Identification of genetic networks by strategic gene disruptions and gene overexpressions under a boolean model

Tatsuya Akutsu\textsuperscript{a}, Satoru Kuhara\textsuperscript{b}, Osamu Maruyama\textsuperscript{c}, Satoru Miyano\textsuperscript{a,∗}

\textsuperscript{a}Human Genome Center, Institute of Medical Science, University of Tokyo, 4-6-1 Shirokanedai, Minato-ku, Tokyo 108-8639, Japan
\textsuperscript{b}Graduate School of Genetic Resources Technology, Kyushu University, Hakozaki 6-10-1, Higashi-ku, Fukuoka 812-8581, Japan
\textsuperscript{c}Graduate School of Mathematics, Kyushu University, Hakozaki 6-10-1, Higashi-ku, Fukuoka 812-8581, Japan

Abstract

Analysis of the interactions between genes by systematic gene disruptions and gene overexpressions is an important topic in molecular biology. This paper analyses the problem of identifying a genetic network from the data obtained by multiple gene disruptions and overexpressions in regard to the number of experiments and the complexity of experiments. An experiment consists of simultaneous gene disruptions and overexpressions and the complexity of an experiment is the number of genes disrupted or overexpressed. We define a genetic network as a boolean network and show a series of algorithms which describe methods for identifying the underlying genetic network by such experiments. Some lower bounds on the number of experiments required for the identification are also proved for some cases.
© 2002 Elsevier Science B.V. All rights reserved.

Keywords: Genetic network; Gene expression; Computational biology; Boolean network
1. Introduction

The whole DNA sequences of many organisms have been determined by international collaborations. Thus, the main topic of researches on microorganisms has shifted to the systematic functional analysis of the genes. Due to the recent progress of the DNA microarray technology, it has become possible (to some extent) to measure the gene expression levels of most of the genes of a microorganism simultaneously [6]. It has also become possible (to some extent) to disrupt genes and to overexpress genes [6]. Since genes are related to one another, expression pattern of genes will change by gene disruptions and/or gene overexpressions. Therefore, many attempts are being done in order to identify genetic networks [15] (networks of interactions of genes) by observing changes of gene expression patterns. However, the cost (both money and labour) for a gene disruption or a gene overexpression is high. The cost for measuring a gene expression pattern is also high. Therefore, it is important to develop an efficient strategy in order to identify a genetic network using gene disruptions and gene overexpressions.

The main contribution of this paper is a series of algorithms and analyses on strategies for identification of genetic networks. Our main interest is the number of experiments required for determining a network by multiple gene disruptions and gene overexpressions, where we assume that all genes are in one of the two states: being expressed or not being expressed (i.e., we assume a boolean network model). This paper shows upper bounds and lower bounds on the numbers of experiments.

Since analysis of gene expression patterns is a very important topic in molecular biology, a lot of methods have been proposed for the analysis. Various clustering algorithms were applied to classification of genes from time series data of gene expressions, and functions of some genes whose functions had not been known were predicted [6,18,19]. However, information obtained by clustering analyses is limited. Thus, many computational methods have been proposed for inferring genetic networks from gene expression data, based on various mathematical models of networks. For example, inference algorithms were proposed using a standard boolean network model [14], an edge-labelled directed graph [5], a model based on linear differential equations [8], a model based on non-linear differential equations [16,22], neural network like models [21], and the Bayesian network model [10].

Among such studies, the REVEAL (REVerse Engineering ALgorithm) algorithm proposed by Liang et al. [14] has some similarity with our algorithms. REVEAL is also based on a boolean model of a genetic network. However, it should be noted that our work was done independently, and both a conference paper by Liang et al. [14] and a preliminary version of this paper [1] appeared in January, 1998. After these two papers, many algorithms were proposed for identifying genetic networks [5,8,10,13,16,22]. The difference between this paper and the paper by Liang et al. is not small. Liang et al. [17] employed the standard boolean network in which expression levels of genes change synchronously as in digital circuits, whereas this paper employs a static boolean network model in which expression levels of genes that are statically determined can only be observed. It seems that our static model is more natural than the standard synchronous boolean network model because expression levels of real genes do not change synchronously. Liang et al. made no theoretical analysis on the number
of experiments, whereas the main results of this paper concern theoretical analyses of
the number of experiments. Moreover, techniques used in this paper can be applied
to the analysis of the standard boolean network. It should be noted that we gave a
theoretical proof in [2] for an observation derived from computational experiments on
REVEAL [14] that a large boolean network could be identified from a small number
of expression patterns (O(log n) patterns) in most cases. Proposition 4 of this paper
played an important role in that proof.

