Pacific Symposium on Biocomputing 4:102-111 (1999)

NetWork: AN INTERACTIVE INTERFACE TO THE TOOLS FOR ANALYSIS OF GENETIC NETWORK STRUCTURE AND DYNAMICS

M.G. SAMSONOVA, V.N. SEROV
Institute for High Performance Computing and Data Bases,
p.o. box 71, St.Petersburg, 194291 Russia

We designed a Java applet called NetWork which enables a user to interactively construct and visualize a genetic network of interest, and to and to evaluate and explore its dynamics in the framework of a Boolean network model. NetWork displays the mechanism of gene interactions at the level of gene expression and enables the visualization of large genetic networks. NetWork can serve as an interactive interface to tools for the analysis of genetic network structure and behavior.

1 Introduction

Understanding the mechanisms of cellular function requires careful study of the complex behavior of ensembles of interacting genes. Standard biochemical methods are highly biased towards the identification of single genes and their effects. Thus the elucidation of cell function requires new approaches which go beyond the traditional experimental methods.

One of these approaches is based on the application of mathematical and computational methods to build a formal model. Mathematical modeling has made an enormous impact on biology. It has given rise to a great many new theoretical ideas and hypothesis, and moreover stimulated a great deal of experimental work aimed at their verification or disproof. In this way mathematical modeling provides insight into the function of biological systems.

At present genetic networks are a commonly accepted model for the representation of ensembles of interacting genes. Recent progress in molecular genetics has led to the acquisition of a wealth of data, but it is still an open question how to represent real genetic networks mathematically. At present the simplest and the most computationally effective model system that gives some insight into the overall behavior of large genetic networks is the Boolean network[1], in which the state of a gene is denoted by “1” if it is turned on and “0” if it is turned off. In this model genes are represented as the elements of the Boolean net, and the wiring of the elements corresponds to the functional links between genes. The state of each gene is determined by its inputs and a Boolean function (mechanism of genes interaction). For a given set of elements, wiring and rules, a particular trajectory of a network can be calculated. Such a trajectory eventually reaches either a steady state or a repeating cycle.

This final state is called an "attractor". Because an attractor is dynamically stable against perturbations it provides a good mathematical representation of a terminally differentiated cell type at the end of development[2].

A full understanding of the logic and complex behavior of genetic networks will require the development of new theories and algorithms. In particular, tools for the visualization and interactive exploration of complex networks are essential. Genetic networks consist of tens or hundreds of genes involved in complex regulatory interactions. Visualization is critical for comprehending the dynamical behavior of mathematical models with many components and for the comparison of modeling results with the experimental data. Thus it is of prime importance to develop computational tools specifically designed for the visualization of the structure and dynamics of genetic networks in the framework of different models.

We have designed a prototype of a tool of this kind using Java, which is a simple, object-oriented, distributed, interpreted, robust, secure, architecture neutral, portable, high performance, multithreaded, and dynamic programming language[3]. Java has been successfully applied to deliver a wide range of interactive services over the Internet. In particular it is widely used in scientific applications, including the visualization of biological data and dynamical systems[4]. To begin with, we chose the Boolean network model for simulation of network dynamics due to its simplicity and computational efficiency.

An applet called GeneNet has been developed recently which displays genetic networks automatically generated from the data contained in the database[5]. This tool works well in representing information on the interaction of elements comprising the genetic regulatory networks: genes together with their RNA and protein products. However, GeneNet displays a genetic network in a purely static fashion and does not permit the visualization of large genetic networks.

We previously developed the GeneGraph applet for the visualization of genetic networks. GeneGraph is the interface to the GeNet database[6] and can be used as a Web publishing tool by molecular biologists studying the mechanisms of gene interactions[7].

The new applet NetWork, which we describe here, enables a user to construct interactively her own genetic network, to work with a genetic network (as specified by user interactively, or chosen by her from the menu of choices) and to evaluate the network dynamics in the framework of a Boolean network model. NetWork can be used as an interactive interface to the tools for the analysis of genetic network structure and behavior.

