Computer application for simulation of gene regulatory networks

M. Sc. O. Simov[1], Prof. A. Dika[2], PhD, Prof B. Percinkova[3], PhD,
[1]SEE University, Republic of Macedonia,
[2]SEE University, Republic of Macedonia,
[3]European University, Republic of Macedonia

Abstract:
The DNA micro array chips allow measuring of the expression of thousands of genes simultaneously. The experimental data obtained during such measurements may be used for a functional understanding of genome dynamics by means of mathematical modelling[5]. The network which controls the expression of the genes is called Genetic Regulatory Network – GRN.

In this work we are present computer application which simulate GRN. In the presented application, two models for representing GRN are supported. First model is Boolean GRN[1], second model is extended Boolean GRN model.

First model represent pure Boolean GRN. There are algorithms for transform entire state space with defined number of transitions, find numbers of basins of attraction, basins of attraction structures and periodical cycles, hamming distances[2] between beginning states after defined number of transitions and differences what appear after minimal perturbations. GRN can be generated randomly or custom defined by user.

Second model represent extended Boolean GRN. With this model is possible to calculate proteins concentration after defined number of time steps and defined start position. GRN can be generated randomly or custom defined by user.

1. Introduction

The Genetic regulatory networks -GRN have a management role in cells. A GRN is a set of genes and other components of a cell, which interact directly or indirectly with each other and thus control the rate of gene expression. Proteins that regulate GRN are called transcription factors and these can only activate or repress expression of specific genes.

While in multi-cellular organisms all cells have the same complement of DNA molecules, the function of differentiated cells depends on the set of proteins that are synthesized in them and these are determined by the genes that are expressed in them.

In single-cell organisms, the Genetic regulatory networks manages each cell, optimizing its functioning in response to the external environment by activating or deactivating specific genes.

Through representing of GRN with appropriate mathematic model and simulation of the same with computer software, the behavior of the cell is possible to be predicted in different conditions. The more the model is real the more accurate results will be obtained in the simulations. The precision of the proposed build up model may be adjusted through the topology of the graph, the initial concentration of the proteins, the number of the newly created proteins after expression of an appropriate gene, the functions which define the dynamics of degradation of the proteins and the threshold of influence.

2. Boolean GRN and Appropriate Simulation Tool

The Boolean regulatory networks can model GRN together with gene products (output) and substances from the environment (input) which can influence the genetic regulatory process. Stuart Kauffman [1] was among the first to use Boolean networks for modeling genetic regulatory processes.

In this concept a Genetic regulatory networks is represented by an oriented graph G=(V,E), where V is a set of nodes which represent components of the network, and E is a set of oriented edges which connects the nodes and the set of Boolean functions. E defines the topology of GRN. Nodes represent genes or other biomolecules which are not genes or gene products, but which influence the Genetic regulatory networks.

$$2^N$$

If there is a K value of inputs in a node, the total number of possible Boolean functions that can represent the output of that node in the next step shall be:

$$(2^2)^K = 2^{2K}$$

If one imagines the Genetic regulatory network as a Finite State Automate - FSA, or more precisely as a Deterministic Finite State Automate – DFSA, in certain conditions the GRN will switch from one state to another, a process which is called transition. Since there is a finite number of states, after some time, the Genetic regulatory network will enter into a state in which it has been before and following that, because it is deterministic, it will remain in a loop of states. The number of states in a loop

Manuscript received August 5, 2010
Manuscript revised August 20, 2010

is called a “length of cycle” and its value can amount from 1 to total number of
states 2N. These states constitute dynamical attractors of Boolean GRN. The set of states flowing into one cycle of states constitutes the basin of attraction of that states cycle. These networks have at least one dynamical attractor, however often they have more then one.