It might be more acceptable if we formulate genetic networks using real valued func-
tions [8,15,20] since real biological networks are not discrete. However, measurement
effects of expression levels by DNA microarrays are not yet small and are typically on
the order of 10%. More crucial difficulty in using real valued functions is that there
are no established mathematical models which can describe most gene regulation rules.
Of course, several mathematical models such as Michaelis–Menten equation were ap-
plied to the analysis of metabolic pathways. But, it is unknown whether or not such
models are adequate for describing gene regulation rules. Therefore, we give up the
use of real valued functions and define a genetic network as a boolean network. Not
a few discussions have been made on the appropriateness of the use of boolean net-
works as the model of genetic networks [7,12]. In these discussions, several examples
are shown in which genes behave like ON–OFF switches. It is also shown that many
interesting behaviours corresponding to cell differentiation, proliferation and apoptosis
emerge from boolean networks. Nevertheless, the gaps between boolean networks and
real genetic networks are not small and thus the results of this paper are not directly
applicable to the identification of real genetic networks. However, the methodologies
in this paper might be applied to other network models and might be useful for the
analysis of real genetic networks in the future. Possible extensions are discussed in the
final section.

Before making formal definitions, we will explain our boolean network model using
an example. We assume that all genes are in one of the two states: being expressed
or not being expressed. Fig. 1 shows an example of a genetic network with 16 genes.
Genes A, C, I, K, N, X1, X2 are expressed under no condition. Arrows with ⊕ and ⊖
mean activation and repression, respectively. Genes B, E, H, J, M are expressed if their
direct predecessors are not expressed. For gene D, it is expressed if its all predecessors
C, F, X1, X2 are expressed. The same holds for genes L and G. Gene F is activated
by gene A and is also repressed by gene L. F is expressed if A is expressed and L
is not expressed. On the other hand, F is not expressed if A is not expressed or L is
expressed. Three cases of gene expressions (normal, disruption of A, overexpression
of gene B) are shown in Table 1, where 1 (0) means that the gene is expressed (is
not expressed). However, the disruption of gene K yields the activation of L which
represses F while gene A is expressed and activates F. Such conflict may occur in the
network. In such case the gene expression may be ambiguous or may change from time
to time. To be explained later, our network model can treat such case in a reasonable
way.

We define the indegree of a gene by the number of genes directly affecting it in a
genetic network. The indegree of most genes will be small except some special genes
if we ignore weak interactions. Therefore, it has an important sense in practice to cope
with genetic networks with a small indegree constraint when the genes are restricted to a specific region. In particular, algorithms for a small indegree constraint are useful for testing whether a known genetic network of another organism is realized in the target organism.

We consider a genetic network model such that the expression of a gene is determined by a boolean function of the expressions of the genes directly affecting it. We prove the upper and lower bounds of experiments required for identifying a genetic network with \( n \) genes in regard to the indegree constraint and acyclicity. Table 2 summarizes the results. Computationally, all algorithms for the results with polynomial experiments in Table 2 run in polynomial time. Additional results on a heuristic algorithm, testing the consistency of a given network, and testing the stability of a given network are also presented to provide some insights into the problem.
Table 2
Summary of results on identification, where $n$ denotes the number of genes

<table>
<thead>
<tr>
<th>Constraints</th>
<th>Number of experiments</th>
</tr>
</thead>
<tbody>
<tr>
<td>No constraint</td>
<td>$\Omega(2^{n-1})$</td>
</tr>
<tr>
<td>Indegree $\leq D$ &amp; all genes are AND-nodes (OR-nodes)</td>
<td>$O(n^{2D})$</td>
</tr>
<tr>
<td>Indegree $\leq D$ &amp; acyclic</td>
<td>$O(n^D)$</td>
</tr>
<tr>
<td>Indegree $\leq 2$ &amp; all genes are AND-nodes (OR-nodes) &amp; no repression edges &amp; strongly connected</td>
<td>$O(n^2)$</td>
</tr>
</tbody>
</table>

In the literature [4,11], the problems of identifying various boolean functions from examples have been extensively investigated. The problem in this paper looks similar to these problems. But it differs from these approaches at the point that we deal with network structures augmented with biological meanings. As mentioned before, various algorithms have been proposed for identification of genetic networks. But a preliminary version of this paper is one of the pioneering papers [1,14]. Moreover, to our knowledge, there are few papers which study identification strategies [13].