2 Materials and Methods

2.1 The algorithm for simulation of network dynamics

We represent a gene network by a set of N binary genes a_i, $i = 1, ..., N$. When $a_i = 0$, the ith gene is considered to be switched off, while when $a_i = 1$, it is considered to be switched on. The network state at time t is specified by the vector $\mathbf{A}(t) = \{a_1(t), ..., a_N(t)\}$. The state at the next time step is described by the matrix equation

$$\mathbf{A}(t+1) = \mathbf{\Theta}\left(\mathbf{A}(t) \cdot \mathbf{M}\right),$$

where the $N \times N$ matrix $\mathbf{M}$ with components m_{ij} is the constant matrix of gene interactions.

A Boolean function in which at least one variable in at least one state can completely determine the output of the function is called a "canalizing" function; Boolean networks constructed from canalizing functions tend to settle to stable attracting states [1]. The Boolean functions used in NetWork are constrained to be canalizing by requiring that the presence of at least one repressor among regulators of the given gene definitively turns it off at the next step regardless of the action of the other genes. Hence elements of the matrix $\mathbf{M}$ may have the values

$$m_{ij} = \begin{cases} 0, & \text{no interaction between } i\text{-th and } j\text{-th genes;} \\ 1, & i\text{-th gene activates } j\text{-th gene;} \\ -N, & i\text{-th gene represses } j\text{-th gene.} \end{cases}$$

Gene interactions are in general not symmetric, so that $m_{ij} \neq m_{ji}$. Each diagonal element m_{ii} of the matrix represents the autoregulatory action of a gene on itself.

As the gene states are binary, the step operator, $\mathbf{\Theta}$, defined as follows, is applied. If $\mathbf{X}$ is a vector with the components $\{x_1, ..., x_N\}$, then

$$\mathbf{\Theta}(\mathbf{X}) = \{\mathrm{sgn}(x_1), ..., \mathrm{sgn}(x_N)\},$$

where

$$\mathrm{sgn}(x) = \begin{cases} 0, & x \leq 0; \\ 1, & x > 0. \end{cases}$$

Thus we get

$$a_i(t+1) = \mathrm{sgn}\left(\sum_{j=1}^{N} a_j(t)\ m_{ij}\right),$$

which reduces to zero in two cases: *a)* the sum is zero, i.e. there is no interaction between the i-th gene and all the genes of the genome, activated at time t, including itself; *b)* the sum is negative, i.e. there is at least one gene, which represses the i-th gene. Otherwise the sum becomes positive and $a_i(t+1) = 1$.

An attractor is a subspace of the network's state space which dynamical trajectories enter but do not leave. An attractor is described by a finite set of network states $\{\mathbf{A}_1, \ldots \mathbf{A}_k\}$, where k is the number of states in the attractor. Once the system reaches the attractor it is described by the equation $\mathbf{A}(t_{i+k}) = \mathbf{A}(t_i)$, $i > I$, where t_I is the time at which attractor is reached. Most attractors in NetWork have $k = 1$ because of the canalizing condition.

2.2 The applet architecture

The applet NetWork **(http://www.csa.ru:81/Inst/gorb_dep/inbios/Dyn_bool/Dyn.htm)** is written in Java. For the presentation of genes and gene interactions, we modified the Nodes and Edges classes developed by Sun Microsystems Inc[3].