This is an example of a simple Boolean GRN:

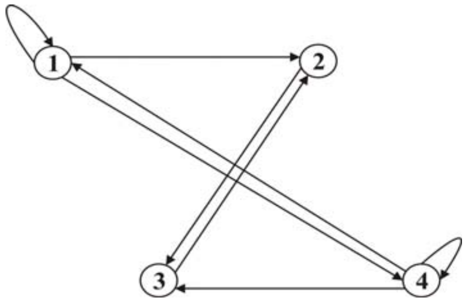
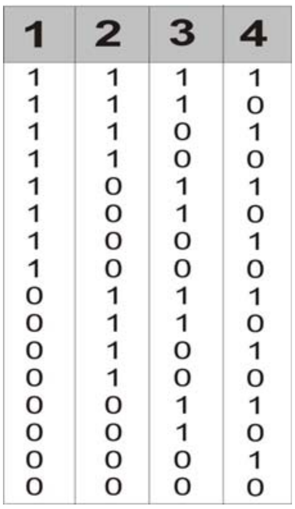


Figure 1. Simple Boolean Genetic regulatory network with three nodes

There are four nodes in this simple Boolean GRN. The state vector consists of three values which can be either 1 or 0. The total number of states is 24=16. The sixteen possible states shall be:

Table 1. The total set of states for GRN with three nodes



1	2	3	4
1	1	1	1
1	1	1	0
1	1	0	1
1	1	0	0
1	0	1	1
1	0	1	0
1	0	0	1
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	1
0	1	0	0
0	0	1	1
0	0	1	0
0	0	0	1
0	0	0	0

All nodes have a number of inputs K=2. The total number of possible Boolean functions which will describe the state of the node in the following step is 2^{2K}, for this case 16. Some possible of functions are[6]:

Table 2. Possible Boolean functions

1	2	1 or 2
1	1	1
1	0	1
0	1	1
0	0	0

1	2	1 and 2
1	1	1
1	0	0
0	1	0
0	0	0

1	2	1 xor 2
1	1	0
1	0	1
0	1	1
0	0	0

1	2	1 if 2
1	1	1
1	0	1
0	1	0
0	0	1

Let us choose these functions to describe the nodes output for GRN given in figure 1:

Table 3. Boolean functions for calculating states of nodes dependent upon the previous states of inputs

2	3	1'=f(1,2)
1	1	1
1	0	0
0	1	0
0	0	1

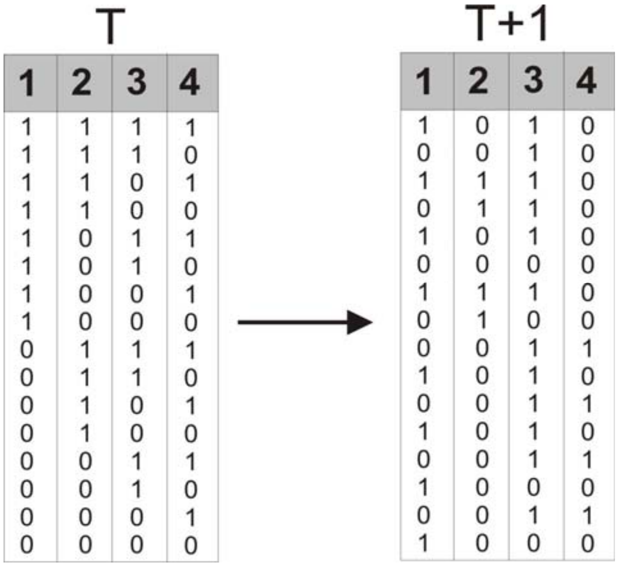
1	3	2'=f(1,3)
1	1	0
1	0	1
0	1	0
0	0	0

1	2	3'=f(2,4)
1	1	1
1	0	1
0	1	1
0	0	0

		4'=f(1,4)
1	1	0
1	0	0
0	1	1
0	0	0

Table 4. Transition of entire state space

and topology, one can calculate the future state of nodes depending upon the present state of inputs. The calculations for a complete set of states after one transition are as follows:



T

1	2	3	4
1	1	1	1
1	1	1	0
1	1	0	1
1	1	0	0
1	0	1	1
1	0	1	0
1	0	0	1
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	1
0	1	0	0
0	0	1	1
0	0	1	0
0	0	0	1
0	0	0	0

T+1

1	2	3	4
1	0	1	0
0	0	1	0
1	1	1	0
0	1	1	0
1	0	1	0
0	0	0	0
1	1	1	0
0	1	0	0
0	0	1	1
1	0	1	0
0	0	1	1
1	0	1	0
0	0	1	1
1	0	0	0
0	0	1	1
1	0	0	0