2. A boolean model of a genetic network

We define a genetic network in a rather general way. A genetic network $G = (V,F)$ is a boolean network with the set $V$ of nodes and the set $F = \{ f_v \mid v \in V \}$ of boolean functions assigned to the nodes, where this network may have cycles. We confuse a genetic network $G = (V,F)$ with its underlying directed graph $G(V,E)$. We call a node of $G$ a gene. The boolean function assigned to the node represents the condition for the gene to be expressed and is a gene regulation rule. We assume that a genetic network satisfies the following conditions: When a boolean function $f_v$ assigned to $v$ has $k$ inputs, $k$ input lines (directed edges) come from $k$ distinct nodes $u_1, \ldots, u_k$ other than $v$ (denoted by $f_v(u_1, \ldots, u_k) \to v$). Moreover, $f_v$ has the property that for each $i = 1, \ldots, k$ there exists an input $(a_1, \ldots, a_k) \in \{0,1\}^k$ with $f_v(a_1, \ldots, a_i, \ldots, a_k) \neq f_v(a_1, \ldots, \neg a_i, \ldots, a_k)$, where $\neg a_i$ is the negation of $a_i$. A node $v$ with no input has a constant value (1 or 0) and we denote it by $1 \to v$ ($0 \to v$). Unless otherwise stated, we assume that the value of a node with no input is 1 although all the results are valid in a general case. We say that a gene $v$ is expressed (is not expressed) if the value of $v$ is 1 (0). The value of $v$ is also called the state of $v$. If the value $f_v(u_1, \ldots, u_k)$ of node $v$ is determined by the formula $l(u_1) \land (u_2) \land \cdots \land (u_k)$ $(l(u_1) \lor (u_2) \lor \cdots \lor (u_k))$, we call $v$ an AND node (OR node), where $l(u_i)$ is either $u_i$ or $\neg u_i$. We call an edge $(u_i,v)$ an activation edge (repression edge) if $l(u_i)$ is a positive literal (negative literal).

For a gene $v$, gene overexpression of $v$ forces $v$ to be expressed and gene disruption of $v$ forces $v$ not to be expressed. Let $x_1, \ldots, x_p$, $y_1, \ldots, y_q$ be mutually distinct
genes of $G$. An experiment with gene overexpressions $x_1, \ldots, x_p$ and gene disruptions $y_1, \ldots, y_q$ is denoted by $e = \langle x_1, \ldots, x_p, \neg y_1, \ldots, \neg y_q \rangle$. The cost of $e$ is defined by the number $p + q$.

A global state of $G$ is a mapping $\psi : V \to \{0, 1\}$. Each global state must satisfy $\psi(x_i) = 1$ and $\psi(y_j) = 0$ under an experiment $\langle x_1, \ldots, x_p, \neg y_1, \ldots, \neg y_q \rangle$. The global state of the genes need not be consistent with the gene regulation rules. We say that a global state $\psi$ of $G$ is stable under an experiment $\langle x_1, \ldots, x_p, \neg y_1, \ldots, \neg y_q \rangle$ if it is consistent with all gene regulation rules except those assigned to nodes $x_1, \ldots, x_p, y_1, \ldots, y_q$, i.e., for each node $v \in \{x_1, \ldots, x_p, y_1, \ldots, y_q\}$ with inputs from $u_1, \ldots, u_k$, $\psi(v) = f_v(\psi(u_1), \ldots, \psi(u_k))$. Otherwise, it is called unstable. We say that a genetic network $G$ is stable under an experiment $e$ if there is a global state of $G$ which is stable under $e$. When no experiment is made on $G$, we simply remove “under $e$” from the terminology.

Note that the stable global state is not necessarily unique. For example, if $V = \{v_1, v_2\}$ with $v_1 \to v_2$ and $v_2 \to v_1$, both of the global states $[v_1 = 0, v_2 = 0]$ and $[v_1 = 1, v_2 = 1]$ are stable. Moreover, the genetic network becomes unstable under some experiment. For example, if $V = \{v_1, v_2, v_3\}$ with $1 \to v_1$, $\neg v_1 \wedge v_3 \to v_2$ and $\neg v_2 \to v_3$, there is no stable global state $\psi$ satisfying $\psi(v_1) = 0$. As a result of an experiment, however, all the states of the genes can be observed. Hence we should define an observed global state.

