Analysis of Gene Expression Data **Spring Semester, 2007**

Lecture 9: June 21, 2007

Lecturer: Ron Shamir *Scribe: Maria Natanzon and Yevgenia Koren*[1]

9.1 Genetic networks

9.1.1 Preface

An ultimate goal of a molecular biologist is to use genetic data to reveal fundamental cellular processes, and their impact on complex organisms. In order to achieve this goal one has to study how complex systems of several genes and proteins function and interact.

Definition A *genetic network* is a set of molecular components such as genes, proteins and other molecules, and interactions between them that collectively carry out some cellular function.

9.1.2 Experimental strategies

Using a known structure of such networks it is sometimes possible to describe the behavior of cellular processes, reveal their function and determine the role of specific genes and proteins in them. That is why one of the most important and challenging problems today in molecular biology is that of *functional analysis* - discovering and modeling Genetic Network from experimental data.

Biological tools

There are two central approaches in addressing this problem: The first approach tries to find out the relation between two specific genes. An example of this approach is the usage of 2-hybrid systems [1]. The second approach takes ”snapshots” of the expression levels of many genes in different conditions, and tries to describe the network of relations between genes that will fit these observations. An example of this approach is the usage of DNA microarray, commonly used to monitor gene expression at the level of mRNA. The main contribution of this technology is that numerous genes can be monitored simultaneously, making it possible to perform a global expression analysis of the entire cell. In this scribe we will cover techniques related to the second approach.

Genes and proteins functionalities and interactions can be examined by causing perturbations to genes in the network.

There are two possible types of perturbations:

[1] Based on scribes by Karin Inbar and Anat Lev-Goldstein 2005, Omer Czerniak and Alon Shalita 2004, Meital Levy and Giora Unger 2002, Koby Lindzen and Tamir Tuller 2002

- *Genetic* perturbation is an update of the expression levels of one or more genes by *knockout* (removal of the gene), or *overexpression* (setting the expression higher than it's usual level).
- *Biological*(environmental) perturbation is altering one or more non-genetic factors, such as a change in environment, nutrition, or temperature. Such biological experiments are very costly and very few such perturbations may be performed at one time. Thus, reducing the number and cost of experiments is crucial.

The functional analysis of the data can be defined as a computational problem, aiming to infer some plausible model of the network from the observations, while keeping the number or cost of biological experiments at a minimum. The model should describe how the expression level of each gene in the network depends on external stimuli and expression levels of other genes.

Such methods can be also used for construction of a knowledge-base of gene regulatory networks, and verification of pathways or genetic network hypotheses.

9.1.3 Genetic networks

Genetic networks describe complicated functional pathways in a given cell or tissue, representing processes such as metabolism, gene regulation, transport and signal transduction.

Let us examine several examples of genetic networks describing various processes:

1. **Expression of the gene proB**

   Figure 9.1 depicts the gene's expression and its role in catalyzing a specific chemical reaction in the cell. The proB gene is being expressed into the gamma-glutamyl-kinase protein, which catalyzes a reaction involving glutamate and ATP, that produces gamma-glutamyl-phosphate and ADP compounds.

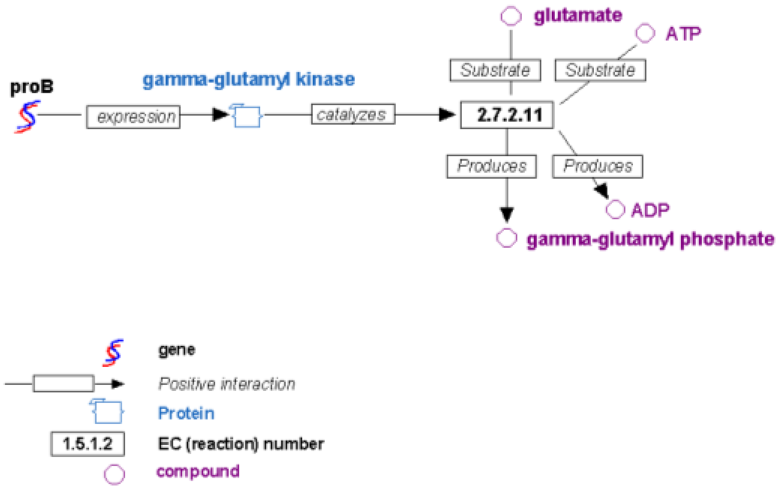


Figure 9.1: An example of the role of gene expression in catalyzing chemical reactions.

2. **A simple metabolic pathway - Proline biosynthesis**

   The next example is part of a simple metabolic pathway, involving a chain of generated proteins, which is shown on Figure 9.2. One of the final products of the chain, proline, inhibits the initial reaction that started the whole process. This "feedback inhibition" pattern is highly typical to genetic networks, and serves to regulate the process execution rate.

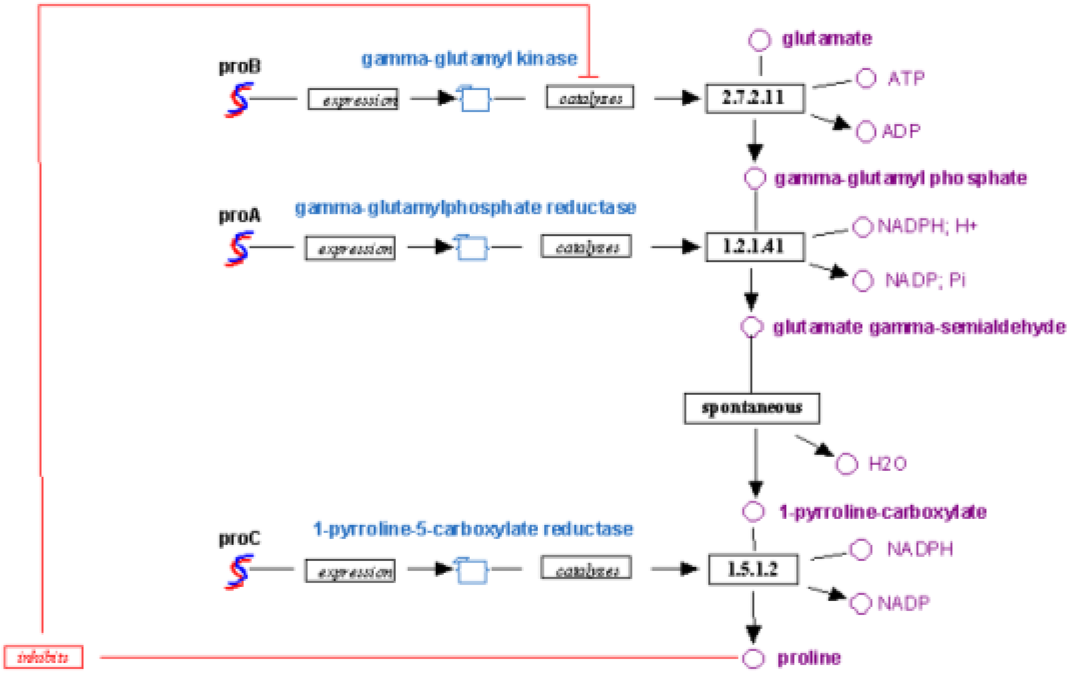


Figure 9.2: An example of a metabolic pathway: Proline biosynthesis.

3. **Methionine biosynthesis in E-coli.**

   The following two figures (9.3 and 9.4) show a more complex genetic network, describing Methionine biosynthesis in E-coli. In this network, for example, one can observe two genes (metL and metB) activating two different pathways in the process. The second figure is a schematic representation of the pathway, with most nodes omitted, but it can give a better idea of the overall topology.

4. **Signal transduction network**

   This example, depicted in Figure 9.5 is describing a behavioristic change of a cell as a reaction to an external event - a complex cellular process initiated by a signaling protein, arriving from outside of a cell, through the cell membrane. This process eventually affects gene expression in both the cytoplasm and inside the nucleus.