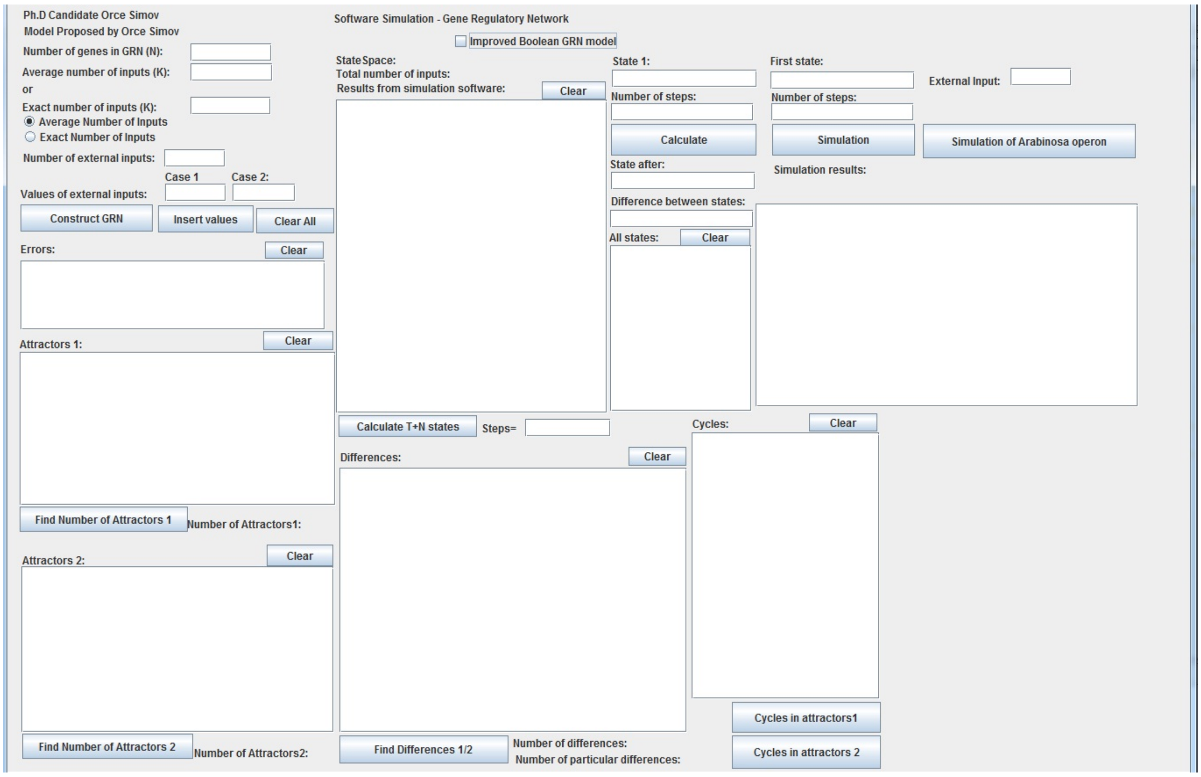


Figure 2. Interface of computer application

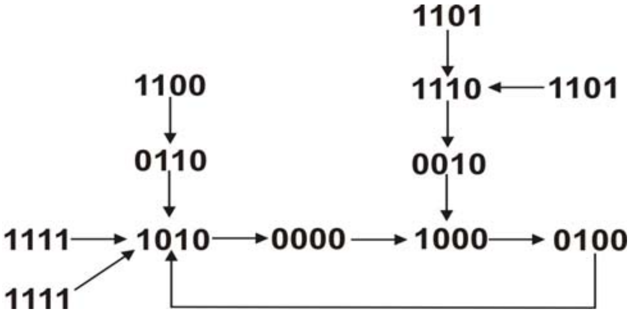


Figure 3. Basin of attraction 1

The presented GRN from the example has two basins of attraction:

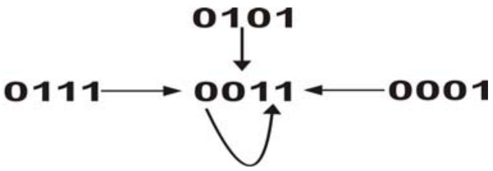


Figure 4. Basin of attraction 2

If this system is left on its own, depending upon the initial state, after some time it will enter into one of the two periodic state cycles. These state cycles are Dynamic attractors of GRN.

