

# Modeling and Determination of the Regulation of Gene Expression: the Binary Switch Model

David Venet  
IRIDIA and IRIBHN  
808, Route de Lennik, CP 609  
B-1080 Brussels, Belgium  
davenet@ulb.ac.be

Carine Maenhaut  
IRIBHN, Campus Hopital Erasme  
808, Route de Lennik, CP 609  
B-1080 Brussels, Belgium  
cmaenhau@ulb.ac.be

Hugues Bersini  
IRIDIA, Université Libre de Bruxelles  
50, Av. F. Roosevelt, CP 194/6  
B-1050 Brussels, Belgium  
bersini@ulb.ac.be

## ABSTRACT

The levels of expression of the genes are controlled by certain genes and by other factors. There exist many different models for these gene regulations. Most of these models only consider the case where the regulators are measured. We propose here a modification of the Boolean network model of gene regulation which permits to describe the case where the regulators are not measured. In our model, the state of the genes is not determined by values found inside the network, but by external switches. The evolution of the profiles of expression can be reduced to those switches, which are expected to represent a biological reality. A technique permitting to infer the values of the switches from the data is presented. This technique is applied to two real sets of data. The switches recovered offer a simple explanation of the behavior of the cells and permit to identify a large part of the regulatory network.

## 1. INTRODUCTION

New technologies allowing the quantification of the level of expression of thousands of genes simultaneously have appeared recently. Their availability has raised the hope that a complete regulatory network in a cell type could be inferred in a systematic and comprehensive way. Different models of such networks exist. The most common ones are Boolean networks [1,8], qualitative model [14], Bayesian networks [12], weight matrices [4,7,15], systems of linear or non-linear differential equations [3,16] and hybrid models [2,10]. This work focuses on Boolean networks.

In the Boolean network model, each gene has only two possible states, “on” and “off”. The state of every gene is a Boolean function of the states of some other genes. Usually, the number of genes necessary to determine the state of a gene is limited in order to avoid overly complex solutions.

This model, like most other models of gene regulation, has an important limitation. The variables which control the expression of the genes are supposed to be gene expressions themselves. In the real world, this is often not the case. Gene expressions can also be controlled by the concentration and the activity of certain proteins, or certain properties of the cell environment (glucose concentration, temperature...). Those parameters are hard to introduce in the framework of a Boolean network in a clean and consistent manner. They can sometimes be estimated using *a priori* knowledge of the experimental conditions, but this is not always possible. For instance, in a study relative to the temporal evolution of gene expression during the cell cycle, the experimental conditions should be labeled in function of the

average state of the cells. Such labeling is hard to do, and prone to error and bias.

We propose here a modification of the Boolean network paradigm which addresses this issue while still keeping its inherent simplicity. This model, the “binary switch model”, states that the genes are controlled by a few “switches”, which are not necessarily among the measured values. Those switches might for instance represent the presence of a certain protein, its activity, or high temperature. This model is a generalization of the Boolean network model. When that model is applicable, the switches can be assimilated to the expression of certain genes. This generalization allows the treatment of a much wider panel of experiments in a systematic fashion.

In order to apply the binary switch model to real world data, a binarization must be performed. The same applies of course to other models which use discrete values, like Boolean and Bayesian networks. Such discretization is usually done by another, independent, algorithm [12]. We show here that the binarization can be performed at the same time than the determination of the regulations. This approach leads to a better definition of the problem, and potentially to better solutions.

We firstly present the binary switch model in more detail and discuss some of its implications. Secondly, we show a technique allowing to infer the parameters of the model from the data. Thirdly, we demonstrate this technique on simulated data. Finally, the models are determined for two real data sets, showing the applicability of the method.

## 2. THE BINARY SWITCH MODEL

The binary switch model describes the regulation of the genes in a simple and understandable way. We present here the hypotheses behind this model in some detail.

Gene expression data can be organized as a matrix,  $G$ . A value  $g_{ij}$  of this matrix corresponds to the level of expression of the gene  $i$  in the experimental condition  $j$ . A first hypothesis is that the matrix  $G$  can be reduced to a binary matrix,  $B$ . This means that there is a threshold  $t_i$  for each gene that can be used to binarize the matrix  $G$ :

$$b_{ij} = g_{ij} > t_i \quad (1)$$

**Table 1. Example of genes respecting the binary switch model. "Val" are the real values measured. "Bin" are the binarized values: "1" if the real value is over the gene's threshold ("Thresh") and "0" otherwise. Gene 1 is Switch 1. Gene 2 is NOT(Switch 1 AND Switch 2). Gene 3 is XOR(Switch 1, Switch 2). Gene 4 is NOT(Switch 2).**