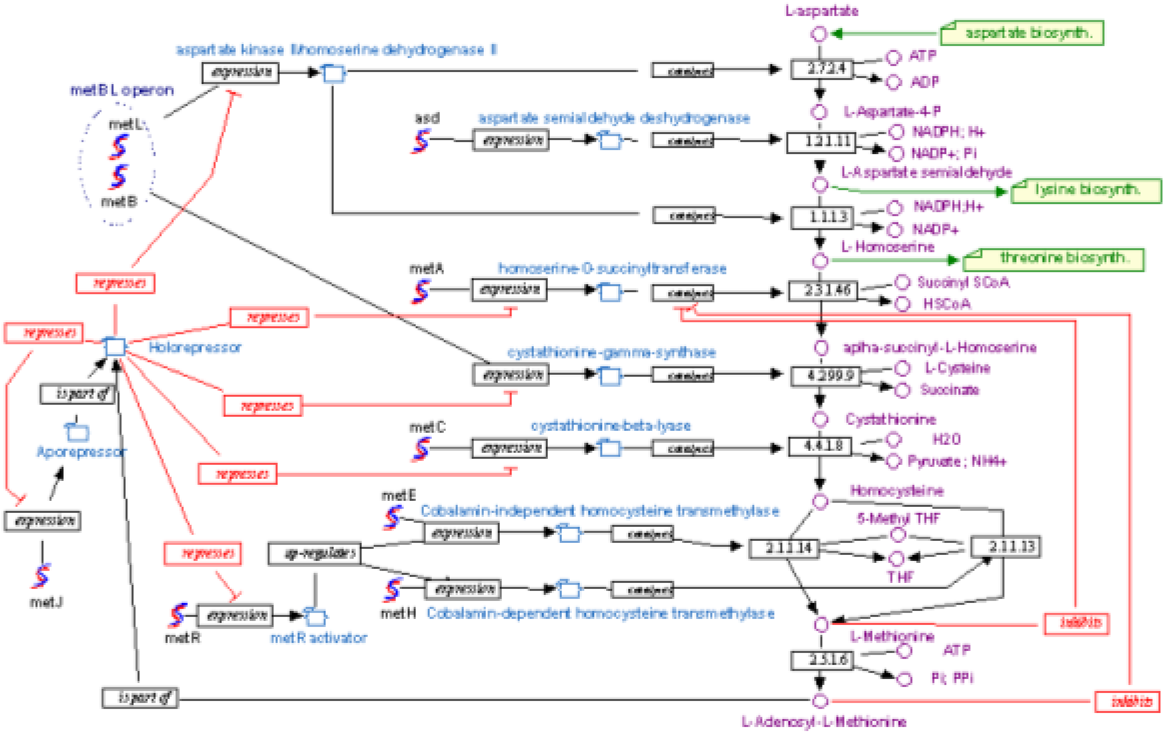


Figure 9.3: Methionine biosynthesis network in E-coli.

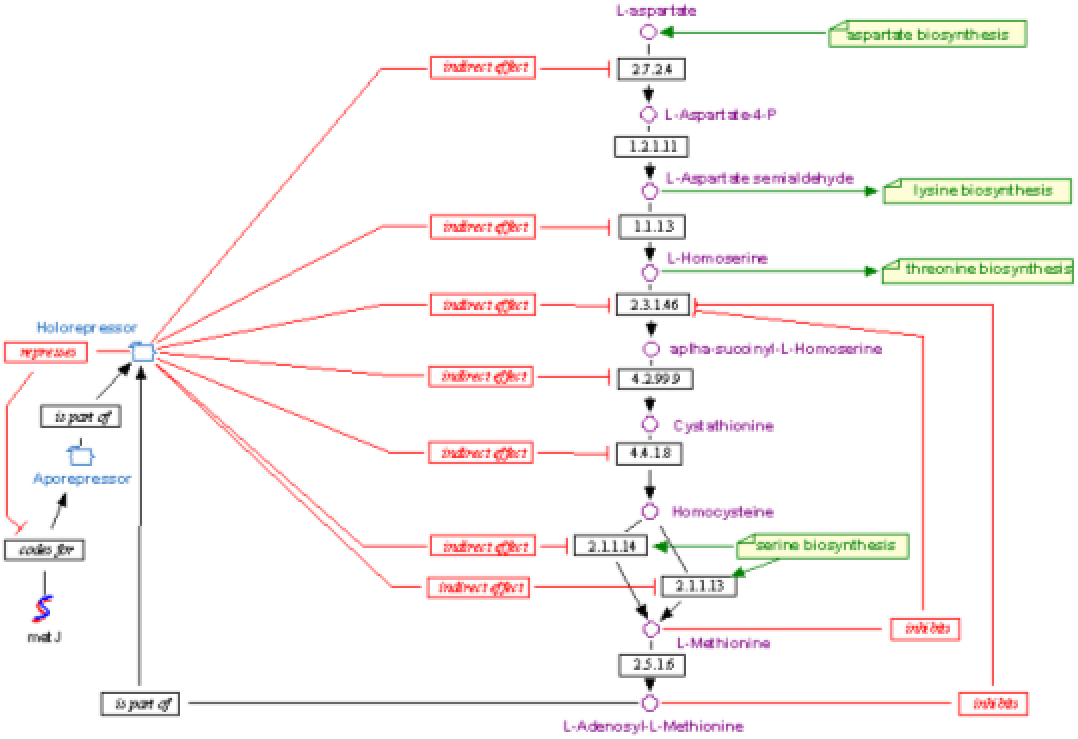


Figure 9.4: Schematic representation of the biosynthesis pathway presented in Figure 9.3.

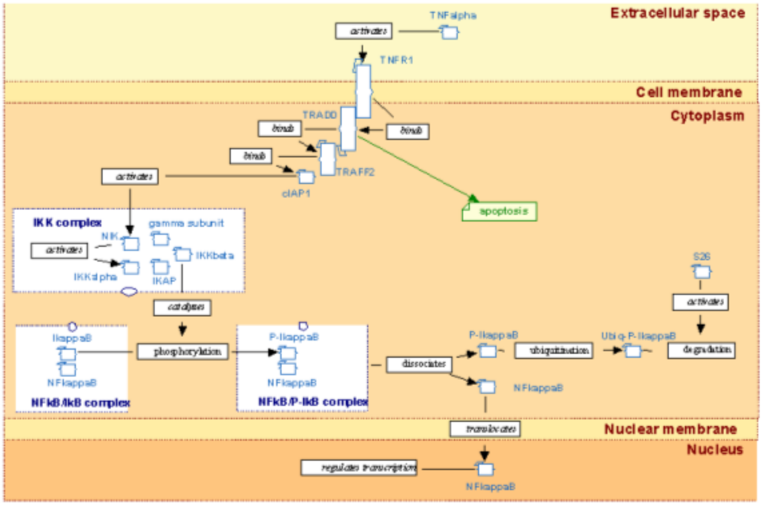


Figure 9.5: A genetic network that performs signal transduction from outside the cell into the nucleus.

5. **Sea urchin endomesoderm development**

   The following example, depicted in Figure 9.6, shows a semi-digital representation of a genetic network controlling early development of sea urchin endomesoderm [2].

Genetic networks structure

Let us examine the genetic network structure components:

1. **Linear Cause-Effect Chain**

   The simplest structure is depicted in Figure 9.7a. The growth is unlimited because there is no feedback (DAG structure).

2. **Feedback loops**

   The following structure, depicted in Figure 9.7b, shows a little more complicated structure, in which the growth is controlled by feedback signals.

3. **Web of interacting circuits**

   The following structure, depicted in Figure 9.7c, shows a more complicated structure, where there are feedback signals running across the network, that is, feedback signals arriving from other circuits in the network , producing crosstalks.