Before defining an observed global state, we define the set of invariant nodes $I$ under an experiment $e = \langle x_1, \ldots, x_p, \neg y_1, \ldots, \neg y_q \rangle$, along with a mapping $\phi$ from $I$ to $\{0, 1\}$. Invariant nodes are defined inductively by the following rules:

1. If $v$ appears in $e$ or has no incoming edge (i.e., indegree $= 0$), $v$ is an invariant node. Moreover, $\phi(x_i) = 1$ for $x_1, \ldots, x_p$, $\phi(y_j) = 0$ for $y_1, \ldots, y_q$, and $\phi(v) = f_v$ for the other nodes $v$.

2. Let $U$ be the set of incoming nodes to $v$ and let $U' = \{u_1, \ldots, u_k\} \subseteq U$ be the current set of invariant nodes in $U$, where $\phi$ is defined for any node in $U'$. If $f_v$ with inputs $\phi(u_1), \ldots, \phi(u_k)$ is invariant for any states of nodes in $U - U'$, then $v$ is invariant and we define $\phi(v)$ by the value of $f_v$.

Although there is an ambiguity in the order of selecting $v$ in step (2), it is not difficult to see that the set of invariant nodes is determined uniquely independent of the order of selecting nodes. Therefore, the above definition is sound. Once invariant nodes are defined, an observed global state $\psi$ under an experiment $e$ is an arbitrary global state such that $\psi(v) = \phi(v)$ for all $v \in I$. A native global state of $G$ is an observed global state when no experiment is made on $G$ (i.e., $p = q = 0$). We say that an experiment $e$ activates (represses) $v$ if $v$ is observed to be expressed (not to be expressed) under the experiment $e$. Note that if $v$ is not an invariant node under the experiment, the state of $v$ may not be determined uniquely.

This paper investigates the number of experiments together with the cost of each experiment for identifying a genetic network from the observed global states. Note that disruption or overexpression of a gene is not an easy task and may take several days or more. Therefore, the number of experiments is a very important factor in practice. The cost of an experiment is also a crucial matter in the experiment since many changes in the genes may cause the death of organisms.
3. Upper and lower bounds of the number of experiments

3.1. General bounds

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

**Proposition 1.** At least $2^{n-1} - 1$ experiments must be done in order to identify a genetic network in the worst case.

**Proof.** If any boolean function can be assigned to a node, the result directly follows from the well-known result on identification of boolean functions [4]. Using a similar technique as in [4], we can show in this case that the theorem holds even if a very simple boolean function can only be assigned to a node.

We consider a network such that $V = \{x_1, \ldots, x_p, y_1, \ldots, y_q, z\}$ ($n = p + q + 1$), all nodes but $z$ are of indegree 0, and the following function is assigned to $z$:

$$x_1 \land x_2 \land \cdots \land x_p \land \neg y_1 \land \neg y_2 \land \cdots \land \neg y_q \rightarrow z.$$  

For a set of nodes $\{x_1, \ldots, x_p, y_1, \ldots, y_q\}$, there are $2^{p+q}$ possible assignments, among which only one assignment (i.e., experiment) $e$ can activate $z$ (note that even if only a subset of genes are disrupted and/or overexpressed, this corresponds to some assignment to all nodes). Therefore, from any assignment but $e$, we can know only that this assignment does not activate $z$. Therefore, $2^{p+q} - 1 = 2^{n-1} - 1$ assignments must be tested in the worst case ($2^{n-2} - 1$ assignments even in the average case). 

Note that we can identify an AND node (resp. an OR node) by $O(n)$ experiments with maximum cost $n - 1$ if this node is expressed (resp. is not expressed) in a native state. Assume that $x_i \land \cdots \land x_k \rightarrow x_n$ is assigned to node $x_n$, where $V = \{x_1, \ldots, x_n\}$. For each $i < n$, we examine an experiment $e_j = (x_1, x_2, \ldots, x_{j-1}, \neg x_j, x_{j+1}, \ldots, x_{n-1})$. Then, $\{x_i, \ldots, x_k\} = \{x_j \mid x_n \text{ is not expressed under } e_j\}$ holds and we can identify an AND node $x_n$.