The first periodic cycle has four states. The 1010 state transit into 0000, 0000 transit into 1000, 1000 transit in 0100 and 0100 transit in first state of periodic cycle 1010.

The second periodic cycle has one state 0011 and
with transition it came back to the same state 0011.

If we want to use software for simulation of pure Boolean GRN model, check box “Improved Boolean GRN model” should not be checked.

This computer application can:

- create random GRN network
- accept user definition of GRN
- Find number of basins of attraction
- Find constitution of basins of attraction with proper position of every state
- Find periodic cycles in basins of attraction and their length
- Compute state after defined number of transitions
- Compute situation of states after transition of

entire state space for defined number of steps
Compute hamming distances between two states after defined number of transitions

3. Extended Booleean GRN Model and Appropriate Sumulation Tool

The proposed model for representing a Gene regulatory network basically has Boolean model of genetic regulatory network [1] proposed by S. Kauffman, in which there have been made several changes given in the following part.

In a system, in which we have N genes, it may be produced N different types of proteins. During each transition of the state, it is expressed a different genes, and it is produced an appropriate proteins. The concentration (the number) of the proteins of each type is kept as vector:

$$P=\{P_1, P_2, P_3, \ldots.. P_N\}$$

After the transition, during obtaining the new state of the nodes, it is updated also the value of the number of the proteins.

Expression of gene mean that information from DNA are transcribed and mRNA molecule is created. Then in the process of translation, the written data of mRNA molecule are read and the protein is synthesized. Depending on the lifetime of the mRNA molecule and the content of structural elements in the environment, a creation of larger number of proteins after one expression of the appropriate gene is possible.

In the vector:

$$S=\{Sp_1, Sp_2, Sp_3, \ldots.. Sp_N\}$$

we keep the data for the number of the newly synthesized proteins if appropriate gene is expressed.

In addition to the process of synthesis of proteins, there is a process of degradation of proteins. After a certain lifetime, different for each protein, they disintegrate. The concentration of a specific protein is determined by the balance of the synthesis and the degradation of the protein. The lack of balance in the synthesis and the degradation of the proteins leads to hypotrophy (loss of proteins in the cell), if the rate of degradation is bigger than the rate of synthesis and hypertrophy (increase of concentration of proteins in the cell), if the rate of synthesis is bigger than the rate of degradation. The regulatory proteins has lifetime from 5 to 120 minutes [7]. The regulatory proteins with short lifetime disintegrate through local proteolytic mechanisms. The proteins with short lifetime are marked with identifying marks, which allow their identification from the proteases and degradation. There have been identified a number of molecular identifying signals, but those mechanisms are not yet completely disclosed. Special mechanisms of degradation exist for the proteins, which contain translational or post translational errors, or they are damaged in some other way[7].

The degradation of the proteins in the proposed model is projected by existence of a system for decreasing the number of the proteins which exist in the system. The number of disintegrated proteins of a certain type is a function of the type of the protein and its concentration. The number of disintegrated proteins in one step is proportional of the concentration of such protein. If the protein is regulatory, the functions which allow faster degradation are used. The functions of degradation for each protein is kept in the vector:

$$D=\{fdp_1, fdp_2, fdp_3, \ldots.. fd_N\}$$

where fdp_i is a function which determines the rate of degradation of the protein **i**, proportionally with its concentration. In the proposed model there have been given several functions for definition of degradation, which can be assigned separately for each protein. For the regulatory proteins there have been used functions, which provide larger rate of degradation, i.e. shorter lifetime. The inappropriate functions for degradation lead to enormous decrease and increase of the concentration of specific proteins, which in reality leads to extinction of the cell.

The number of the proteins is calculated with the following formula:

$$Pi(t+1)=Pi(t)+Si(t)-fdpi(t)$$

Where:

Pi(t+1) – number of the protein **i** in the following step,

Pi(t) - number of the protein **i** in the moment,

Si(t) - number of the newly synthesized molecules of the protein **i,** and

fdpi(t) – function which gives the number of the disintegrated proteins of type **i.**

If the number of the newly synthesized proteins is bigger than the number of disintegrated proteins, the total number of proteins will increase. If the number of the newly synthesized proteins is smaller than the number of disintegrated proteins, the total number of proteins will decrease.