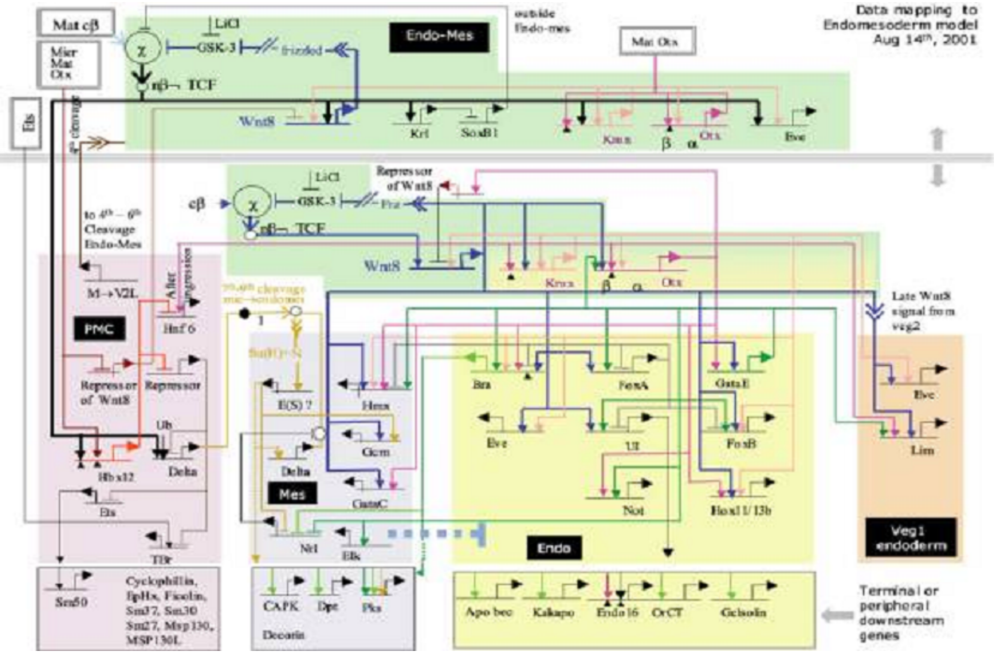
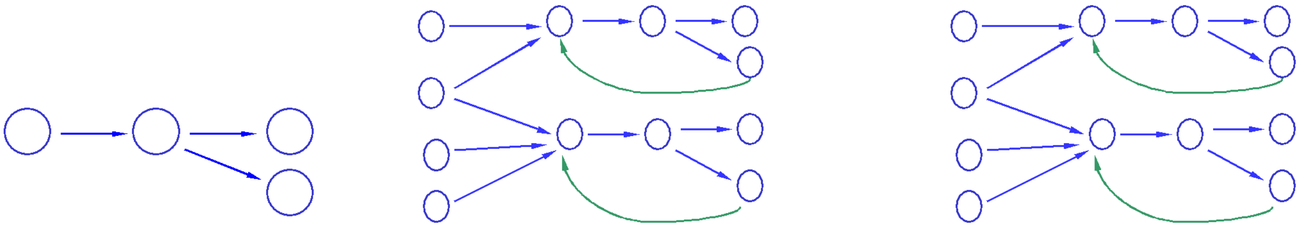


Figure 9.6: A genetic network that controls early development of sea urchin endomesoderm.



a. Linear cause-effect chain. b. Feedback loops: modular circuits. c. Web of interacting circuits.

Figure 9.7: Genetic Network Structure

9.1.4 Control points

There are several points of control during the process of protein creation:

- Basic transcription control (of the whole gene)
- RNA processing and splicing
- mRNA transport or mRNA degradation
- miRNA silencing - short RNA sequences preventing translation of mRNA into protein.
- Translational control
- Protein activity control: post-transitional modifications

The control points are depicted in Figure 9.8

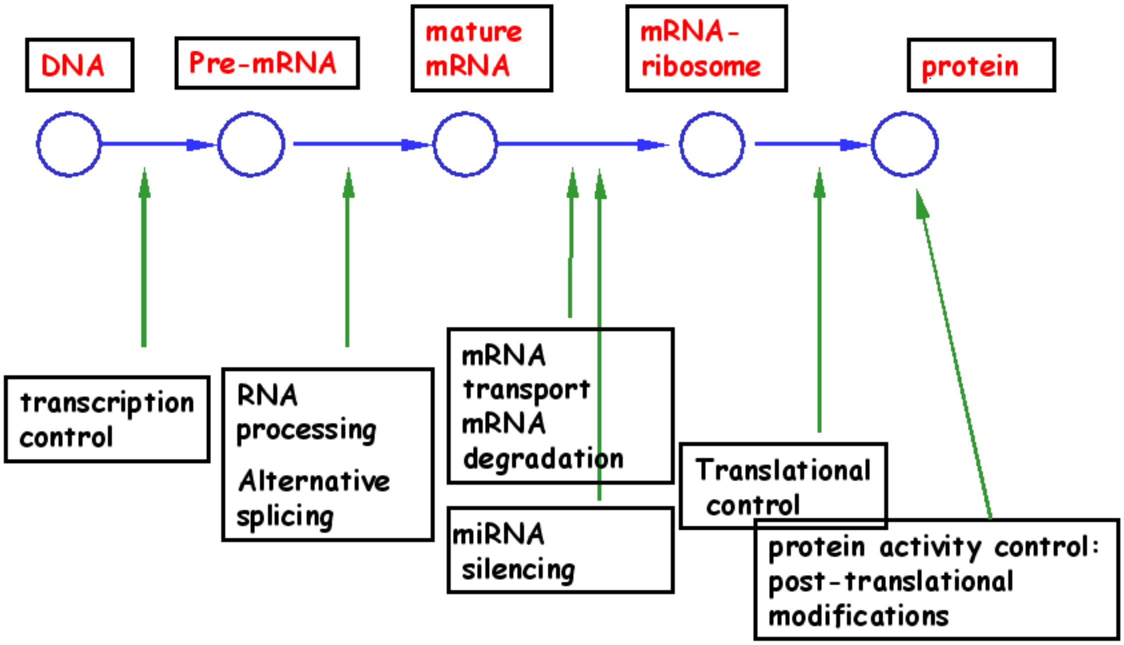


Figure 9.8: Control Points

9.1.5 Goal of genetic network analysis

- Construct a knowledge-base of gene regulatory networks
- Verify pathways or gene network hypotheses
- Reconstruct gene networks from experimental data

A crucial point is the number and cost of the perturbation experiments. So another goal would be optimizing experiments to verify or reconstruct networks.

9.1.6 Genetic network models

In the process of modeling a genetic network, one tries to find out which components are involved in the network and the interactions between them. Several models proposed in the literature to capture the notion of genetic networks and allow mathematical solutions of the computational problem of modeling biological processes.

Simpler models have been suggested, which might be less accurate, but can enable us to understand network properties. These models may use mRNA or protein concentration as gene expression level, and may be expressed in continuous or discrete way. Sometimes simple boolean approximation is appropriate. Some simple model types:

- **Linear model:**
  This model, proposed by D'haeseleer *et al.* [5, 4], assumes that the expression level of a node in a network depends on a linear combination of the expression levels of its neighbors.

- **Boolean model:**
  Proposed by Kauffman[7]. This will be discussed in detail later on.

- **Bayesian model:**
  Proposed by Friedman *et al.* [6]. It attempts to model the behavior of the genetic network as a joint distribution of different elements. This will be described in the next lecture.

9.1.7 Boolean network model

The boolean model assumes only two distinct levels of gene expression - 0 and 1. According to this model, the value of a node at time $t+1$ is a boolean function of the values at time t of the genes that control it, thus assuming one step memory. This kind of model does not support more complicated reactions, that might require more time than others. In addition, it does not support any kind of stochastic behavior. A network is represented by a directed graph $G = (V, F)$, where:

- V represents nodes (elements) of the network.

- F is a set of boolean functions (see below), that defines the topology of edges between the nodes. These functions are deterministic and synchronous.

A node may represent either a gene or a biological stimulus, where a stimulus is any relevant physical or chemical factor which influences the network and is itself not a gene or a gene product. Each node is associated with a steady-state expression level x_v, representing the amount of gene product (in the case of a gene) or the amount of stimulus present in the cell. This level is approximated as high or low and is represented by the binary value 1 or 0, respectively.

Network behavior over time is modeled as a sequence of discrete synchronous steps. The set $F = \{f_v | v \in V\}$ of boolean functions assigned to the nodes defines the value of a node in the next step, depending on the current values of other nodes, which influence it. The functions f_v are uniquely defined using truth tables. An edge directed from one node to another represents the influence of the first gene or stimulus on that of the second. Thus, the expression level of a node v is a boolean function f_v of the levels of the nodes in the network which connect (have a directed edge) to v. All updates are deterministic and synchronous.