In a general case, we can identify a boolean function assigned to node $x_n$ and edges incoming to $x_n$ by examining $2^{n-1}$ assignments to $\{x_1, \ldots, x_{n-1}\}$. Therefore, we can identify the genetic network by $n2^{n-1}$ experiments.

**Theorem 2.** Exponential number of experiments are necessary and sufficient for the identification of a genetic network.

3.2. Bounded indegree case

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

Proposition 3. Ω(n^2) experiments are necessary for the identification even if the maximum indegree is 2 and all nodes are AND nodes, where we assume that the maximum cost is bounded by a fixed constant C.

Proof. First we consider a case of C = 2. Assume that \( \neg a \land \neg b \rightarrow c \) is assigned to c and the other nodes are of indegree 0. Among all experiments, only \( \langle \neg a, \neg b \rangle \) can activate c. Therefore, we must test \( \Omega(n^2) \) pairs in order to find a pair \((a, b)\) from a similar argument as in Proposition 1.

Next we consider a case of C = 3. If we disrupt and/or overexpress \( x, y, z \) such that \( a \not\in \{x, y, z\} \) or \( b \not\in \{x, y, z\} \), we can only know that three pairs \((x, y), (y, z), (x, z)\) are different from \((a, b)\). Since there are \( \Theta(n^3) \) triplets and only \( \Theta(n) \) triplets can include \{a, b\}, at least \( \Omega(n^2) \) triplet must be examined in the worst case.

For cases of \( C > 3 \), we can use a similar argument.

Note that the above proposition does not necessarily hold if the maximum cost is not bounded by a constant. Indeed, we can identify the above pair \((a, b)\) by \( O(\log n) \) experiments of maximum cost \( n \), using a strategy based on binary search. Here we briefly describe the strategy. We assume w.l.o.g. (without loss of generality) that \( V = \{x_1, x_2, \ldots, x_k, c\} \). First, we make two experiments:

\[ e_1 = \langle \neg x_1, \ldots, \neg x_k, x_{k+1}, x_{k+2}, \ldots, x_{2k} \rangle, \]
\[ e_2 = \langle x_1, x_2, \ldots, x_k, \neg x_{k+1}, \neg x_{k+2}, \ldots, \neg x_{2k} \rangle. \]

If c is expressed under \( e_1 \), \( \{x_1, x_2, \ldots, x_k\} \) contains both a and b. If c is expressed under \( e_2 \), \( \{x_{k+1}, x_{k+2}, \ldots, x_{2k}\} \) contains both a and b. In each case, a and b can be searched recursively. If c is not expressed under \( e_1 \) or \( e_2 \), \( \{x_1, x_2, \ldots, x_k\} \) contains a and \( \{x_{k+1}, x_{k+2}, \ldots, x_{2k}\} \) contains b (or vice versa). In this case, a and b can be searched independently and recursively. For example, in order to find a, we begin with the following two experiments:

\[ e_3 = \langle \neg x_1, \ldots, \neg x_{\lceil k/2 \rceil}, x_{\lceil k/2 \rceil+1}, \ldots, x_k, \neg x_{k+1}, \ldots, \neg x_{2k} \rangle, \]
\[ e_4 = \langle x_1, \ldots, x_{\lceil k/2 \rceil}, \neg x_{\lceil k/2 \rceil+1}, \ldots, \neg x_k, \neg x_{k+1}, \ldots, \neg x_{2k} \rangle. \]

Although this strategy might be generalized for some other cases, we do not investigate it because experiments with high costs are not realistic.

Next, we show an upper bound.

Proposition 4. \( O(n^4) \) experiments with maximum cost 4 are sufficient for the identification if the maximum indegree is 2.

Proof. We assume w.l.o.g. that all nodes are of indegree 2 because identification of nodes of indegree 1 and 0 is easier.

Let c be any node in \( V \). We examine all assignments (i.e., experiments) to all quadruplets \{a, b, x, y\} with \( c \not\in \{a, b, x, y\} \). Then a boolean function \( g(a, b) \) is assigned to c (i.e., \( f_c = g \)) if and only if there exists a boolean function \( g(a, b) \) such that
c = g(a, b) for any assignment to \{a, b, x, y\}, where c = g(a, b) means that the state of c is equal to g(a, b).