	Exp 1		Exp 2		Exp 3		Exp 4		Exp 5		Exp 6		Exp 7		Exp 8		
Switch 1	1		1		1		0		0		0		1		0		
Switch 2	1		0		1		0		1		0		1		0		
	Val	Bin	Val	Bin	Val	Bin	Val	Bin	Val	Bin	Val	Bin	Val	Bin	Val	Bin	Thresh.
Gene 1	4.1	1	3.5	1	5.1	1	2.0	0	1.5	0	.23	0	2.7	1	1.1	0	2.35
Gene 2	.5	0	1.3	1	.21	0	1.2	1	.9	1	2.1	1	.17	0	1.5	1	0.7
Gene 3	.5	0	1.2	1	.8	0	.4	0	1.7	1	.7	0	.8	0	.1	0	1
Gene 4	1	0	1.3	1	.5	0	1.4	1	.3	0	8.2	1	.1	0	1.5	1	1.2

where  $b_{ij}$  is the binary expression of gene  $i$  in condition  $j$ . Its value is "1" if  $g_{ij}$ , the real valued expression of gene  $i$  in condition  $j$ , is higher than a gene specific threshold  $t_i$  and "0" otherwise.

A second hypothesis is the existence of a certain number  $N$  of binary switches. For a condition  $j$ , there is a value  $sw_{jk}$  of the switch  $k$  associated. Those switches describe the regulatory state of the cells. They could be a function of the presence or absence of a given protein, or of its activity, or of high temperature, or of anything else.

Finally, the model implies that the binarized values of gene expression are Boolean functions of the switches:

$$b_{ij} = f_i(sw_{j1}, sw_{j2}, \dots, sw_{jN}) \quad (2)$$

where  $f_i$  is the function linking the value of the gene  $i$  to the switches.

This model is a generalization of the Boolean network model. In the network model, it is supposed that the states of the genes are a function of the state of other genes. When the Boolean network model is applicable, the switches can be assimilated to the expression of certain genes. The binary switch model generalizes the Boolean network model to the case where the variables responsible for the evolution of the profile of gene expression are not measured. Since this is often the case, the binary switch model is more realistically applicable than the Boolean network model.

The inference of the switches from the data permits the determination of a large part of the regulatory network. In order to determine the rest of this network, the switches must be tentatively identified with an underlying biological reality. If this biological reality is not among the quantities measured, *i.e.* if it is not a gene expression, then no identification is possible.

In the special case of a kinetic study (temporal evolution), if a regulation is performed via a gene regulated at the mRNA level, then it should be possible to identify a switch with this gene. In that case, the switch represents the activity of the transcription factor while the gene expression measured is the corresponding mRNA level. The evolution of the activity of a gene should be similar to the evolution of its mRNA level, with a certain delay caused by the time taken for the translation, the folding and sometimes the activation of the corresponding protein. Since the activity of a transcription factor should be represented as a switch,

such switch should be correlated with the level of expression of the gene, with a certain delay. Hence, the gene could be identified using its delayed correlation with a switch.

Methods similar to this one have been proposed to identify Boolean networks. In these approaches, the complexity of the identification is much higher, since many possible links are taken into consideration. Also, the delays between the cause and the effect are usually arbitrarily set to the time between the measurements taken, which limits the applicability of the techniques. And of course, those methods are not resilient at all to missing values.

The model presented has certain implications concerning the maximum possible number of different experimental conditions and gene profiles. For a given number  $N$  of switches, there are at most  $2^N$  different experimental conditions. The same limitation applies to Boolean network models as a function of the number of transcription factors. This maximum number of conditions could be considered as too high or too low, depending on the point of view.

In the framework presented, the experimental conditions which have the same combination of switches must be grouped for the determination of the switches. This is done by clustering the conditions in at most  $2^N$  groups. In order to render this clustering meaningful, the number of experimental conditions should be much higher than the number of groups, hence usually higher than  $2^N$ . This means that the number of experiments needed to identify the parameters of the model grows exponentially with the number of switches, and so must be quite large for even a moderate number of switches.

However, taking a different point of view, the maximum number of conditions can be considered as low. Each experiment being done in a different setup, the results are different. If the switches are supposed to explain the cell's entire behavior, their number should be such that  $2^N$  is higher than the number of experiments. But then, no identification is possible using the framework presented.

In this work, we consider that the number of switches is such that  $2^N$  is much smaller than the number of experimental conditions. The hypothesis is that there exists a sufficient amount of similarity between the conditions with the same values of the switches for

those switches to predict the behavior of many genes. We do not consider that we are in a situation where everything can be deduced from the measurements, but that certain switches nevertheless exist, which explain a large part of the cell's behavior.

The model does not constrain at all the functions linking the switches and the genes. For a given number of switches  $N$ , the number of possible different gene functions is  $2^{2^N}$ . This number raises very fast with  $N$ . Four switches are enough to allow every gene in the human genome to have a different behavior. Since the expressions of many genes are correlated, this level of freedom seems too high. The gene functions should probably be constrained somehow. Such constraints would also allow the determination of  $N$  switches with less than  $2^N$  experimental conditions.