Definition If genes A and B regulates the expression level of gene C, then:

- A and B are *regulators* in this context.

- C is the *regulatee* in this context.

- f_c is the logical function governing the level of c: X_C(t+1) = f_c(X_A(t), X_B(t)).

For example, see Figure 9.9.

Definition :

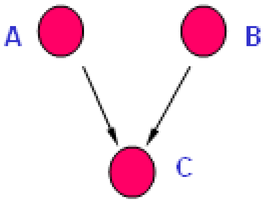


Figure 9.9: $C(t+1)$ the regulatee depends upon the regulators $A(t)$ and $B(t)$. Note that the *wiring diagram* alone doesn't provide the logical function of the regulator. Thus, in this example, it's not clear for example if X_C(t+1) = X_A(t) *OR* X_B(t) or if X_C(t+1) = X_A(t) *AND* X_B(t) or any other logical combination.

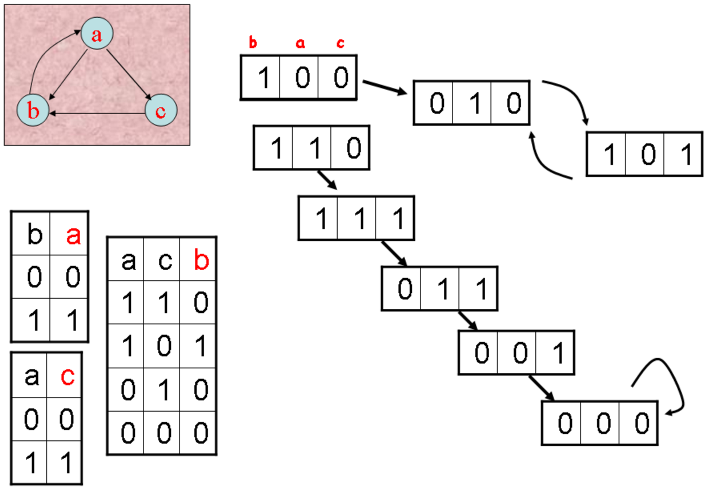
- A *global state* is the vector $(x1(t), x2(t), ...xn(t))$ that represents the levels of all the genes at time t. If the global state of time t is known, and the logical functions are known, its possible to compute the next vectors.
- A *trajectory* is a sequence of consecutive states of the network. It can be viewed as a list of N-dimensional vectors (N being the number of nodes in the network), each representing a state (a path is the graph).
- An *attractor* is a loop in the graph.
- A *basin of attraction* is a set of all the states that lead to some attractor.

Figure 9.15 gives an example of a simple boolean network and associated truth tables. This example shows a network of three nodes - a, b and c. As one can see, the expression of c directly depends on the expression of a, which in turn directly depends on b. Note that a influences more than one node, b and c (*"pleiotropic regulation"*), and that b is influenced by more than one node (*"multigenic regulated"*). The assignment of values to nodes fully describes the *state* of the model at any given time. The change of model state over time is fully defined by the functions in F. Initial assignment of values uniquely defines the model state at the next step and consequently, on all the future steps. Thus, the network behavior is represented by its *trajectory*.

In figure 9.15 two such trajectories are presented for the sample network. Since the number of possible states is finite, each one of possible trajectories eventually ends up in a single *attractor*.

One or more attractors are possible. The network in our example has two attractors - one is the steady state $(0, 0, 0)$, and the other is a cycle $(0, 1, 0) \leftrightarrow (1, 0, 1)$. The attractors are reached when $t \rightarrow \infty$. In a finite boolean network, one of the attractors is reached in a finite time.

States in genetic networks are often characterized by *stability* - "slight" changes in value of a few nodes do not change the attractor. Biological systems are often *redundant* to ensure that the system stays stable and retains its function even in the presence of local anomalies. For example, there may be two proteins, or even two different networks with the same function, to backup each other.



b	a
0	0
1	1

a	c
0	0
1	1

a	c	b
1	1	0
1	0	1
0	1	0
0	0	0

Figure 9.10: An example of a boolean model graph, functions and trajectories. The upper trajectory has a cycle of 2 steady states, while the lower trajectory ends in a single steady state. The basin of attraction of the upper trajectory is $[1,0,0],[0,1,0],[1,0,1]$, and the attractor of the upper trajectory is $[0,1,0],[1,0,1]$

9.2 Identification of gene regulatory networks by gene disruptions and overexpressions

9.2.1 Preface

This section is based on the article of Akutsu *et al.* [3]. Almost all proofs and all figures were taken from this paper. In this section we show how to identify a gene regulatory network from data obtained by multiple gene perturbations (disruptions and overexpressions) taking into account the number of experiments and the complexity of experiments. An experiment consists of parallel gene perturbations and their total number is the complexity of an experiment.

9.2.2 Model Description and Definitions

We define the gene regulatory network as in the previous section. We further assume that it satisfies the following conditions:

1. When the boolean function f_v assigned to v has k inputs, k input lines (directed edges) come from k distinct nodes $u_1, ..., u_k$ other then v. Only relevant variables are included in f_v. f_v is the regulation rule of node v.

2. For each $i = 1, ..., k$ there exists an input $(a_1, .., a_k) \in \{0,1\}^k$ with $f_v(a_1, ..., a_k) \neq f_v(a_1, .., \bar{a}_i, ..., a_k)$ where $\bar{a}_i$ is a complement bit of a_i.

3. A node v with no inputs has a constant value (0 or 1).

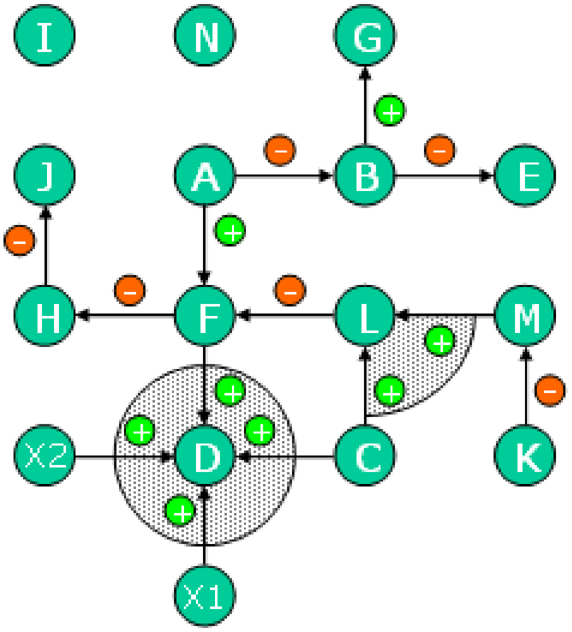
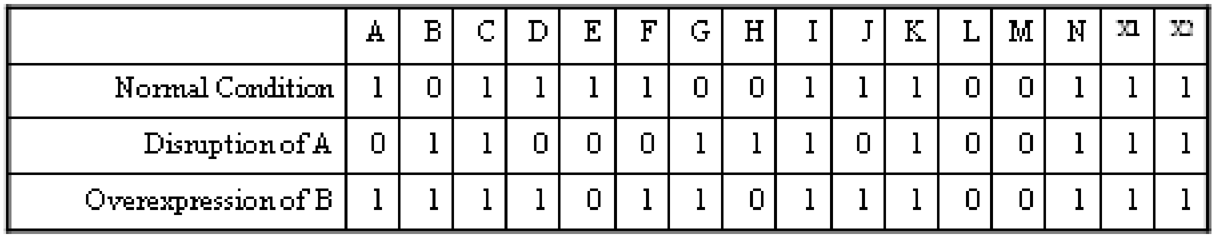


Figure 9.11: Source: [3]. Example of a gene regulatory network with 16 genes (⊕ means "activation" and ⊖ means "deactivation" of the gene). Gene F is *activated* by gene A and is also *inactivated* by gene L ($f_F(A, L) = l(A) \wedge \neg l(L)$). Gene D is expressed if all its predecessors $C, F, X1, X2$ are expressed (AND - node).