A new class DYNAMICS was developed for simulation of network dynamics based on the Boolean algebra technique. To achieve that, the Dynamics class has the following methods:

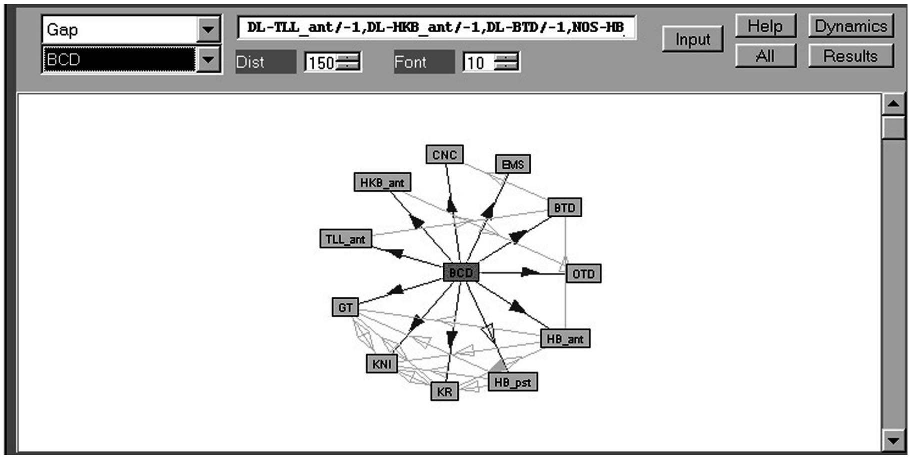
- calculate the initial values of both the vector of network states and the gene interactions matrix,
- calculate the network transition from one state to another,
- save a sequence of network states to a 2D array and perform the comparisons until a state that has occurred previously is encountered,
- output the network trajectory from the array in another applet window.

The applet can be accessed through any World Wide Web browser conforming to the Java standards. The source code is available from the authors upon request.

In the current implementation NetWork operates with objects and propagates events (e.g. mouse clicks) on them. Currently, the object types that can be displayed by this applet include genes and arrows which reflect gene interactions. Data presentation and display functionality are controlled by the developer via parameters (applet tags) in the HTML file containing the applet.

The applet takes a number of parameters following HTML 3.2 and Java applet tag specifications: AllNetNames, NetN and NetName.

The AllNetNames tag **<param name=AllNetNames value="Name1; Name2; Name3;...">** determines the name of genetic network, which can be chosen by the user from the set specified by the developer.



Gap

BCD

DL-TLL_ant/-1,DL-HKB_ant/-1,DL-BTD/-1,NOS-HB

Dist 150

Font 10

Figure 1: NetWork displays the result of gene selection from the GENE CHOICE box. The NETWORK CHOICE box shows the name of the selected genetic network, while the GENE CHOICE box displays the name of the selected gene. NetWork shows the interactions of BCD with its regulators and targets in one browser window.

The NetN parameter (tag) defines the number of genetic networks available in the set.

The structure of each genetic network present in the set is defined by the NetName parameter as follows <**param name=NetName value="gene1-gene2/k, ...,genei-genej/k...."**>. In each pair of genes the first one is the regulator, the second gene is the target, **k** is +1 if the regulator activates the target and −1 otherwise.

The applet architecture gives great flexibility for displaying various genetic networks, as specified by the user interactively or chosen by her from a menu of choices. For the developer it also means that a new genetic network can be added easily and any existing network can be modified or replaced without affecting the functionality of others.

3 Results

Our goal was to provide the user with a convenient interactive interface to the tool for network dynamics simulation, which allows the clear display of both network structure (no matter how large it is) as well as the simulation results.

The applet NetWork subdivides the browser window into two panels: GraphPanel and CtrlPanel. The first one displays the information on the genetic network, while the second contains the control elements, which are the TEXT, GENE, and NETWORK CHOICE boxes; the buttons INPUT, ALL, DYNAMICS, RESULTS, and HELP; as well as the lists DIST and FONT. The size of the GraphPanel is determined automatically depending on the number of network genes. The vertical scrollbar provides easy navigation in the network.