The binary switch model can also be viewed as a high-level description of the state of the cells. The switches often represent understandable experimental conditions, like temperature or starvation. Our technique may then offer an explanation of a large part of the gene expression measurements in terms of simple concepts. Such simplicity makes it a useful tool for biological understanding.

### 3. IDENTIFICATION OF THE MODEL

The binary switch model implies that all experimental conditions sharing the same combination of switches have the same binarized profiles of expression. This means that the conditions can be grouped in function of their switches. However, there is an indetermination in the values of the switches. Once the conditions are grouped, any solution for which each group has a different combination of switches respects as well the model. The switches are not determined by the data, only the grouping of the conditions is. This is due to the lack of constraint put on the form of the Boolean functions linking the switches and the genes. A criterion will be defined later in order to choose the "right" switches among all possible combinations.

The number of groups of switches is not determined by the model. The only limitation is that this number is at most  $2^N$ , where  $N$  is the number of switches. The number of groups is expected to be lower than this maximum, because usually not every combination of switches is experimentally available. If for instance one switch is relative to excess heat and another to excess cold, it is unlikely that there exists an experimental condition where both of these conditions are simultaneously "on". As the number of switches increases, the number of missing combinations increases as well. The choice of the optimal number of groups and switches has to be done by judging the solutions obtained and by using biological insights concerning the data.

The determination of the parameters of the model is divided in two parts: firstly, the thresholds and the groups are determined, such that a maximum of genes has a constant value inside each group. Secondly, the best values of the switches for each group are determined.

## 3.1 The Thresholds and the Groups

### 3.1.1 Function to Maximize

We show here how the grouping of the samples and the binarization of the data are done. A quality function is defined whose maximum should correspond to the best possible groups and thresholds. Those are then determined by maximizing the function.

Trivial solutions for which every gene respects the model always exist. For instance, if the thresholds are sufficiently low all binary values become "1" and any grouping gives a solution which perfectly fits the model. The quality function should be very low for such trivial solutions. A more interesting solution should be more informative concerning the regulation of the cells.

It is not expected anyway that every gene fits the model. Many reasons can prevent a gene from doing so. The binarization of the values might not be a reasonable hypothesis for certain genes. Some genes could be regulated by other, less important, switches which control only small groups of genes in the experiments performed. Noise can also prevent certain genes from following the model. A gene which fits the model is called a predicted gene. By definition, the binarized values of the predicted genes are constant inside each group of experimental conditions.

The solution should maximize the number of predicted genes while keeping their profiles as interesting as possible. There are many different ways to quantify the "interestingness" of a profile. This could be done using an information function:

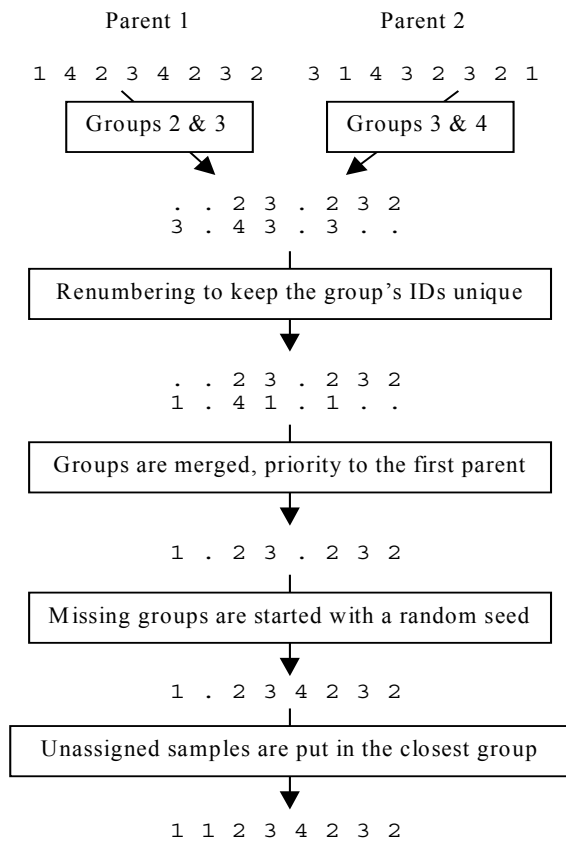
$$I = -\sum_i^{PG} (pI_i \log(pI_i) + p0_i \log(p0_i)) \quad (3)$$

where the sum is on the  $PG$  predicted genes.  $pI_i$  is the fraction of "1" in the predicted gene  $i$  and  $p0_i$  the fraction of "0". This function effectively sets a trade-off between the number of genes predicted and the information each of those carries.