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	X1	X2
Normal Condition	1	0	1	1	1	1	0	0	1	1	1	0	0	1	1	1
Disruption of A	0	1	1	0	0	0	1	1	1	0	1	0	0	1	1	1
Overexpression of B	1	1	1	1	0	1	1	0	1	1	1	0	0	1	1	1

Figure 9.12: Source: [3]. Gene expressions by disruption and overexpression from the gene regulatory network of Figure 9.11 (0 - the gene is not expressed , 1 - the gene is expressed).

Definition The *state* of a gene v is active (inactive) if the value of v is 1 (0).

Definition The node v is called $AND(OR)$ node if the value of $f_v(a_1, ..., a_k)$ is determined by the formula $\ell(u_1) \wedge \ell(u_2) \wedge ... \wedge \ell(u_k)$ $(\ell(u_1) \vee \ell(u_2) \vee ... \vee \ell(u_k))$, where $\ell(u_i)$ is either u_i or $\neg u_i$.

Definition An edge (u, v_i) is called an *activation edge (inactivation edge)* if $\ell(u_i)$ is a positive literal (negative literal).

For a gene v, *a disruption* of v forces v to be inactive and *overexpression* of v forces v to be active. Let $x_1, ..., x_p, y_1, ..., y_q$ be mutually distinct genes of G. An *experiment* with gene overexpressions $x_1, ..., x_p$ and gene disruptions $y_1, ..., y_q$ is denoted by $e = \langle x_1, ..., x_p \ , \neg y_1, ..., \neg y_q \rangle$. The *cost* of e is defined as $p + q$. Three examples of gene expression conditions (normal, disruption of gene A, overexpression of gene B) are presented in figure 9.12.

Let us define the nodes with fixed values given *experiment* e:

Definition The node v is said to be *invariant* if it satisfies one of the following conditions:

- v belongs to e, i.e. in experiment e, v is disrupted or overexpressed and it's value does not change during the experiment.
- v has in-degree 0.
- v depends only on invariant nodes.

We now define different types of states of gene regulatory network G:

1. A *global state* of G is a mapping $\psi : V \to \{0, 1\}$, that represents the levels of all nodes at a specific point of time. The global states of the genes need not be consistent with the gene regulation rules. Disruption or overexpression of a specific gene, may produce a conflict within the network. For instance in 9.11 the disruption of gene K yields the activation of L, which inactivate F, while gene A expresses and activates F.

2. The global state ψ of G is *stable* under experiment $e = \langle x_1, ..., x_p \ , \neg y_1, ..., \neg y_q \rangle$ if $\psi(x_i) = 1$ $(i = 1, ..., p)$, $\psi(y_j) = 0$ $(j = 1, ..., q)$ and it is consistent with all gene regulation rules, i.e., for each node v with inputs $u_1, ..., u_k$, $\psi(v) = f_v(\psi(u_1), ..., \psi(u_k))$. Otherwise, it is called *unstable*.

3. The global state ψ of G is an *observed global state* under experiment $e = \langle x_1, ..., x_p \ , \neg y_1, ..., \neg y_q \rangle$ if it satisfies all gene regulation rules for invariant nodes. Several observed global states are possible for the same set of invariant nodes.

4. The observed global state ψ of G is a *native global state* when no perturbations are made $(e = \langle \rangle)$.

We shall now prove upper and lower bounds for the number of experiments required for identifying a gene regulatory network with n genes, depending on the in-degree constraint and acyclicity. Table 9.1 summarizes the results. Computationally the running time of all algorithms when the in-degree is bounded is polynomial.

9.2.3 Upper and lower bounds on the number of experiments

We first show that an exponential number of experiments are required in the worst case.

Proposition 9.1 $\Theta(2^{n-1})$ *experiments must be performed in order to identify a gene regulatory network in the worst case.*

Constraints	**Lower bounds**	**Upper bounds**
None	$\Omega(2^{n-1})$	$O(2^{n-1})$
In-degree $\leq D$	$\Omega(n^D)$	$O(n^{2D})$
In-degree $\leq D$ All genes are AND-nodes (OR-nodes)	$\Omega(n^D)$	$O(n^{D+1})$
In-degree $\leq D$ Acyclic	$\Omega(n^D)$	$O(n^D)$
In-degree ≤ 2 All genes are AND-nodes (OR-nodes). No inactivation edges.	$\Omega(n^2)$	$O(n^2)$

Table 9.1: Source: [3]. Bounds on the number of experiments needed for reconstruction (n - number of genes, D - maximum in-degree). As seen from the table, forcing more constrains on the possible network topologies can improve experimental complexity significantly. The cases of acyclic topologies and restricted monotone logic (AND/OR gates only) are simpler mathematically but have no biological motivation.

Proof: Consider a boolean function of $(n-1)$ variables $f(x_1, x_2, .., x_{n-1})$ which is assigned to the node x_n. There are 2^{n-1} possible inputs, therefore there would be $2^{2^{n-1}}$ possible sets of outputs or $2^{2^{n-1}}$ boolean functions of $(n-1)$ variables. Hence we can identify this function by examining $log(2^{2^{n-1}})$ or 2^{n-1} assignments and less examinations will not suffice (we get one output bit per experiment). ■

Proposition 9.2 *$n2^{n-1}$ experiments always suffice in order to identify a gene regulatory network.*

Proof: For each node 2^{n-1} experiments are sufficient to identify its Boolean function by Proposition 9.1. Hence $n2^{n-1}$ experiments suffice in order to identify the whole network. ■

Theorem 9.3 *An exponential number of experiments are necessary and sufficient for the identification of a gene regulatory network.*

9.2.4 Bounded in-degree case with bounded cost

Since an exponential lower bound was proved in the general case, we consider a special but practical case, in which the maximal in-degree is bounded by a constant D. First, we consider the case $D = 2$.

Proposition 9.4 $\Omega(n^2)$ *experiments are necessary for identification even if the maximum in-degree is 2 and all nodes are AND nodes, where we assume that the maximum cost is bounded by a fixed constant C (number of genes among the N input genes that are perturbated).*

Proof: First, consider the case of $C = 2$. Assume that $\neg x \wedge \neg y \rightarrow z$ is assigned to z and all other nodes have in-degree 0. Among all experiments only $(\neg x, \neg y)$ can activate z. Therefore, we must test $\Omega(n^2)$ pairs of nodes in order to find (x, y).

Next, we consider the case of $C = 3$ with the same function $\neg x \wedge \neg y \rightarrow z$. If we disrupt or overexpress u, v, w such that $x \notin \{u, v, w\}$ or $y \notin \{u, v, w\}$, we can only learn that $(u, v), (u, w), (v, w)$ are different from (x, y). Since there are $\Theta(n^3)$ triplets and only $\Theta(n)$ triplets can include $\{x, y\}$, $\Theta(n^2)$ triplets must be examined in the worst case (each experiment removes at most a constant number of pairs out of the $\Theta(n^2)$ possible ones).

For cases of $C > 3$, similar arguments work: suppose $C = k > 3$, if we disrupt and/or everexpress $u_1, ..., u_k$ such that $x \notin \{u_1, ..., u_k\}$ or $y \notin \{u_1, ..., u_k\}$, we can only know that $\frac{k!}{2! \cdot (k-2)!}$ pairs are different from (x, y). Since there are $\Theta(n^k)$ k-mers and only $\Theta(n^{k-2})$ k-mers can include $\{x, y\}$, $\Theta(n^2)$ triplets must be examined in the worst case (each experiment removes at most a constant number of pairs out of the $\Theta(n^2)$ possible ones). ■

If C is not bounded, the above proposition does not hold. It is possible to identify the above pair (x, y) by $O(\log(n))$ experiments of maximum cost n, using a strategy based on binary search. Although this strategy may be generalized for other cases, we do not investigate it because experiments with high cost are not realistic. (The cells simply die if they are heavily mutated.)

Next, we consider the upper bound.