In the model it is also proposed the existence of threshold of influence. The threshold of influence is different for each protein. The regulatory protein will have influence over the expression of the genes, which it regulates, if its concentration is bigger or equal to that given in the threshold of influence. The threshold of influence is given with the vector:

$$Th=\{Th_1, Th_2, Th_3, \ldots..Th_N\}$$

The influence of the node corresponding with an appropriate protein, i.e. with an appropriate gene, in the following states for nodes, for which it represents an input, will be with value one, if the number of proteins of that type is bigger or equal to the threshold of influence. The same will be zero, if the number of proteins is smaller than the threshold of influence, notwithstanding its current value is one or zero. Its current value has influence only over the number of the appropriate protein. If its value is one, with the process of transcription it will be created mRNA molecule, during which it may be synthesized more than one protein. The number of proteins which will be synthesized is given in the vector S.

This model can be used for calculation of protein concentrations for different cases. Precision of results depend how much model is close to real model. The precision of the proposed extended model may be adjusted through the previous mentioned parameters. The model can be used if "improved Boolean GRN" check box is checked. In that case simulation software work using all new mentioned features. After defining beginning conditions, we can enter number of steps in "Number of steps" field and click "Simulation". "In simulation results" text box will be received protein concentration status for every time step.

For example, we can take generic Genetic regulatory network with 6 nodes from which one is external node. States of vectors let be:
P={2 0 1 0 16 5}
S={4 1 3 8 8 8}
D={1 2 1 0 0 1}
Th={2 3 3 1 3 1}
For first state of network 100011 and state of external node 1 we have dynamic of protein concentration in 30 steps, shown in Figure 5

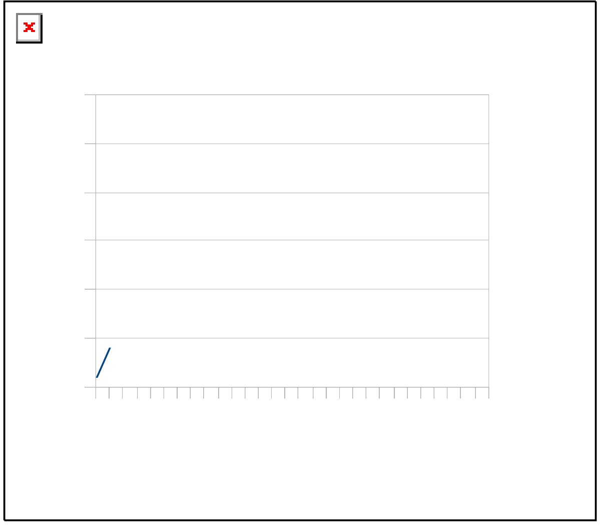


Figure 5. Dynamic of protein concentration

We will present example of simulation of real case, transcription of arabinose operon in E. coli in presence and absence of Arabinose[3],[4].