In practice, it is necessary to modify the information function (3). The penalty for less informative genes in (3) is not large: with 10 samples, a gene with five "1" and five "0" is only 2.1 times more informative than a gene with nine "1" and one "0". The fits on small groups being likely to emerge from random fluctuations, this penalty seems too small. In order to widen the difference, the cube of the information is taken:

$$I = -\sum_i^{PG} (pI_i \log(pI_i) + p0_i \log(p0_i))^3 \quad (4)$$

This modification increases the importance of the information in the trade-off between the number of genes predicted and the information they carry. Even with this modification, solutions with groups formed of just one experimental condition are still a concern. To address this issue, the genes for which only one condition has a different binary value than the others are excluded from the calculation of the information function.



**Figure 1. Example of the creation of a child in the grouping genetic algorithm**

The maximum of (4) lies on top of a narrow hill. The function decreases very fast when the grouping is not perfect because one bad group may be enough to render the information predicted null. Any non-exhaustive search algorithm will tend to converge to a local maximum, consisting of many groups containing just one sample. In order to help the search for the global maximum, a widening function is added to the information function to soften the base of the hill. The idea is to evaluate not only the quality of a complete solution, but also the quality of the groups forming the solution.

Certain genes may follow the model, i.e. have constant values inside each group, only on some subsets of groups. The value of the widening function for each gene is the information function obtained on these subsets. This information is multiplied by the fraction of the conditions which appears in that subset, in order to lower the values for the solutions based on few groups. The best possible subset is kept for each gene. This leads to the following widening function:

$$W = - \sum_i^{NPG} \max_{S_i} \left( \frac{n_i(S_i)}{n} (pI_i(S_i) \log(pI_i(S_i)) + p0_i(S_i) \log(p0_i(S_i))) \right)^3 \quad (5)$$

where the sum is on the  $NPG$  non predicted genes.  $S_i$  are the various possible subsets of groups for which the gene  $i$  is correctly predicted,  $n_i(S_i)$  is the number of experimental conditions in the groups in  $S_i$ , and  $pI_i(S_i)$  (resp.  $p0_i(S_i)$ ) is the number of “1” (resp. “0”) in the binarized gene  $i$  while keeping only the groups in  $S_i$ .

The function maximized is the sum of the information function (4) and the widening function (5), multiplied by a constant alpha:

$$F = I + \alpha W \quad (6)$$

Alpha should be small enough so that the maximum of (6) corresponds to the maximum of (4), but large enough so that  $\alpha W$  is larger than the values of (4) obtained with a solution comprising many small groups.

It is of course necessary to check that the maximum of (6) found is indeed a maximum of (4). If it is not the case, a new search is performed with a lower alpha.

### 3.1.2 Maximization of the Function

Two different things must be determined in order to maximize (6): the thresholds for the binarization of the genes, and the clustering of the experimental conditions. The thresholds are determined, for a given clustering, by an exhaustive search. The clustering is determined using a grouping genetic algorithm [5].

For this algorithm, a population of individuals is created. Each individual is encoded as a vector, with as many elements as there are experimental conditions. The values of the individuals represent the group membership of the conditions. For instance, an individual encoded as [ 1 2 1 2 ] has a first group consisting of the first and third conditions, and a second group consisting of the second and fourth conditions. At each round, the best individuals are selected using a tournament. Offspring is created from the best individuals. The offspring replaces the worst individuals from the last round. Mutations are then applied to the population.

The success of such algorithm depends on the choice of the mutation and crossover operators. Its effectiveness can be raised with the use of an appropriate heuristic. Since the problem is a modified clustering, k-means is chosen as the heuristic. It is used for the creation of the starting individuals. For the creation of a child, two parents are chosen (see figure 1). Some randomly chosen groups from each parent are inherited in the child, the groups being renumbered so that each group has an unique identifier. Conditions which belong to groups inherited from both parents are set to the first parent’s group. The number of groups being a parameter of the problem, if the number of groups in the child is too low, one randomly chosen condition among the ones which are not assigned to any group is assigned to each missing group. The conditions which do not belong to any group are assigned to the closest group, i.e. the group whose members have on average the highest correlation with the condition to assign. Another child is made with the parents inverted.

In order to widen the search, mutations are made on randomly chosen individuals. Two types of mutations are used. In the first, three groups are randomly selected. The samples belonging to those groups are clustered into three new groups using a k-means. In the second type of mutation, the group membership of a sample is randomly modified.

The number of groups is not determined by the model. When this number is too high, the solutions found tend to overfit the data. Insights concerning the experiments must be used to detect such overfitting and determine the real number of groups.

### 3.2 The Switches

Many different combinations of switches can fit the clustering found. Using Occam’s razor, the simplest solution should be favored. Simplicity here lies in the functions linking the switches and the predicted genes.