Proposition 9.5 $O(n^4)$ *experiments with maximum cost 4 are sufficient for identification if the maximum in-degree is 2 (No limitations on boolean function).*

Proof: We assume (w.l.o.g.) that all nodes are of in-degree 2 since identification of nodes of in-degree 1 or 0 is easier. Let c be any node of V. We examine all assignments to all quadruplets $\{a, b, x, y\}$ with $c \notin \{a, b, x, y\}$. The boolean function $g(a, b)$ is assigned to c (i.e., $f_c \equiv g$) if and only if $c \equiv g(a, b)$ for any assignment to $\{a, b, x, y\}$, where $c \equiv g(a, b)$ means that the *state* of c equals to $g(a, b)$. The 'only if' part is trivial. We shall prove the 'if' part. Suppose that $g(a, b)$ is not assigned to c, i.e., $f_c = h(a, b)$ and $h(a, b) \neq g(a, b)$. Clearly, $c \equiv g(a, b)$ does not hold. Next, consider the case where $h(p, q)$ is assigned to c where h may be equal to g and $\{p, q\} \cap \{a, b\} = \emptyset$. In this case, c takes both 1 and 0 by changing assignments to $\{p, q\}$ even if the assignment to $\{a, b\}$ is fixed. Therefore, $c \equiv g(a, b)$ does not hold. In the case remaining $\{p, q\} \cap \{a, b\} \neq \emptyset$. Suppose $f_c \equiv h(p, b)$ and $a \neq p$. Then there is a value of b so that $h(0, b) \neq h(1, b)$, but then $f_c(a, b, p = 0, y) \neq f_c(a, b, p = 1, y)$ and $c \equiv g(a, b)$ does not hold again. Since all assignments to all quadruplets are examined, in total $\frac{n!}{4! \cdot (n-4)!} * 2^4$ or $O(n^4)$ experiments are sufficient. ■

The above property holds even for an *unstable* graph because c is consistent under any experiment on $\{a, b, x, y\}$ if $f_c \equiv g(a, b)$.

Theorem 9.6 *$O(n^{2D})$ experiments with maximal cost $2D$ are sufficient for the identification of a gene regulatory network of bounded in-degree D. On the other hand, $\Omega(n^D)$ experiments are necessarily in the worst case if the cost of each experiment is bounded by a constant.*

9.2.5 Generalizations

Following generalizations can be made for the results descried above, for cases when all in-degrees $\leq D$.

- $\Omega(n^D)$ experiments of maximum cost lower than constant C are necessary for reconstruction of a network with in-degrees $\leq D$.

- $O(n^{2D})$ experiments with maximal cost of $2D$ suffice.

9.2.6 Efficient strategies for special cases

In this section we consider the case where the network consists of AND and/or OR nodes. In this case we assume that any AND (resp. OR) node c is *inactive* (resp. *active*) if at least one literal appearing in the boolean function assigned to c is forced to 0 (resp. 1) by disruption or overexpression of the gene corresponding to the literal. The above assumption is biologically reasonable even when the network contains inconsistent nodes.

Theorem 9.7 *A gene regulatory network which consists of AND and/or OR nodes and has maximum in-degree D can be identified by $O(n^{D+1})$ experiments.*

Proof: Here we only show strategy for a network that consists of AND nodes of in-degree 2. It can be generalized though, to the other cases. We examine all assignments to all triplets $\{a, b, x\}$ with $c \notin \{a, b, x\}$. The function $g(a, b)$ is assigned to c (i.e., $f_c = g$) if and only if $c \equiv g(a, b)$ for any assignment to $\{a, b, x\}$. Following the proof in Proposition 9.5, we only have to consider the case that $h(p, q)$ is assigned to c and $\{p, q\} \cap \{a, b\} = \emptyset$. Consider an assignment to $\{a, b, p\}$ for which $g(a, b) = 1$. If c is not *active* we can conclude that $c \equiv g(a, b)$ does not hold. If c is *active*, we can inactivate c by changing the assignment to p since only one assignment to $\{p, q\}$ can activate c. Thus , $c \equiv g(a, b)$ does not hold. Therefore, the above property holds and $O(n^3)$ experiments are sufficient in total. ■

Next, we consider the acyclic case for which we obtain an optimal bound.

Definition A set of nodes $\{x_1, x_2, ..., x_k\}$ has *influence* on y if there exist two experiments e_1 and e_2 on $\{x_1, x_2, ..., x_k\}$ such that e_1 activates y and e_2 inactivates y.

Definition A set of nodes $\{x_1, x_2, ..., x_k\}$ has *influence* on $\{y_1, y_2, ..., y_p\}$ if $\{x_1, x_2, ..., x_k\}$ has influence on at least one of $\{y_1, y_2, ..., y_p\}$.

Definition A set of nodes $\{x_1, x_2, ..., x_k\}$ has strong *influence* on y if there exist two experiments e_1 and e_2 on $\{x_1, x_2, ..., x_k\}$ such that e_1 activates y and e_2 inactivates y, and e_1 differs from e_2 only on a single x_i.

The above definitions are invalid if the network is unstable (i.e., has an inconsistent node) or has multiple stable states. Henceforth , we assume that the network cannot have inconsistent nodes except ones that are disrupted or overexpressed. Moreover, for stable networks, we make a biologically reasonable assumption that a set of nodes $\{x_1, x_2, ..., x_k\}$ does not have influence on a node to which there is no direct path from any of $\{x_1, x_2, ..., x_k\}$.

Theorem 9.8 *An acyclic gene regulatory network of maximum in-degree D can be identified by $\Theta(n^D)$ experiments.*

Proof: The lower bound directly follows from Proposition 9.4 and Theorem 9.6. We prove the upper bound only for $D = 2$. Other cases can be proved in similar way. Moreover, we only show the strategy for a node with $a \wedge b \rightarrow c$ although it can be generalized to other types of nodes. We assume (w.l.o.g.) that all nodes are of in-degree 2 as in Proposition 9.5. Let P be a set of pairs (x, y) satisfying the following conditions: c is *active* under $\langle x, y\rangle$, and c is *inactive* under the other assignments to (x, y). Then $a \wedge b \rightarrow c$ if and only if $(a, b) \in P$ and (a, b) does not have influence on any other pair $(x, y) \in P$. If $a \wedge b \rightarrow c$, then $(a, b) \in P$ must hold. Moreover, (a, b) does not have influence on any other pair in P since the network is acyclic. Conversely, if $a \wedge b \rightarrow c$ does not hold, then $(a, b) \notin P$ or (a, b) has influence on at least one node x, such that there is an edge from x to c. Therefore, we can identify the network by $O(n^2)$ experiments with maximum cost 2. ■

For cyclic networks with maximum in-degree D, $O(n^2)$ experiments of cost D do not suffice. It is possible to identify such network in some cases in $O(n^D)$ experiments. The strategy is based on detection of strongly connected components.

9.2.7 Related problems: Consistency and stability of networks

Along with the identification of the gene regulatory network, there exist several important problems. Here we observe two of them.

1. The consistency problem: given a network $G'(V', F')$, check whether or not this network coincides with an underlying gene regulatory network $G(V, F)$, that is not given explicitly.

   Theorem 9.9 *Exponential number of experiments are necessary and sufficient to check the consistency of a given gene regulatory network.*

2. The stability problem: given a network $G(V, F)$, check whether or not it is stable (in a native state), i.e., there is a global state consistent with all gene regulation rules. In other words we would like to find out whether there is a attractor of size 1.

   Theorem 9.10 *Testing the stability of a given gene regulatory network under an experiment is* NP-complete.

In work done by Akutsu *et al.* [3], it was shown that assuming $O(2^{2k}(2k + \alpha)logn)$ expression patterns (input/output pairs) drawn uniformly randomly are given, then with probability $>$ $1 - (1/n^{\alpha})$ there exists ≤ 1 Boolean network of n nodes with max indegree $\leq k$ consistent with patterns. But, note that uniform random sampling is very improbable - a living cell usually expresses very few of the patterns consistent with the network.

9.3 Kaufman's model

9.3.1 A Physicist's approach

Kaufman's model [7, 8] presents a physical approach, in which the aim is to understand general properties and characteristics of large networks.

We can view an organism as a very large genetic network. If we knew all the interactions of such a network, we could perfectly understand every single detail in the organism. That is, we could understand which genes, proteins and other molecules are involved in every biological process, how exactly the process takes place, etc.

