in/out degree equal to 5; a select number of nodes are hubs with in/out degree greater than 25;

For each class we generated four networks with increasing number of nodes: 10, 20, 50, and 100 nodes. For each network we applied the attractor search algorithm considering increasing number of initial probes (10K, 100K, 1M states), and each experiment has been repeated 10 times to account for the casualty of the initial state generation. The attractor search algorithm explores the network states exploiting all available cores. In this way, the search is 8-times faster then in a single-core implementation. The only actual limitation is the memory consumption since the search complexity considerably increases with the number of nodes and edges of the network. With 8GB of available RAM we noticed a performance breakdown when the GRN configuration reaches 100 nodes with an average number of incoming edges per node higher than 29.

The results reported in Figure 4 show that it is possible to find the attractors of networks up to 100 nodes in a reasonable time (Real Time is lower than CPU Time thanks to the multithreading approach).

4 Conclusions

The BNT presented in this paper is only a first step towards a more realistic analysis of the high-level functional and topological characteristics of GRNs. Resorting to the tool facilities, such as multicore implementation and support for common input/output formats, dynamics of real networks of significant size

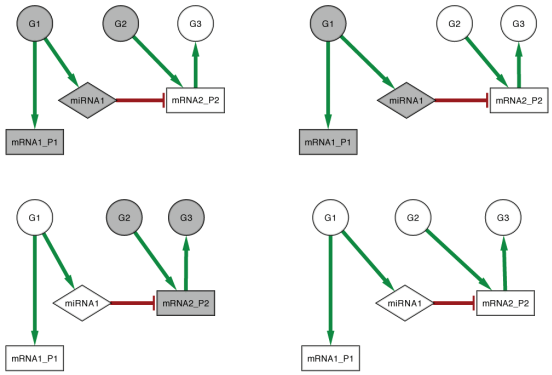

Fig. 3: Set of attractors found for the Network of Figure 2

can be analyzed. Moreover, thanks to the extended model that includes post-transcriptional regulation, not only the network simulation can be more reliable, but also it can offer new insights on the role of miRNAs from a functional as well as structural point of view. To do this, the toolkit will be soon improved with several network analysis algorithms.

References

1. Albert, I., Thakar, J., Li, S., Zhang, R., Albert, R.: Boolean network simulations for life scientists. Source Code for Biology and Medicine 3(1), 16 (2008)
2. Albert, R.: Boolean modelingof genetic regulatory networks. Complex Networks pp. 459–481 (2004)
3. Benedettini, S., Roli, A.: An efficient simulator for boolean network models. In: European Conference on Complex Systems (2012)
4. Benso, A., Di Carlo, S., urRehman, H., Politano, G., Savino, A.: Using gnome wide data for protein function prediction by exploiting gene ontology relationships. In: Automation Quality and Testing Robotics (AQTR), 2012 IEEE International Conference on. pp. 497 –502 (2012)
5. BNSim. `http://code.google.com/p/bnsim/` (2011), viewed: November 2011
6. Cytoscape: Cytoscape: An open source platform for complex network analysis and visualization. [Available Online]: http://www.cytoscape.org
7. Dawes, B., Abrahams, D., Rivera, R.: Boost c++ libraries. [Available Online]: http://www.boost.org/doc/libs/
8. Gershenson, C.: RBNLab. `http://sourceforge.net/projects/rbn/` (2011), viewed: November 2011
9. Huang, S., Ernberg, I., Kauffman, S.: Cancer attractors: a systems view of tumors from a gene network dynamics and developmental perspective. Seminars in cell & developmental biology 20(7), 869–876 (Sep 2009), `http://dx.doi.org/10.1016/j.semcdb.2009.07.003`

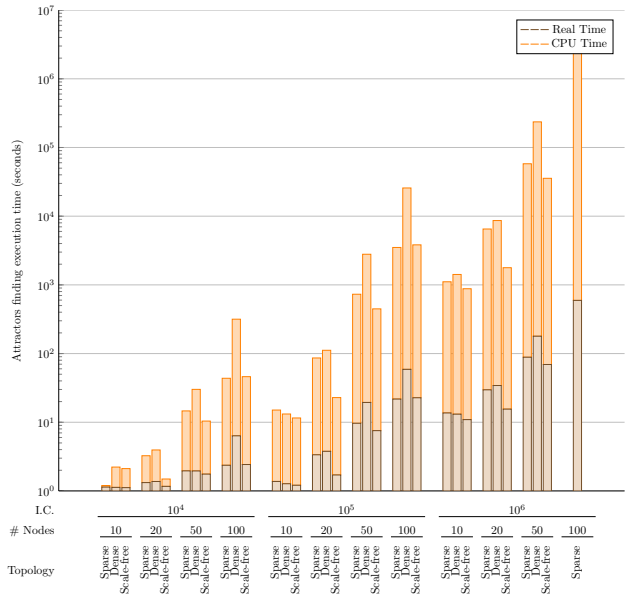

Figure 1: **Boolean Network Toolkit** (running on 9 threads) —
HW: Intel(R) Core(TM) i7-2670QM CPU @ 2.20GHz ; 8GB RAM 1333MHz

Fig. 4: Computational performances for the selected experiments.

10. Kaneko, K.: Life: An introduction to complex systems biology, vol. 171. Springer Heidelberg, Germany: (2006)
11. Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. J Theor Biol 22(3), 437–67 (Mar 1969)
12. Lähdesmäki, H., Shmulevich, I., Yli-Harja, O.: On learning gene regulatory networks under the boolean network model. Machine Learning 52(1), 147–167 (2003)
13. Luo, J.X., Turner, M.S.: Evolving sensitivity balances boolean networks. PLoS One 7(5), e36010 (2012)
14. Moler, E.J., Radisky, D.C., Mian, I.S.: Integrating naive bayes models and external knowledge to examine copper and iron homeostasis in s. cerevisiae. Physiol Genomics 4(2), 127–135 (Dec 2000)
15. ur Rehman, H., Benso, A., Di Carlo, S., Politano, G., Savino, A., Suravajhala, P.: Combining homolog and motif similarity data with gene ontology relationships for protein function prediction. In: IEEE International Conference on Bioinformatics and Biomedicine (2012)
16. Shmulevich, I., Dougherty, E., Kim, S., Zhang, W.: Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks. Bioinformatics 18(2), 261–274 (2002)
17. van Someren, E.P., Wessels, L.F., Reinders, M.J.: Linear modeling of genetic networks from experimental data. Proc Int Conf Intell Syst Mol Biol 8, 355–66 (2000)
18. Werner, T.: Next generation sequencing in functional genomics. Brief Bioinform 11(5), 499–511 (Sep 2010)