Here, we prove the above property. Since “only if” part is almost trivial, we prove “if” part (see Fig. 2). Suppose that g(a, b) is not assigned to c, but h(p, q) is assigned to c. First, we consider a case of \{p, q\} = \{a, b\}. In this case, clearly c = g(a, b) does not hold. Next, we consider a case of \{p, q\} \cap \{a, b\} = \emptyset. In this case, c takes both 1 and 0 by changing assignment to \{p, q\} even if assignment to \{a, b\} is fixed. Therefore, c = g(a, b) does not hold. Similarly, we can prove that c = g(a, b) does not hold in a case of |\{p, q\} \cap \{a, b\}| = 1.

Note that the above property holds even for an unstable network because c is consistent under any experiment on \{a, b, x, y\} if g(a, b) \rightarrow c.

Since all assignments to all quadruplets are examined in total, O(n^4) experiments are sufficient.

It is straightforward to generalize the above discussion to a case of any fixed D.

**Theorem 5.** O(n^{2D}) experiments with maximum cost 2D are sufficient for the identification of a genetic network of bounded indegree D. On the other hand, \Omega(n^D) experiments are necessary in the worst case if cost of each experiment is bounded by a constant.

It should be noted that boolean functions assigned to nodes of indegree \leq D can be identified correctly by using the strategy for bounded indegree D even if the maximum indegree of the network exceeds D.

4. More efficient strategies for special cases

Although we have shown an O(n^4) upper bound for D = 2, we can obtain better upper bounds for some special cases. First, we consider a case that the network consists of
AND and/or OR nodes. It should be noted that any AND node \( c \) is not expressed if at least one literal appearing in the boolean function assigned to \( c \) is forced to be 0 by disruption or overexpression of the gene corresponding to the literal.

**Theorem 6.** A genetic network which consists of AND and/or OR nodes of maximum indegree \( D \) can be identified by \( O(n^{D+1}) \) experiments.

**Proof.** Here we only show a strategy for a network that consists of AND nodes of indegree 2. But, it can be generalized to the other cases.

We examine all assignments to all triplets \( (a,b,x) \) such that \( c \in \{a,b,x\} \). Then \( g(a,b) \) is assigned to \( c \) (i.e., \( f_c = g \)) if and only if there exists a boolean function \( g(a,b) \) such that \( c = g(a,b) \) holds for any assignment to \( \{a,b,x\} \).

Since the other cases of this property can be proved as in Proposition 4, we consider a case that \( h(p,q) \) is assigned to \( c \) where \( \{p,q\} \cap \{a,b\} = \emptyset \). Consider an assignment to \( \{a,b,p\} \) for which \( g(a,b) = 1 \). If \( c \) is not expressed, we can conclude that \( c = g(a,b) \) does not hold. If \( c \) is expressed, we can repress \( c \) by changing an assignment to \( p \) because only one assignment to \( \{p,q\} \) can activate \( c \), and thus \( c = g(a,b) \) does not hold.

Therefore, the above property holds and \( O(n^3) \) experiments are sufficient in total.

Next, we consider an acyclic case for which we can obtain an optimal bound (ignoring a constant factor).

We say that a set of nodes \( \{x_1,\ldots,x_k\} \) has influence on \( y \) if there exist two experiments \( e_1 \) and \( e_2 \) to \( \{x_1,\ldots,x_k\} \) such that \( e_1 \) activates \( y \) and \( e_2 \) represses \( y \). We also say that \( \{x_1,\ldots,x_k\} \) has influence on \( \{y_1,\ldots,y_h\} \) if \( \{x_1,\ldots,x_k\} \) has influence on at least one \( y_i \). If \( y \) is not an invariant node, \( y \) may or may not be influenced. However, we need not care whether \( y \) is influenced or not in the following argument.

**Theorem 7.** An acyclic genetic network of maximum indegree \( D \) can be identified by \( \Theta(n^D) \) experiments.

**Proof.** The lower bound directly follows from Proposition 3 and Theorem 5. Here, we prove the upper bound only for \( D = 2 \), where those for the other cases can be proved in a similar way. Moreover, we only show a strategy for a node with \( a \land b \rightarrow c \) since it can be generalized for the other types of nodes. Note that in an acyclic network, states of all nodes are determined uniquely (i.e., all nodes are invariant nodes) under any experiment.