This might be the ultimate goal of the biological science, but obviously we are years away from it. We therefore make a simplifying assumption. We model the organism as many distinct genetic networks, which loosely interact among themselves.

Indeed, this is a heavy assumption, but it is necessary in order for genetic networks to be useful in modeling biological processes.

Instead of looking at a specific network, we look at general properties of "network of the kind" (eg., networks where each component has exactly 2 related components). Given such a group of genetic networks, we can explore their properties (global structural features, types of possible dynamic behaviors, etc.). The search for generic properties may also provide hints for the analysis of specific circuits (like which features to expect, what questions to ask, etc.).

Definition An *ensemble of genetic networks* is composed of similar networks that share some features. The non constrained features vary at random between networks in the ensemble.

Properties of an ensemble of networks:

- Every network consists of N nodes (genes).
- Each gene is influenced directly by exactly k other input genes.
- For each node, the k input genes are chosen at random.
- For each node, its boolean function is chosen at random from the 2^{2^k} possible functions (the table of the input has size of 2^k states, and for each state the function can return 0 or 1).

9.3.2 Simplified description

Following are a few assumptions taken in order to simplify the model:

- The activation of genes depends on proteins and chemicals.
- The synthesis of proteins participating in a regulatory process is very fast compared to the regulatory process itself.
- Regulatory proteins decay much faster than the duration of the regulatory processes.

- The concentrations of the regulatory chemicals are constant.

As a result of those assumptions, we can express the activation level (mRNA level or protein level) in time $t + \delta t$ as a function of the activation at time t. We will later use $\delta t = 1$. This means that loss of memory occurs within δt time, that is, knowledge of steps before time T is not needed.

9.3.3 Generic questions

After sampling a number of network from the same ensemble, we can look for dynamic behavior in that certain type of networks such as fixed points, limit cycles; islands of activation spreading through the network. We can check how sensitive are the asymptotic states to perturbations of inputs / network structure. We can also ask questions such as: what kind of topology shall we expect; how does information flow from one point to the rest of the network (how far, how fast).

9.3.4 Kaufman's model

Kauffman's model uses boolean gene levels, 1 for active and 0 for inactive. It also assumes that time $t + 1$ is determined by a boolean function of the levels of a fixed set of input genes at time t. This means it can use only 1-step memory. All updates are executed in a deterministic way and are synchronized. The module assumes we have N nodes. We choose random topology between the nodes, than we choose random functions betweens the genes that effect a gene ("regulators") and the gene itself (the "regulatee"), and than we choose random initial values for the nodes at time 0.

Kauffman's model is dynamic:

- At time 0, a level is given to every gene.
- At each time step $t = 1, 2...$ every gene has a level $x_i(t)$, which is determined according to the boolean functions.
- The global state of the system is $X = [x_1, x_2, ...x_n]$ and we say that $X(t)$ alone determines $X(t + 1)$. As time passes, the system moves from state $X(t)$ to $X(t + 1)$, $X(t + 2)$ and so on, following a trajectory.

The states can be thought of as corners in the unit hypercube and a step from one global state to another can be thought of as shifting from one corner to another. Note, that a legal move does not have to be between two adjacent corners, since adjacent corners differ only by one bit. See 9.13 a 3-dimensional cube.

Example

Figure 9.14 shows basin of attraction of 12-gene boolean genetic network model - each node is a vector of 12 bits of 0/1. After a finite number of steps, an attractor of size six is reached.

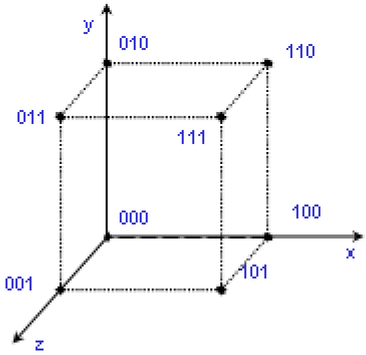


Figure 9.13: The state space of 3 states.

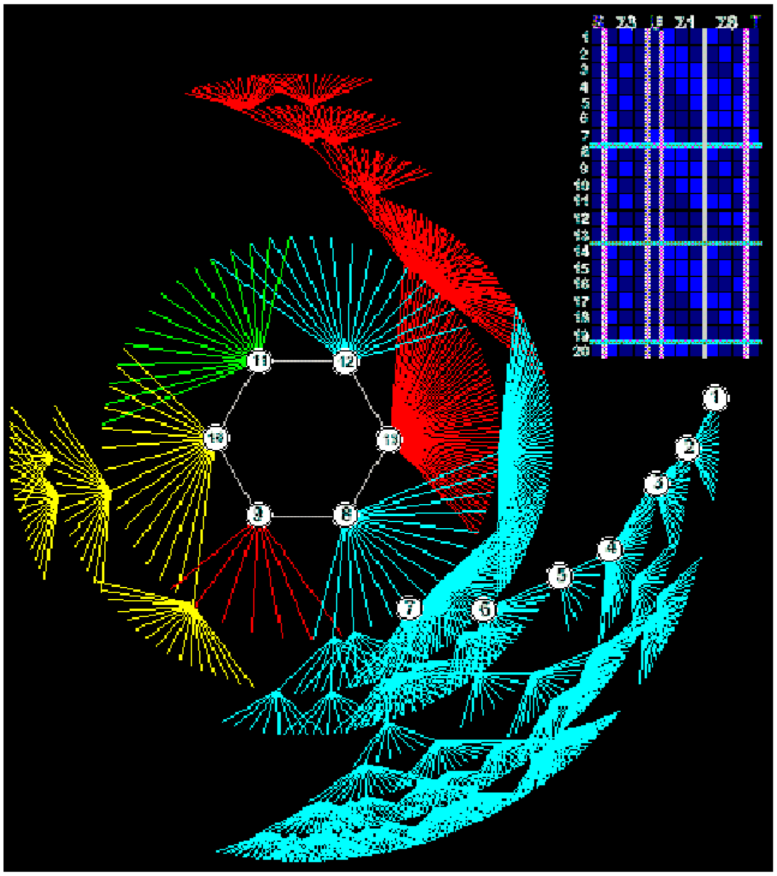


Figure 9.14: source: [9].Basin of attraction of 12- gene Boolean genetic network model.

9.3.5 Terminology

The terminology used for boolean network model is also relevant for Kaufman's model. *Trajectory* is a sequence consecutive states. *Attractor* is a set of states that are either approached when t goes to infinity, or reached in a finite time and no longer abandoned. Here, as well, the set of initial conditions that evolve towards a given attractor is its *basin of attraction.*

9.3.6 Features of Kaufman's model

Each network has it's own dynamics. An ensemble of networks can be analyzed by various means, such as starting from random initial conditions of a network and varying the network connections and rules. The main features of the model, attractors and basins, are determined by the degree of connectivity in each network. A degree of connectivity k means that the in-degree of each node is exactly k. See Figure 9.15.

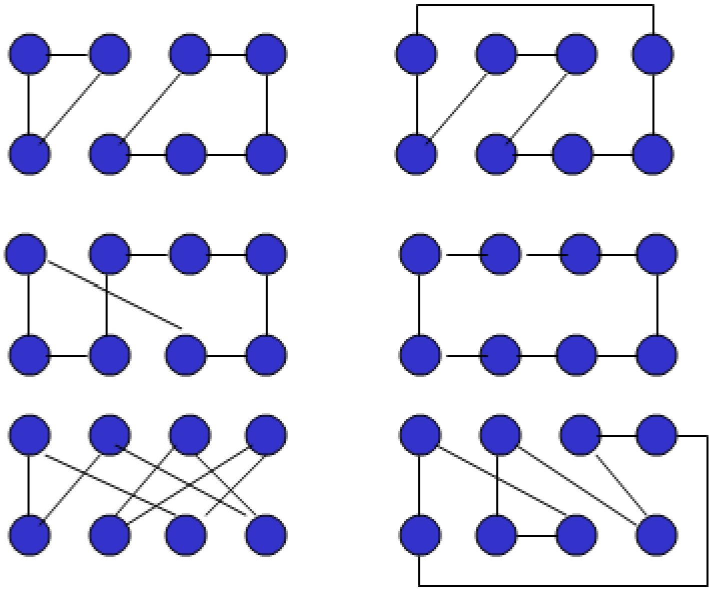


Figure 9.15: An ensemble of random networks with ($k = 2$). Note that every node in every network has degree 2.