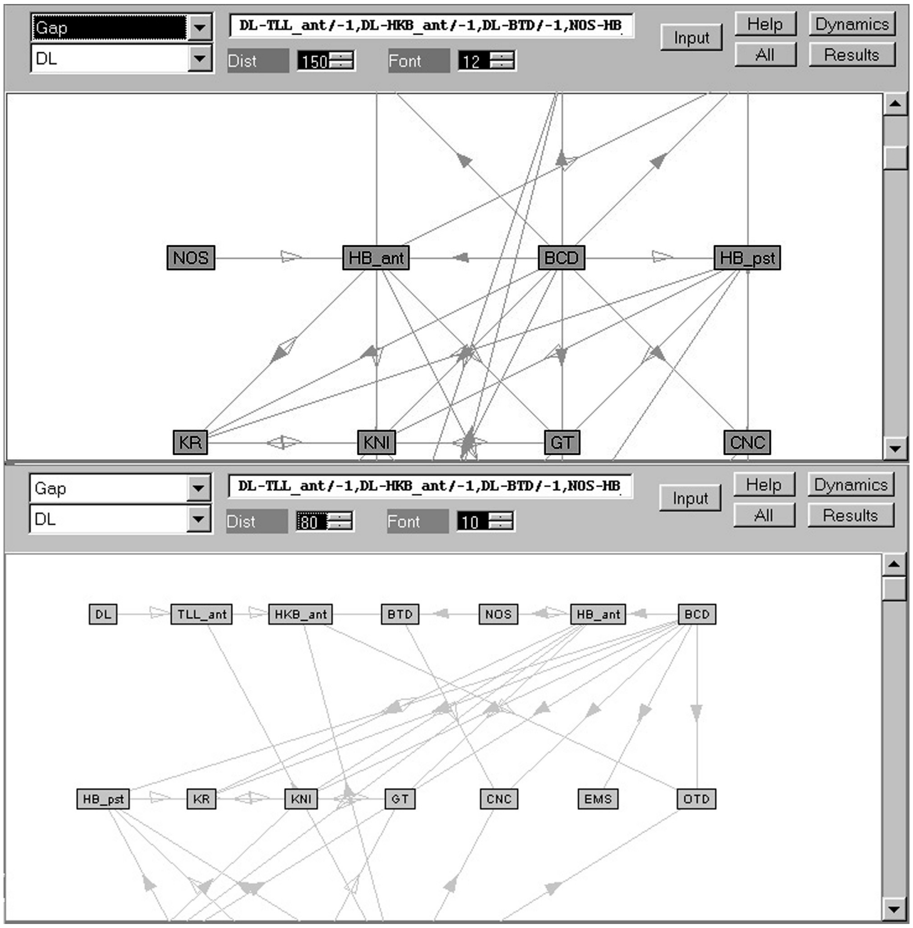
The basic elements displayed by NetWork are genes, represented as rectangles, and their interactions, shown as arrows. A given gene may be selected either by clicking with a mouse or by a name from a popup menu of network genes in the GENE CHOICE box. Selection of a gene displays its interactions with regulators and targets in one browser window, while the other network gene images disappear (Figure 1). If the gene was selected by clicking with a mouse these images come into view after selection while the interactions of a gene with its regulators and targets remain highlighted. Red arrows connect a gene with the upstream genes of a network, and blue arrows with the downstream genes. Filled and hollow arrows reflect a mode of gene action—activation and repression correspondingly.

The use of the GENE CHOICE box to select a gene, as well as the representation of gene interactions in one browser window enhances the visualization of gene interactions in large genetic networks. It relieves the user from the tedious necessity of carefully inspecting a complex diagram in order to find a gene or its regulators and targets.

A user is provided with the opportunity to vary a size of the network region displayed in the browser window. To do it a user can choose the sizes of rectangles representing the genes, as well as the distance between genes by selecting the required values from the FONT and DIST lists. As the result a user can see in the browser window the whole network or only a part of it (Figure 2).

A gene can be dragged to a new place, that improves the visualization of links between genes in case of large network.