Those functions are, in general, in the form of equation (2):

$$b_{ij} = f_i (sw_{j1}, sw_{j2}, \dots, sw_{jN})$$

Often,  $b_{ij}$  does not explicitly depend upon all switches. A subset of switches can be enough to determine  $b_{ij}$ . We introduce simplicity here as the number of switches necessary to predict the value of gene. In the most extreme case,  $b_{ij}$  can be a value of just one switch,  $sw_{js}$ :

$$b_{ij} = f_i (sw_{js})$$

The solution should present as many of those simple functions as possible.

In practice, the information predicted directly by the switches is maximized. This information is calculated in a fashion similar to (4), except that here only the directly predicted genes are taken into account. Directly predicted genes are genes whose values correspond to the values of a switch, up to a “NOT” transformation. For example, in table 1, gene 1 is directly determined by the switch 1 and gene 4 is directly determined by the switch 2.

In complex cases, when the number of switches is very high, no gene is determined by a single switch anymore. In that case, some other type of simple functions (e.g. conjunctions of a few switches) should be considered. We suppose here that the situation is simple enough for a large number of genes to be directly determined by each switch.

For the search of the switches, the data can be simplified. Since the thresholds are known from the grouping, the data matrix can be considered as being binarized. Only the genes fitting the model are kept. The switches being only determined up to a “NOT” transformation, their values can be set to “0” in an arbitrary group. It remains then to find for each other group the values of the switches.

In a valid solution, each group of samples must have a different combination of switches. This constraint is stronger when the number of groups is the maximum possible for a given number of switches. In that case, among other things, every switch must have as many groups with “1” as groups with “0”. These strict constraints are a reflection of the unlikeness of having the maximum possible number of groups, especially when the number of switches is high.

Since the switches should maximize the information directly predicted, they necessarily correspond to the binarized values of a gene or its opposite. The information directly predicted by each of these switches is easy to calculate. The different possible switches are then sorted in function of the information they predict, in order to try the most likely solutions first. See the first part of figure 2 for an example of possible switches with the corresponding informations.

Groups	1	2	3	4	5	6	Number	Info
SW1	1	1	0	0	0	0	30	6.45
SW2	1	1	1	0	0	0	15	5.00
SW3	1	0	0	1	0	0	17	4.38
SW4	1	1	0	1	1	0	12	3.09
SW5	1	0	1	0	1	0	5	1.67
SW6	1	0	0	0	0	0	15	1.37

**Possible switches information data. Each group is supposed to have the same number of conditions. “Number” is the number of genes directly determined by the switches. “Info” is the total information predicted by a switch: it is the product of the information of the switch with the number of genes directly determined. The switches are sorted in function of “info”.**

Current switches in the partial solution	Best possible quality	Comments
1	19.35	OK
1, 2	-	Not valid, backtrack
1, 3	-	Not valid, backtrack
1, 4	12.63	OK
1, 4, 5	11.21	New best solution
1, 5	9.79	Insufficient quality, backtrack
2	15	OK
2, 3	13.76	OK
2, 3, 4	12.47	New best solution
2, 4	11.18	Insufficient quality, backtrack
3	13.14	OK
3, 4	10.56	Insufficient quality, backtrack
4	9.27	Insufficient quality, finished

**Trace of the program.**

**Figure 2. Example of the determination of the switches on an artificial data set.**

The solution is calculated using a branch and bound algorithm (see figure 2 for an example). This is done by firstly creating partial solutions consisting of just one switch, starting from the most informative switch. The validity and the quality of these partial solutions are checked. If by completing a partial solution with any other switch it is impossible to create a valid solution of a better quality than the best actual solution, the partial solution can be discarded. Otherwise, another switch is added to generate a new, more complete, partial solution and the process is started again. When a generated partial solution is indeed a full solution, it replaces the current best solution. Its quality can then be used as a new bound on the quality of the partial solutions.

A partial solution is valid only if it can be part of a complete solution in which each group has a different combination of

**Table 2. Results obtained with the cell cycle experiment.**

Time (min)	0	7	14	21	28	35	42	49	56	63	70	77	84	91	98	105	112	119
Group	1	1	4	4	4	2	2	2	2	2	3	3	3	2	2	2	2	2
Switch 1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Switch 2	1	1	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1

switches. This implies that the largest number of groups sharing the same combination of switches in the partial solution should be at most  $2^N$ , where  $N$  is the number of switches which still have to be determined.

The best solution which contains a partial solution has at most a quality equal to the quality of the partial solution plus  $N$  times the quality of the worst switch in the partial solution, where  $N$  is the number of switches which still have to be determined. This limit is due to the sorting of the switches in function of their quality, which implies that the switches that can be added to the partial solution are not better than any of the switches already in that solution.

These bounds based on the validity and the quality of the partial solutions limit dramatically the size of the search space, allowing a quick computation of the optimal solution.

