# AUTOMATED MODELING OF GENETIC CONTROL IN *Arabidopsis thaliana*

Katarzyna Bożek, Anna Gambin,
Bartek Wilczyński and Jerzy Tiuryn

Institute of Informatics, Warsaw University, Banacha 2, 03-916 Warsaw, POLAND
e-mail: aniag@minuw.edu.pl

(Received September 15, 2005/Accepted October 30, 2005)

## ABSTRACT

Challenges of contemporary molecular biology include predicting how genes are regulated in a network, which proteins participate in metabolic pathways and how they interact. The high-throughput biotechnologies like microarrays provide us with gigabytes of data describing the expression profiles of genes. There is a need of bioinformatics tools enabling the analysis of these data. In this paper we describe an automatic method of predicting gene regulation functions. In our approach we apply the procedure from (Shamir et al., 2004), appropriately modified to handle real-life examples of genetic control in *Arabidopsis thaliana*.

**Key words:** gene regulation, gene network, graph models, microarrays, boolean functions

## INTRODUCTION

The great challenge of postgenomic biology is to understand how cellular phenomena arise from the connectivity of genes and proteins. This connectivity generates molecular network diagrams. A systematic approach to analyze them requires the development of mathematical models. Ideally, to benefit from the increasing amount of data, this kind of model should be inferred automatically using computational methods.

During the last ten years, a lot of research has been done on mathematical models of gene regulation (De Jong, 2002). There are several approaches; one of them is to describe the behavior of the system using differential equations. Unfortunately, this approach is often impossible to implement. Another approach, that allows to express some level of non-determinism in a real cell, are stochastic differential equations (Chen et al., 2005). A stochastic framework

is also applied in Bayesian networks, which turn out to be a very effective tool for inferring gene interactions from microarray data (Dojer et al., 2005).

In this paper, we describe gene interactions as a dynamic system in which the regulation of a gene is modeled using a logical function associated with it. This function yields the state of a given gene in a following time step, depending on the states of its regulators in the preceding step. The most important feature of the model is its ability to make the predictions about the behavior of the system. The process of inferring the model has several stages. We focus here on the problem of modeling the dynamics of the gene network. Throughout the paper, we assume that the topology of interactions is known from previous experiments. The behavior of the dynamical system we deal with can be characterized by the set of steady states, attractors. These are the endpoints of possible trajectories of the system's dynamical behavior. It is assumed that the outcomes of the microarray experiments correspond exactly to the steady states of the observed system. Hence, we will consider the given set of interactions and the given set of stationary states and we will find an appropriate set of regulation functions.

The same problem was considered by (Mendoza and Alvarez-Buylla, 2000; Mendoza et al., 1999). However, their approach is restricted to a concrete regulation system and cannot be easily generalized. In contrast to this kind of manual approach, we propose a fully automatic procedure for inferring the dynamical behavior of the system with a given set of attractors. To this aim, we adopt the algorithm proposed recently by Shamir et al. (2004). Since this method fails when applied to some more complex networks, we proposed necessary modifications yielding a significant improvement with respect to the original version.

Following (Mendoza et al., 1999), we will consider the model plant Arabidopsis thaliana. There is a large body of published data that support the existence of complex regulatory networks of two molecular processes in *A. thaliana*: flower morphogenesis and root hair development.

In the last ten years, A. thaliana has become universally recognized as a model plant for molecular studies. It is a small flowering plant that belongs to the Brassica family, which includes species such as broccoli, cauliflower, cabbage, and radishes. Although it has no commercial uses, it is favored among scientists because it develops, reproduces, and responds to stress and disease in much the same way as many crop plants.

In this paper two examples of gene networks corresponding to different developmental processes are considered. The first network includes genes involved in flower morphogenesis that has been intensively studied and forms the basis of the 'ABC' combinatorial model (Coen and Meyerowitz, 1991). Roughly speaking, there are three different genetic activities, each of them present in two adjacent whorls and each whorl requires a specific combination of genetic activities. For example, activities A and B combined in the second whorl determine petal identity, A alone determines sepal identity, B and C determine stamen identity, and finally C determines carpel identity. The

second example consists of modeling the root hair development process. We consider the network of interacting genes which control root epidermal differentiation.

## MATERIAL AND METHODS

The model that we analyze has been proposed by Irit Gat-Viks, Amos Tanay and Ron Shamir (2004). The approach consists of building an initial model based on biological knowledge and refining it in order to increase the adequacy between model predictions and biological data. The model encompasses heterogeneous biological entities (mRNAs, proteins, metabolites) and a wide variety of regulation mechanisms.