High in-degree

In the case that k is as high as $N - 1$:

- $X(t+1)$ is completely uncorrelated to $X(t)$, the output associated to each input set is random. There is no correlation between outputs corresponding to two inputs which differ even by a single bit. The system is chaotic and the homeostatic stability is very low, nearby initial states go to different attractors, and changing one input function completely destroys the basin structure.

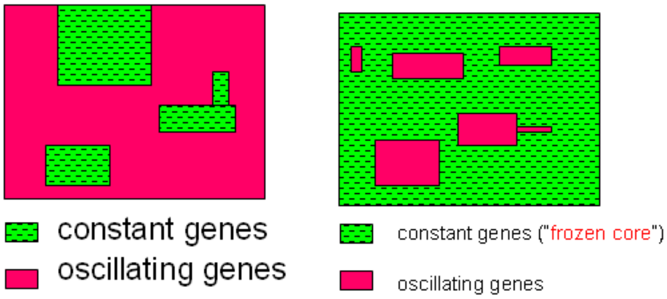


Figure a Figure b

Figure 9.16: Figure a: A 2-dimensional lattice view of a generic network, i.e., every cell in the lattice represents a gene. It can be seen, that when the in-degree is high, most of the genes are oscillating, that is, their state changes very frequently, and only few genes reach a constant state. Furthermore, the oscillating genes form a giant component, instead of being scattered all over the lattice; Figure b: A 2-dimensional lattice view of a generic network with low in-degree. One can see that the effect is opposite to that observed in figure a - most of the genes are constant, forming a giant component, while only few genes oscillate.

- The number of attractors, about N/e, is very small compared to the 2^N possible states.
- The cycles are huge, period size is around $2^{0.5N}$.

For example, for $N = 100,000$ we get $10^{30,000}$ states, only 37,000 attractors and cycles are as long as $10^{15,000}$.

One can notice that this kind of a network is not appropriate for analyzing biological systems behavior, since those tend to be stable and not sensitive to slight changes. A genetic network with high in-degree can be seen in Figure 9.16a.

Low in-degree

In the case of $k = 2$:

- Basins are regular: nearby initial states usually reach the same attractor, high homeostatic stability, spontaneous order, even though inputs and functions are completely random.
- The number of attractors is relatively high - about $N^{1/2}$.
- Average cycle length is $N^{1/2}$.

See Figure 9.16b.

For example, for $N = 100,000$ we get $10^{30,000}$ states, but only 317 attractors. Unlike the previous model, this kind of networks is consistent with experimental observations over many different phyla. Number of different cycles in a network represents the number of existing cell types. In addition, the length of a cycle represents the life period of a cell.

Phase transition

For a k-input boolean function, define $P = \max\{\text{no. 1-outputs, no. 0-outputs}\}/2^k$. It's obvious that $0.5 \leq P \leq 1$. For $P \approx 0.5$, the function is chaotic. For $P \approx 1$, the function is almost constant. The phase transition for different values of k is controlled by changing P, for example, by using canalizing functions, a boolean function where there is at least one value of one of the inputs that uniquely determines the output, irrespective of the others (eg. AND, OR).

9.3.7 Concluding remarks about Kauffman's model

A possible explanation of the model

The model is consistent with experimental observations over many different phyla. A ratio that was observed is that the number of cell types is approximately the number of different cycles which is approximately square root of the number of genes. A possible explanation is that a different cell start position will develop different types of cell. Another ratio that was observed is the length of cell life is approximately the length of the cycle in the graph. See Figure 9.17

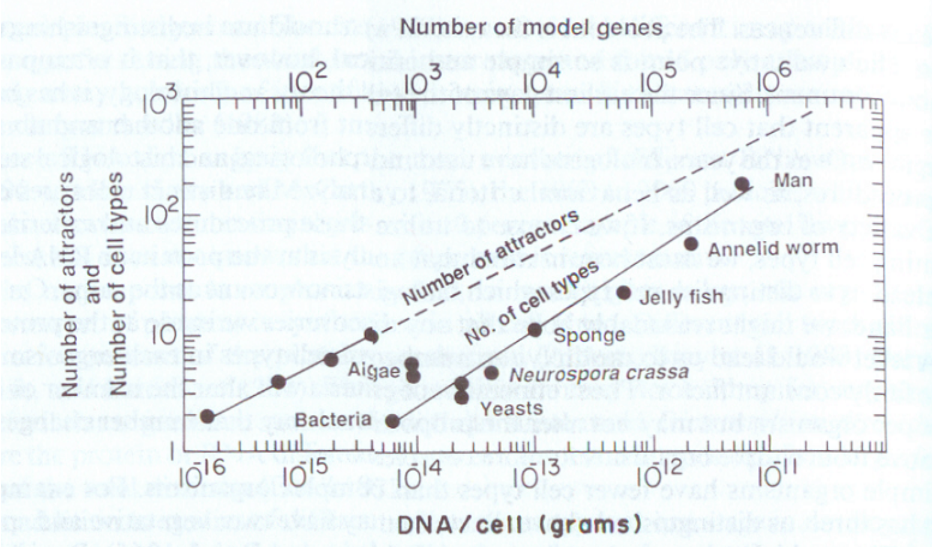


Figure 9.17: Logarithm of the number of cell types in organisms across many phyla plotted against the logarithm of the DNA content per cell. Plot is linear with a slope of 0.5, indicating a power-law relation in which the number of cell types increases as the square-root of the amount of DNA per cell. If total number of structural and regulatory genes is assumed proportional to DNA content, then the number of cell types increases as a square-root function of the number of genes. Number of attractors refers to predictions of numbers of model cell types in model genomic regulatory systems having k=2 input per gene.

Summary

Kauffman's model is a highly idealized representation of real genetic networks, due to the following reasons:

- The relation between genes are discrete (boolean) rather than continuous.
- The network status at time $t + 1$ depends only on its status at time t.
- Chemicals are not taken into account.
- Regulatory proteins are assumed to be synthesized very fast with respect to the regulation process itself.
- Synchronous activation may introduce "spurious cycles" in boolean dynamical systems.
- Fixed in-degree k is assumed for all genes.

However, Kauffman's model allows us to address issues which would otherwise be neglected, and to develop an appropriate language in which we can formulate key questions, such as:

- The importance of attractors in determining the properties of genetic networks.
- Robustness and basins of attraction.
- The importance of the average degree of connectivity.

Kauffman's model also allows us to examine in a new way the interplay between selection and self-organization. Moreover, it demonstrates the importance of studying ensembles of networks to gain insight about their generic properties.

Bibliography

[1] http://www.uib.no/aasland/two-hybrid.html.

[2] http://www.its.caltech.edu/ mirsky/endomeso.htm.

[3] T. Akutsu, S. Kuhara, O. Maruyama, and S. Miyano. Identification of gene regulatory networks by strategic gene disruptions and gene overexpressions. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 695–702, San Francisco, California, 25–27 January 1998.

[4] P. D'haeseleer and S. Fuhrman. Gene network inference using a linear, additive regulation model. *Bioinformatics*, 2000.

[5] P. D'haeseleer, X. Wen, S. Fuhrman, and R. Somogyi. Linear modeling of mRNA expression levels during CNS development and injury. In *Pacific Symposium on Biocomputing*, pages 41–52, Hawaii, Hawaii, 1999. World Scientific Publishing Co.

[6] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using bayesian networks to analyze expression data. *J. Computational Biology*, 7(3):601–620, Nov 1998.

[7] S.A. Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press, 1993.

[8] S.A. Kauffman. *At Home in the Universe: The Search for the Laws of Self-Organization and Complexity.* Oxford University Press, 1995.

[9] R. Somogyi and C. Sniegoski. *Complexity*, 1:45–63, 1996.