#### 4. APPLICATION: ARTIFICIAL DATA

The algorithms were applied on artificial data sets, in order to check that they effectively determine the switch structure when such structure exists in the data. In those data sets, as is expected in the real data sets, there are three categories of genes: random genes, genes directly determined by one switch and genes which are a Boolean function of more than one switch.

The grouping algorithm was applied to a first artificial data set, consisting of 100 experimental conditions organized in 16 groups using 4 switches. 10 genes are directly determined by each switch, 60 genes are determined by a combination of switches and 900 genes act as noise. The algorithm was able to recover the right grouping and thresholds in a few hours. This setup being much more difficult than what is expected with real data, the performance of the grouping algorithm seems satisfying, although a faster version is certainly desirable. The switch determination algorithm was able to recover the switches from the grouping very quickly.

Secondly, a more reasonable case was created. In this setup, there are 1000 genes and 50 experimental conditions clustered into 6 groups using 3 switches. 20 genes are directly determined by each switch, 240 are determined by a combination of switches and the 700 remaining are random.

In this setup, when the algorithm is asked to find six groups in the data, it does so in a relatively short time (a couple of minutes). The switches are correctly recovered from the group information. When the algorithm is asked to find seven or eight groups, the best solutions have single conditions as new groups, the rest fitting the real solution. The quality of the solutions as estimated by (4) does not raise much when the number of groups is

increased. When the algorithm is asked to find four or five groups, it finds solutions similar to the real one, except that some groups are merged. The quality of these solutions is much lower than the quality of the correct solution.

We expect that on real data, the quality of the solutions will keep raising as the number of groups is increased, because there are probably many “small” switches which explain small parts of the data. Nevertheless, the pattern of small groups should still appear, showing the likeliness of overfitting. We use this as a clue that the number of groups is too high.

#### 5. APPLICATION: REAL DATA

We present here two applications of the technique to real world data. We have not tried to make breath-taking new discoveries, but simply to show that the binary switch model can be used to explain a large part of real regulatory networks. The switches discovered are also identified with simple, high level concepts, showing the power of the technique as a tool for biological understanding.

##### 5.1 Cell Cycle

The first set of data comes from the study of Spellman *et al.* [13] concerning the cell cycle in the yeast. There are a few different experiments in that study, which differ in the technique used to synchronize the cells. The one discussed here is alpha factor. Similar results were obtained with *cdc15*.

The data being very noisy, some pre-processing had to be done before the identification of the parameters of the model could be performed. Firstly, a low-pass filter was used in order to remove some noise from the data. The filter was an acausal, zero-phase filter of order 10, determined from a Butterworth filter [11]. The cutting frequency was half of the Nyquist frequency. This filtering rendered the data much smoother. Secondly, certain genes were excluded from the data set. The selection criteria were that at least 30% of the gene’s derivative was conserved after smoothing, and that the smoothed gene had at least a 2-fold variation across the samples. Using this filter, only 608 genes were kept.

The identification of the parameters of the model was performed with four groups and two switches. The results can be seen in table 2. The first switch could be understood as standing for genes which are controlled by the arrest of the cell cycle necessary to synchronize the cells. The second switch is relative to the cell cycle itself. This shows that the technique is able to recover the expected structure in the data.

Among the 608 genes taken into consideration, 251 (41%) fit the groups. 45 genes are directly determined by the first switch, 36 by the second. Group 1, which is Switch 1 AND Switch 2, has quite a lot of success. This could be due to random fitting of noise (it is a small group) or to the sharp raises or falls which seem to be present for many genes in the first time points.

**Table 3. Results obtained with the metal experiments.**

	WT Zn-	WT Zn=	WT Zn+	Zap Zn-	Zap Zn=	Zap Zn+	MacC	MacB	WT Cu-	Cu+ 30	Cu+ 60
<b>3 groups</b>	1	2	2	1	2	2	2	2	3	3	3
<b>4 groups</b>	1	4	4	1	3	3	2	2	1	1	1
<b>Switch 1</b>	1	0	0	1	0	0	0	0	0	0	0
<b>Switch 2</b>	1	0	0	1	0	0	0	0	1	1	1

In order to deduce the regulation network of the 41% of the genes which fits the model, it would only remain to identify the switches with biological counterparts. This task might be performed using gene expression alone if these counterparts are gene expressions. For the others, like probably for the switch relative to the cell cycle arrest, no identification is possible using only the data available. The limits of the deductions that can be made using gene expression data alone are reached.

## 5.2 Metal Work

We have taken two 2-colors microarray data sets concerning the reactions of yeast to variations in the concentration of zinc [9] and copper [6] in the environment. These two data sets were chosen because the experimental conditions are somewhat similar, raising the hope to find some common trends. Nevertheless, the strains of yeast as well as the details of the experimental protocols are different. This demonstrates the applicability of the technique on heterogeneous data.