The regulatory network is represented by a directed graph (Fig. 1). Graph vertices correspond to model variables, and edges to direct variable dependencies, which means that the edge $u \to v$ expresses the fact that $u$ is a regulator of $v$. Additionally, each variable with non-zero in-degree has its regulation function that determines the state of the variable depending on its regulators. A model state is an assignment of states to all model variables. We say that a variable state agrees with the model when its value is induced by the corresponding regulation function applied to the regulators. A model state that agrees with each variable is called a mode.

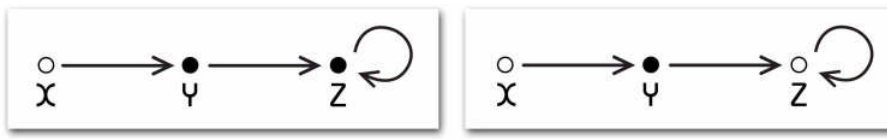


**Figure 1.** Example of a regulation graph

Figure 1 illustrates the above notions. The model includes three variables with binary state space (dark – off, light – on). The regulation functions are:

$$f_v(x) = not\, s(x) \quad \text{and} \quad f_Z(x,z) = s(x)\, or\, s(z)$$

We call the vertex having in-degree zero the source of the graph. In our example the only source is vertex X. We are interesting in the modes of our model – clearly they depend on the state of the source X. For example, if the vertex X is "on" the model has two modes: [X,Y,Z]=[1,0,1] and [X,Y,Z]=[1,0,0].

The algorithm from (Shamir et al., 2004) consists of two main phases: computing the modes and learning regulation functions. Computing modes plays a role of regulation simulation: given an experimental starting condition, it predicts the final mode at which the system eventually arrives. Learning regulation functions aims at improving consistency of the model with the experimental data.

In order to measure the ability of the model to correctly predict the outcome of biological experiments the authors of (Shamir et al., 2004) introduced a discrepancy score. This score measures the discrepancy between experimental measurements e and the mode s resulting from the algorithm for the same experimental conditions. The discrepancy function is defined as follows: where we sum over all vertices in the graph model M. $s(v)$ denotes the state of vertex v in the mode $s$ and $e(v)$ denotes the state assigned to $v$ by the experimental measurements.

$$D(s,e) = \sum_{v \in M} (s(v) - e(v))^2$$

The problem of learning regulation functions reduces therefore to optimizing one particular function in the model. This way, it is possible to derive an improved model with a lower discrepancy. The function optimization problem is computationally hard. The solution proposed in (Shamir et al., 2004) is an approximation obtained by translating the problem to a combinatorial problem on matrices. The outline of the algorithm of (Shamir et al., 2004) is as follows.

**Algorithm 1**

1: construct a graph of gene interactions – the topology should be based on biological knowledge;
2: determine the set of modes of the graph (i.e. simulate the model for every assignment of states to the sources of the graph);
3: compare the simulation results with biological data, i.e. compute the discrepancy function;
4: **while** the results are not satisfactory **do**
5: learn regulation functions,
6: **return** to step 2.
7: **end while**

The first modification we introduced in the algorithm consisted of several different strategies of choosing the regulation function to be optimized in Step 5. This aspect of the algorithm is not described in (Shamir et al., 2004). Since the learning algorithm stops in a local minimum, we suppose that the order of optimization might have an impact upon the result. We have developed strategies that are slightly more efficient than the naive strategy of learning all functions in a random order. From the outcomes of several experiments for various strategies, we concluded that the overall improvement obtained in this way was not significant.

The second modification was to manipulate on the topology of the graph. We have explored the observation that the graph topology is assumed to be known at the start and remains unchanged throughout the algorithm. We have intensively tested the algorithm on different graphs. It turned out that the algorithm is much more efficient on small graphs. This means that the learning procedure proposes the regulation functions yielding smaller discrepancy.

Therefore, we have decided to introduce a modification according to the divide and conquer principle. Our idea was to divide the graph, perform the basic algorithm on its subparts, and then merge the results in the best way.

In our implementation, after the construction phase the algorithm performs user-predefined cuts on the graph and applies Algorithm 1 to the resulting subgraphs. Regulation functions obtained for the subgraphs are merged to form functions in the original graph. This is done by extending each function to the function with a greater number of arguments (corresponding to the edges removed by the cut). Algorithm 1 is then iterated several times on the merged graph.

## RESULTS AND DISCUSSION

The regulation systems we analyzed are flower morphogenesis and root epidermis cells hair development of A. thaliana. Our modification has significantly improved the algorithm performance for these two regulation networks.