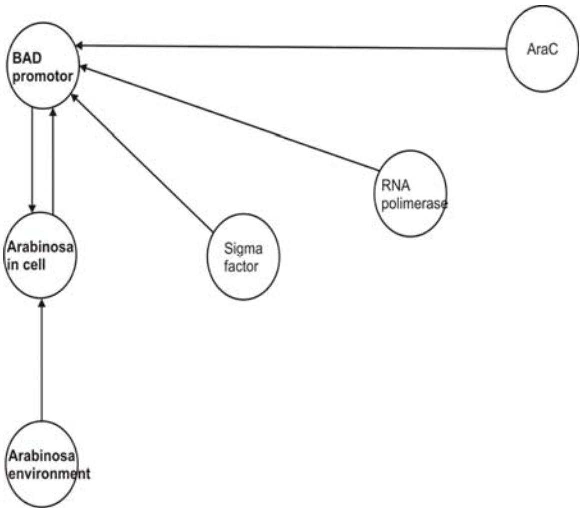


Figure 6. Regulation of BAD operon i E. Coli

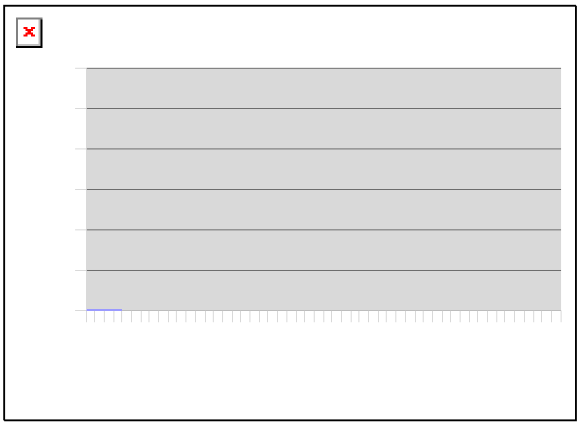


Figure 7. Graphic presentation of concentration of the BAD proteins

Topology of regulatory network is given in Figure 6. If all inputs for "BAD promotor" node are 1, "BAD promotor" in next step will have value 1, otherwise, in other 15 cases "BAD promotor" will have value 0. Node "Arabinosa in cell" will have value 1everythime when "Arabinosa env" node has value 1.
The state of the vector S={8,8,1,1,1}
The state of the vector Th={1,28,1,1,1}
The state of the vector D={1,1,1,1,1}, where 1 addresses function 2*Ln(concentration).

In the simulation it is taken that RNA Polymerase, sigma factor and AraC are always present in sufficient quantities. It was examined the concentration of the BAD proteins depending on whether there is or there is not an Arabinose in the environment of the cell. The concentrations of the observed elements in 53 time steps are given in the diagram shown in Figure 7.

When the arabinose is absent from the environment of the cell, neither there is arabinose in the cell, nor the Arabinose BAD operon is expressed (step 1-5). When the cell is found in environment rich with arabinose, the concentration of arabinose in cell begins to increase (step 6-22). The concentration of BAD proteins begins to increase with a small delay (step 14), which is logic, because a time is necessary for initiation of the transcription and for initiation of the process of translation and for creation of the BAD proteins. If the environment of the cell runs out of arabinose (step 22), the quantity of the same in the cell begins to decrease, and the quantity of BAD proteins begins to decrease with delay (step 25), which is logic, because in the cell there are still active mRNA molecules used for translation and biosynthesis of new BAD proteins. Further, the cycle is repeated once more.

REFERENCES

[1] Stuart A. Kauffman, The Origins of order. University of

[2] Pennsylvania and The Santa Fe Institute, (1993)
B. Derrida, Y. Pomeau. Random Networks of Automata: A Simple Annealed Approximation, (1986)

[3] Lewin B., Genes VII, Oxford University press 2000

[4] Darnel J., Lodish H., Baltimore D., Molecular cell Biology, 1990

[5] Tianhai T., Burrage K., Stochastic neural Network Models for Gene Regulatory Networks, Advanced Computational

[6] B. Janeva, Introduction in theory of sets and math logic, University "Sv. Kiril i Metodij", Skopje, 2001

[7] http://www.biochem.emory.edu/labs/genekdw/protdeg2000/intro.html (10-04-2010 16:20)

Orce Simov receive the B.S. degree in Computer Science, Informatics and Automatics from Elektrotechnical faculty at University St Kiril and Metodij in Skopje, Republic of Macedonia in 1998. He receive M.S. degree in Computer science and genetic engineering, 2001 from Macedonian academy of Science and Art. Now he is PhD student at Contemporary Science faculty of SEE University.

Prof. Dr. Agni Dika holds a Ph.D. degree in field of Computer sciences given by Electro-technical Faculty at University of Zagreb. He is Professor at Faculty of Electro-technical and Computer Engineering at the University of Prishtina. At the same time he is also Professor at Faculty of Contemporary Sciences and Technologies at South Eastern European University in Macedonia.

Prof. Dr. Biljana Percinkova has graduated from Electro-technical faculty in Skopje at the department for Electronic. She has earned her Master's degree at the Electro-techical faculty in Zagreb at the Computer Sciences Department. She has earned her Doctoral degree under mentorship of Prof. Dr. Miodrag Novakovic from Novi Sad University.