After some transformations, it is possible to reduce the measures of the first data set to six experimental conditions and the measures of the second to five other conditions. The two data sets are then merged (see table 3). WT is the wild type yeast, different in the two experiments. Zap is a WT yeast with the ZAP gene knocked out. This reduces its reaction to the lack of zinc. Mac is a WT yeast, with the gene MAC1 constitutively expressed. MAC1 regulates the expression of high affinity copper intake genes. There are two Mac experiments, based on two different strains: MacC, taken from yeast strain CM66J grown exponentially, and MacB, taken from yeast strain BR10 grown to late log phase. In the zinc experiment, WT and Zap cells were cultivated in deplete zinc (Zn-), replete zinc (Zn=) and excess zinc (Zn+) conditions. In the copper experiment, Mac cells were cultivated in normal condition and WT cells were cultivated in deplete copper medium (WT Cu-), and in excess copper medium for 30min (Cu+ 30) and 60min (Cu+ 60).

Since the standards in the two groups of experiments are not identical, the genes were normalized separately in each group. This was done by dividing, separately in the zinc and the copper experiment, the values for each gene by the mean of its values across the conditions of the experiment.

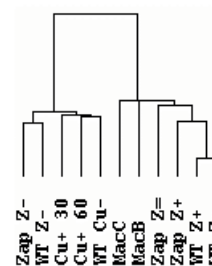
The group search algorithm was applied to the data set, with three and four groups. As shown in table 3, with four groups the best solution has three groups made of pairs of conditions. The experimental conditions in those pairs are very similar. The excess zinc condition is similar to the replete zinc condition, and the two Mac experiments are also similar. This solution can be considered as having groups formed of essentially the same samples, and so is probably due to overfitting. The solution with three groups was kept.

The switch determination algorithm was then run on those three groups, leading to the two switches shown in table 3. It is possible to assign a simple meaning to these switches. The first one is “on” only in the two experiments where the cells are lacking zinc. This switch could be understood as a “zinc starvation” signal. The second switch is “on” in the experiments where the cells are in a difficult situation, lack or excess of something. This switch could be understood as a “sickness” signal. The excess of zinc is not very harmful for the yeast, which explains why those conditions are in the same group than the replete zinc conditions.

Among the 960 genes which are expressed at a reasonable level in all experiments and show more than 3-folds variation across the conditions, 369 (38%) fit the simple explanation given. Among those genes, 161 (44%) are directly determined by the “sickness” switch and 203 (55%) by the “zinc starvation” switch. Since the algorithm is sensitive to noise, as one noisy measurement is sufficient to consider a gene as not explained, and since the experiments were done using different standards, those are quite high figures, showing that a large part of the behavior of a cell can be understood with simple concepts.

The switches found are explained here in a “high-level” way, but we expect that some biological means exist in the yeast which pilot the regulations described. The identification of these means is impossible with gene expression data alone. However, the distribution of the switches provides some clues which might prove useful for such identification.

In a case like the one shown here, where one switch is included in the other, a hierarchical clustering algorithm should give a comparable result (see Figure 3). Nevertheless, the clustering algorithm does not offer a high-level explanation like the technique outlined here does, nor does it establish a link between the variation of the expression of the genes and the clustering. Besides, in this case, the clustering algorithm does not discover the “zinc starvation” switch.



**Figure 3. Clustering of the metal experiments.**

To our knowledge, this is the first time two different data sets are compared in order to deduce something about the samples. Such comparisons have only been done in order to predict gene's functions using clustering or classification techniques. We have shown here that such comparison can be performed and be meaningful. The distribution of the switches and the links established between those switches and the expression of certain genes might prove useful for a biologist. For instance, to an observer interested in the result of a lack of zinc in the environment, the genes which react in a similar way to a lack of copper may be of little interest. A comparison between the two experiments permits to spot such genes and thus to suggest unsuspected relations.

## 6. CONCLUSION

The model presented here allows the description and determination from the data of a large part of the gene expression regulatory network in a simple and consistent fashion. The rest of the network can be determined by identifying the switches with biological realities and discover how they are regulated. This complex task is simplified by the pattern of expression of the switches, which should permit to identify them with measured genes when such identification is possible.

We have demonstrated here that it is possible to deduce the value of the variables which regulate gene expression even when those variables are not measured. This identification is possible because a small number of causes create a large number of different effects, and because the possible links between the causes and the effects are modeled precisely. This is performed here with regulations modeled as Boolean functions, but the same could certainly be done for other, more realistic, models of regulation of gene expression.

We have also demonstrated that it is possible to perform the binarization of the data using the same framework as the one used for the determination of the gene regulations. Such method should lead to better solutions than a discretization based on some pre-processing algorithm.