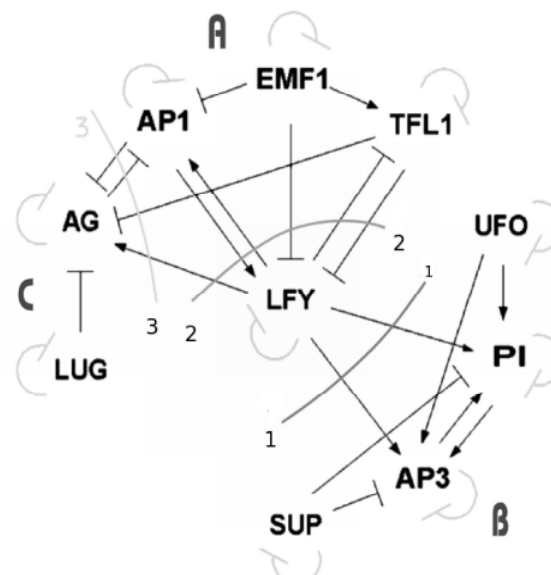


**Figure 2.** Regulation network of *A. thaliana* flower morphogenesis from (De la Fuente et al., 2002)

The first regulation network (Fig. 2) is composed of ten variables. Following (De la Fuente et al., 2002) we were using ten experimental data measurements for this model.

Algorithm 1 has been run ten times on the entire graph yielding an average discrepancy score of 17.2. This is a rather poor result because it means that on average in each simulation there is at least one incorrect regulation function.

Regarding the graph topology we proposed 3 cuts (see Fig. 2):

Cut 1 is the most intuitive for the graph topology – it divides the graph by removing a minimal number of edges of the same direction.

Cut 2 was induced from the results obtained for Cut 1 – the only discrepancies that appeared after Cut 1 came from the subgraph A. The idea therefore was to divide the graph into two equal parts.

Cut 3 is an improvement of the first cut – since dividing the graph into two equal parts did not give satisfying results we decided to keep Cut 1 and additionally divide the subgraph A.

Table 1. Results of cutting algorithm on the flower morphogenesis regulation network

| | Cut 1 | Cut 2 | Cut 3 |
|---|---|---|---|
| Subgraph A | 3.5 | 0.0 | 2.1 |
| Subgraph B | 0.1 | 0.9 | 0.1 |
| Subgraph C | - | - | 0.0 |
| **Merged graph** | **6.9** | **19.1** | **5.1** |
| Min. | 1 | 4 | 1 |
| Max. | 14 | 31 | 9 |
| Graph without cuts | 17.2 | 17.2 | 17.2 |

Table 1 shows the results of each cut.

Already the first cut resulted in an over 50% improvement. Subgraph B is sufficiently small for the algorithm to find the correct regulation functions. Nevertheless, the bigger subgraph with the average discrepancy of 3.5 contributes to the final discrepancy in the merged graph.

The second cut divides graph into two parts of roughly equal size. Even though the correct regulation functions in each part are found, the results after merging are not satisfactory. This observation suggests that the optimal cuts should be minimal. If we remove many edges the partial results are not relevant for merged graph.

Taking into consideration the results of Cut 2 and the graph topology, we concluded that the only way to obtain even better results is to keep Cut 1 and divide the bigger subgraph. This resulted in the most accurate model with an average discrepancy of 5.1. Each of the created subgraphs had small discrepancy and the graph merge did not generate additional discrepancies. The second model we used was the regulation system of root epidermis hair

development in A. thaliana (Mendoza and Alvarez-Buylla, 2000,). This is a graph of eight vertices with a small number of edges. There are two regulation pathways ending in genes which are crucial for the hair development.

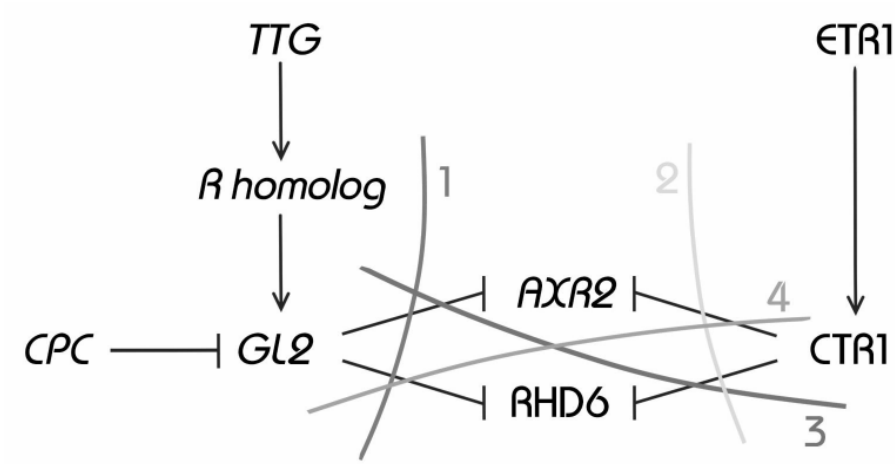


**Figure 3.** Regulation network of root epidermis hair development of *A. thaliana* (Mendoza and Alvarez-Buylla, 2000)

The size and density of the graph (Fig. 3) does not present many cutting possibilities. We proposed four cuts:

Cut 1 seams to be reasonable since it divides the graph into two roughly equal parts and the removed edges have the same direction.