The simulator can make use of stored network circuits chosen from a menu or the user can construct them interactively. To do this, the user first enters in the TEXT box the information on genes and mechanisms of their interactions in the following format **gene1-gene2/k, gene3-gene2/k,...**, where **k** equals to +1 in case of activation and −1 in case of repression; **gene1, gene2** and **gene3** are the genes' names. In each pair of interacting genes the first one is the regulator, while the second gene is the target. After completion of the input the user touches the INPUT button that enables the visualization of input data as a genetic network.



Gap
DL
DL-TLL_ant/-1,DL-HKB_ant/-1,DL-BTD/-1,NOS-HB
Dist 150
Font 12

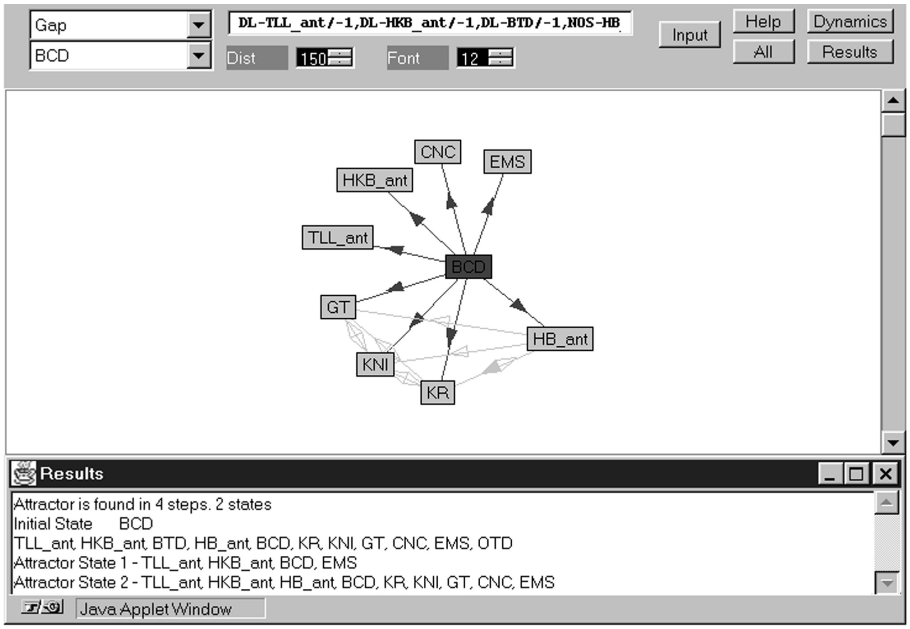
Gap
DL
DL-TLL_ant/-1,DL-HKB_ant/-1,DL-BTD/-1,NOS-HB
Dist 80
Font 10

Figure 2: NetWork displays the genetic network in two different resolution scales. The resolution values are given in the FONT and DIST lists. The NETWORK CHOICE box displays a name of the preselected genetic network.

To work with a genetic network predefined by the developer the selection of a network from the NETWORK CHOICE box, which displays a popup menu of genetic networks available, is required. Pressing the INPUT button brings to a user the image of the selected network.

While working, a user can perform interactive editing of the genetic network by deleting or adding genes or links in the TEXT box. Pressing the

INPUT button shows the modified network in a browser window. This procedure enables a user to model the effect of mutations. Interactive editing can be performed on both user-defined and menu-selected networks.



Gap
BCD
DL-TLL_ant/-1,DL-HKB_ant/-1,DL-BTD/-1,NOS-HB
Dist 150
Font 12

```
Attractor is found in 4 steps. 2 states
Initial State      BCD
TLL_ant, HKB_ant, BTD, HB_ant, BCD, KR, KNI, GT, CNC, EMS, OTD
Attractor State 1 - TLL_ant, HKB_ant, BCD, EMS
Attractor State 2 - TLL_ant, HKB_ant, HB_ant, BCD, KR, KNI, GT, CNC, EMS
```

Figure 3: NetWork displays the result of the genetic network dynamics. On the GraphPanel one of the attractors is displayed. The Results window displays the trajectory of network's dynamics.