The binary switch model, like any Boolean model, is only a very simplistic description of the possible links between the regulators and the genes regulated. Nevertheless, we have shown here that this simplicity does not preclude the ability of the model to fit real data and to suggest interesting links. Simpler models have the advantage of being more understandable and less prone to overfitting. As long as they are expressive enough to describe the systems studied, they usually outperform more complex models. This may explain the maybe surprising success of the real world applications presented.

With the technique presented, it is possible to perform a comparison of different data sets. This way, an explanation of common traits between them can be obtained. As shown in the metal work example, this could be useful to focus the search on genes whose regulations are specific to certain experiments. The simplicity and understandability of the explanations given by the switches should prove useful for the selection of the most interesting genes.

A limitation of the model is that the links between the switches and the predicted genes must be perfect. As the number of samples increases, the likeliness of having at least one measure which does not fit because of the noise raises dramatically. This prevents the application of the framework on large data sets. A better version should allow for errors in the predictions, for instance by using a probabilistic Bayesian model instead of the Boolean model presented here.

In order to compare data sets obtained by different laboratories, with different protocols, it might be better to discretize independently each group of experiments. In our framework, that could be done by using a different threshold for each group of experiments. This should make the comparison of heterogeneous experiments less dependent upon the normalization of the gene expressions. This shows as well the importance of performing the discretization as a part of the estimation of the regulations, and not as a separate process.

Finally, the determination of the groups and the determination of the switches are done independently, which is probably not optimal. The simultaneous determination of the clustering, the thresholds and the switches should lead to better-defined problems.

Even with those limitations, the technique presented here is already applicable to real data sets, offering some interesting results. Would those limitations be lifted, it might be a useful tool for the discovery of regulation networks and the interpretation of biological processes.

## 7. ACKNOWLEDGMENTS

We thank Jacques E. Dumont for insightful discussions. This work was supported by the Région Wallonne and UCB Pharmaceuticals.

## 8. REFERENCES

- [1] Akutsu, T., Miyano, S., and Kuhara, S. Identification of Genetic Networks from a Small Number of Gene Expression Patterns Under the Boolean Network Model. In Proceedings of PSB 1999, 17-28
- [2] Akutsu, T., Miyano, S., and Kuhara, S. Algorithms for Inferring Qualitative Models of Biological Networks. In Proceedings of PSB 2000, 290-301.
- [3] Chen, T., He, H.L., and Church, G.M. Modeling Gene Expression with Differential Equations. In Proceedings of PSB 1999, 29-40.
- [4] D'haeseleer, P., Wen, X., Fuhrman, S., and Somogyi, R. Linear Modeling of mRNA Expression Levels During CNS Development and Injury. In Proceedings of PSB 1999, 41-52.
- [5] Falkenauer, E. Genetic Algorithms and Grouping Problems. Wiley (1998)
- [6] Gross, C., Kelleher, M., Iyer, V.R., Brown, P.O., Winge, D.R. Identification of the copper regulon in *Saccharomyces cerevisiae* by DNA microarrays. *Biol. Chem.* 41 (2000), 32310-32316

- [7] Holter, N.S., Maritan, A., Cieplak, M., Fedoroff, N.F., and Banavar, J.R. Dynamic modeling of gene expression data. *Proc. Nat. Acad. Sci. USA* 98 (2001), 1693-1698.
- [8] Liang, S., Fuhrman, S. and Somogyi, R. REVEAL, A General Reverse Engineering Algorithm for Inference of Genetic Network Architectures. In *Proceedings of PSB 1998*, 18-29.
- [9] Lyons, T.J., Gasch, A.P., Gaither, L.A., Botstein, D., Brown, P.O. and Eide, D.J. Genome-wide characterization of the Zap1p zinc-responsive regulon in yeast. *Proc. Nat. Acad. Sci. USA* 97 (2000), 7957-7962.
- [10] Maki, Y., Tominaga, D., Okamoto, M., Watanabe, S., and Eguchi, Y. Development of a system for the inference of large scale genetic networks. In *Proceedings of PSB 2001*, 446-458.
- [11] Matlab Signal Processing Toolbox. MathWorks inc. (1999)
- [12] Pe'er, D., Regev, A., Elidan, G., and Friedman, N. Inferring subnetworks from perturbed expression profiles. In *Proceedings of ISMB 2001, Bioinformatics* 17, Suppl 1, S215-S224.
- [13] Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D. and Futcher, B. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell* 9 (1998), 3273-3297.
- [14] Thieffry, D., and Thomas, T. Qualitative analysis of gene networks. In *Proceedings of PSB 1998*, 77-88.
- [15] Weaver, D.C., Workman, C.T. and Stormo, G.D. Modeling Regulatory Networks with Weight Matrices In *Proceedings of PSB 1999*, 112-123.
- [16] Wessels, L.F.A., Van Someren, E.P., and Reinders, M.J.T. A Comparison of Genetic Network Models. In *Proceedings of PSB 2001*, 508-519.