Cut 2 removes edges of the same direction but doesn't divide the graph equally.

Cuts 3 and 4 are symmetrical and remove edges of opposite directions.

Table 2 shows the discrepancy scores of the described cuts.

Table 2. Results of the cutting algorithm on the root epidermis of *Arabidopsis thaliana* regulation network

| | Cut 1 | Cut 2 | Cut 3 | Cut 4 |
|---|---|---|---|---|
| Subgraph A | 0.0 | 5.0 | 2.2 | 3.4 |
| Subgraph B | 6.0 | 0.0 | 2.0 | 4.0 |
| **Merged graph** | **6.0** | **7.0** | **5.4** | **8.6** |
| Min. | 6 | 7 | 5 | 8 |
| Max. | 6 | 7 | 8 | 11 |
| Graph without cuts | 10.3 | 10.3 | 10.3 | 10.3 |

In the case of Cut 3, our modification resulted in a nearly 50% (in the case

of Cut 3) improvement in the algorithm performance. What is specific about cuts on such a network is that, due to a small search space, the results obtained for the subgraphs are highly repetitive. In most cases, the algorithm gives the same result and the learning procedure fails to improve it further.

## CONCLUSION

In this paper we have presented a method for automatically inferring gene regulation functions modeled as logical functions. The method is mainly based on the algorithm proposed in (Shamir et al., 2004) in which our contribution includes the decomposition strategy for the network. This modification substantially improves the applicability of the original method which was proved by the tests on two real-life examples of genetic control in developmental processes of *Arabidopsis thaliana.*

## REFERENCES

Chen Kuang-Chi, Tse-Yi Wang, Huei-HunTseng, Chi-Ying F. Huang, Cheng-Yan Kao 2005. A stochastic differential equation model for quantifying transcriptional regulatory network in Saccharomyces cerevisiae. BIOINFORMA-TICS 21: 2883-2890.

Coen E.S., Meyerowitz E.M. 1991.The war of the whorls: genetic interactions controlling flower development. NATURE 353: 31-37.

De la Fuente A., Brazhnik P., Mendes P. 2002. Linking the Genes: Inferring Quantitative Gene Networks from Microarray Data, TRENDS IN GENETICS 18. 395-398, available at: http://staff.vbi.vt.edu/mendes/Tig02/tig02.htm.

De Jong H. 2002. Modeling and Simulation of Genetic Regulatory Systems: A Literature Review. J. COMPUT. BIOL. 9: 67-103.

Dojer N., Gambin A., Wilczyński B., Tiuryn J. 2005. Applying Dynamic Bayesian Networks to Perturbed Gene Expression Data. TECHNICAL REPORT, Warsaw University.

Mendoza L., Thieffry D., Alvarez-Buylla E.R. 1999. Genetic control of flower morphogenesis in Arabidopsis thaliana: a logical analysis. BIOINFORMATICS 15: 593-606.

Mendoza L., Alvarez-Buylla E.R. 2000. Genetic Regulation of Root Hair Development in Arabidopsis thaliana: A Network Model. J. THEORETICAL BIOL. 204:311-326.

Shamir R., Irit Gat-Viks, Amos Tanay. 2004. Modeling and Analysis of Heterogeneous Regulation in Biological Networks. J. COMPUT. BIOL. 11: 1934-1049.

# AUTOMATYCZNE MODELOWANIE GENETYCZNEJ KONTROLI U RZODKIEWNIKA (*Arabidopsis thaliana*)

Katarzyna Bożek, Anna Gambin,
Bartek Wilczyński i Jerzy Tiuryn

S T R E S Z C Z E N I E

Praca prezentuje algorytm modelowania regulacji biologicznej. Jest to automatyczna metoda konstrukcji i optymalizacji modelu heterogenicznej sieci regulacji na postawie danych z eksperymentów biologicznych. Eksperymenty wykonywane są najczęściej z użyciem technologii mikromacierzy i badany jest w nich poziom ekspresji genów. Przedstawiony przez nas algorytm jest modyfikacją metody Shamira. W zaproponowanym podejściu graf obrazujący zależności pomiędzy genami jest dekomponowany na mniejsze podsieci, których optymalizacja przebiega niezależnie. W końcowej fazie wyniki dla części składowych są scalane, aby uzyskać kompletny model.

Wprowadzone ulepszenie znacznie poprawiło efektywność metody i umożliwiło przetestowanie algorytmu na danych dotyczących regulacji genów w rzodkiewniku (*Arabidopsis thaliana*). Udało się zbudować model dla dwóch procesów różnicowania się komórek – morfogenezy kwiatu oraz rozwoju włosków komórek korzenia.

**Słowa kluczowe:** sieć regulacji genów, model grafowy, funkcje logiczne, mikromacierze