To perform a simulation of genetic network dynamics a user selects the genes initially switched on in the network by clicking on them while pressing the Shift key. To facilitate the selection of a gene in the case of large genetic networks a user can employ the GENE CHOICE box. In this situation the initially switched on genes can be selected by clicking on the gene name while pressing the Shift key. Pressing the DYNAMICS button after the selection triggers the network transition from one state to another until an attractor is reached. When more than one state makes up the attractor cycle, all these states can be seen by sequential clicks on the RESULTS button.

The user can hide all genes except those which are switched on in the current attractor state by clicking the RESULTS button. Clicking with the mouse on the ALL button restores the display of all network genes.

In addition the RESULTS button gives a user the possibility to see the results of the network dynamics in another applet window (Figure 3).

It is possible to run the simulator repeatedly so that the effects of changes in initial conditions or modifications of the network can be explored.

4 Applications

The main application of NetWork is to serve as an interactive interface to the tools for the analysis of genetic network structure and behavior via the Worldwide Web. It can also function as a standalone application. At present NetWork simulates the dynamics of genetic network modeled as a Boolean network. However after some modification NetWork can be used to visualize the dynamical behavior of other mathematical models of genetic networks.

In addition NetWork can be applied as a visualizer of gene interactions in large genetic networks.

5 Future developments

For further improvements of presentation of large genetic networks we are working now to determine the optimal layout of a network structure by means of a genetic algorithm.

NetWork represents gene interactions at the level of gene expression. This approach seems to be suitable for modeling the genetic network as the Boolean one. The study of the dynamical behavior of other mathematical models of genetic networks may require also a consideration of regulatory interactions occurring at other molecular levels. Therefore we plan to modify our model of presentation of genetic network structure.

The interactivity of Network and its capability of allowing the user to change the genetic network structure online provides a means of using this tool for assessing the validity of current methods for modeling real genetic networks behavior.

6 Conclusions

The NetWork applet considered here provides an example of an interactive front end for the displaying the behavior of mathematical models of genetic networks. It enables the visualization of large genetic networks structure at the level of gene expression. NetWork enables a user to construct the genetic network by interactive input of the information about gene interactions following very simple rules. NetWork presents the possibility to work both with a user defined and with a data provider specified genetic networks by means of modeling the effect of mutations in the network and evaluating the network dynamics in framework of Boolean network model.

Acknowledgments

We are pleased to thank Drs E. Myasnikova, V. Alenin, A. Spirov, J. Reinitz and Ms O. Kirillova for helpful discussions.

The support provided by the Russian Ministry of Science in the framework of the project "SegNet" of the Program for Federal information center for science and technology is gratefully acknowledged.

References

1. Kauffman,S.A. (1993) The Origin of Order, Self-Organization and Selection in Evolution. Oxford University Press, New York.
2. Somogyi,R. and Sniegoski,C. (1996) Modeling the complexity of genetic networks: understanding multigenic and pleiotropic regulation. Complexity, 1, 45-63.
3. Gosling G. and McGilton H. (1995) The Java language environment. Sun Microsystems White Paper, http://www.javasoft.com/whitePaper/ java-whitepaper-1.html.
4. Stein,L. (1996) Web applets: Java, Javascript and ActiveX. Trends Genet. 12, 484-485.
5. Kolpakov,F.A., Ananko,E.A., Kolesov,G.B., Kolchanov,N.A. (1998) GeneNet: Computing system for automated gene network visualization, http://wwwmgs.bionet.nsc.ru/systems/Mgl/GeneNet/outline.html.
6. Serov,V.N., Kirillova,O.V., Savostyanova,K.G., Surkova,S.Yu., Spirov,A.V., Reinitz,J. and Samsonova,M.G. (1998) GeNet, the database of genetic networks. Bioinformatics, (submitted).
7. Serov,V.N., Spirov,A.V. and Samsonova,M.G. (1998) Graphical interface to the genetic network database GeNet. Bioinformatics, 14, 6.