We assume w.l.o.g. that all nodes are of indegree 2 as in Proposition 4. Let \( P \) be a set of pairs \( (x,y) \) satisfying the following conditions: \( c \) is expressed under \( (x,y) \), and \( c \) is not expressed under the other assignments to \( (x,y) \).

Then, \( a \land b \rightarrow c \) if and only if \( (a,b) \in P \) and \( (a,b) \) does not have influence on any other pair \( (x,y) \in P \). This property is proved in the following way. If \( a \land b \rightarrow c \), \( (a,b) \in P \) must hold. Moreover, \( (a,b) \) does not have influence on any other pair in \( P \) since the network is acyclic. Conversely, if \( a \land b \rightarrow c \) does not hold, then \( (a,b) \notin P \) or \( (a,b) \) has influence on at least one input node \( x \) to \( c \).
Therefore, we can identify the network by $O(n^2)$ experiments with maximum cost 2.

In the above case, we only use experiments with maximum cost 2. But, there is a case in which an experiment with cost 3 is necessary if the network has cycles.

**Proposition 8.** There exist genetic networks of bounded indegree 2 such that experiments with cost 3 are required for the identification.

**Proof.** We consider a network such that $V = \{v_0, \ldots, v_{n-1}\}$ and $v_i \land v_{i+1} \rightarrow v_{i+2}$ is assigned to each node $v_{i+2}$ where indices are computed under modulo $n$. Note that there is a trivial (cyclic) ordering on vertices (i.e., $v_0 \succ v_1 \succ \cdots \succ v_{n-1} \succ v_0 \cdots$).

In this case, for each pair $(v_i, v_j)$, $\langle v_i, v_j \rangle$ may activate all the nodes (although there exist multiple stable global states), while $\langle v_i, \neg v_j \rangle$ and $\langle \neg v_i, \neg v_j \rangle$ always repress all the nodes (except $v_i$). Therefore, we cannot identify the ordering of nodes by experiments with cost at most 2.

Although we have proved that an experiment with cost 3 is necessary if a graph has cycles, we can still develop a strategy with $O(n^2)$ experiments for a special case, which includes the above example.

**Theorem 9.** $\Theta(n^2)$ experiments with maximum cost 3 are sufficient for the identification of a genetic network such that all nodes are AND nodes of indegree at most 2, its underlying graph $G(V,E)$ is strongly connected and all edges are activation edges.

**Proof.** First, we show the upper bound. From the condition of the theorem, it is sufficient to identify input nodes to each node. We only consider the nodes of indegree 2 because identification for the nodes of indegree $\leq 1$ can be done using $O(n^2)$ experiments with cost 2 from Theorem 5.

Let $p$ be an arbitrary node in $V$. For each node $c \neq p$, we identify input nodes $a, b$ to $c$. In a case of $p \notin \{a, b, c\}$, we use the following strategy. We examine assignments $\langle x, y, \neg p \rangle$ for all unordered pairs $(x, y)$ such that $\{x, y\} \cap \{p, c\} = \emptyset$. Let

$$P(x, y) = \{z \mid z \notin \{x, y\} \text{ is expressed under } \langle x, y, \neg p \rangle\}.$$  

Then, the following property holds: $a \land b \rightarrow c$ if and only if $c \in P(a, b)$ and $(\forall (x, y) \neq (a, b))(c \in P(x, y)) \Rightarrow (x \notin P(a, b) \lor y \notin P(a, b))$.

Here, we only prove only if part of this property because if part can be proved in a similar way. Since $a \land b \rightarrow c$ holds, $c \in P(a, b)$ must hold. We assume that there exists another pair $(x, y)$ such that $c \in P(x, y)$. Since $G(V,E)$ is strongly connected, there must exist a path from $x$ to $a$ and a path from $p$ to $x$. Then, it is sufficient to consider the following two cases (see Fig. 3):

(i) There exists a simple path from $p$ to $x$ not including $a$ or $b$.
(ii) Every simple path from $p$ to $x$ includes $a$ or $b$.

Then, $x \notin P(a, b)$ must hold in case (i), and $c \notin P(x, y)$ must hold in case (ii). Therefore, $(\forall (x, y) \neq (a, b))(c \in P(x, y)) \Rightarrow (x \notin P(a, b) \lor y \notin P(a, b))$ must hold.
In a case of $p \in \{a, b\}$, $P(x, y) = \emptyset$ must hold for all $x, y$. Then, we can see that either $a = p$ or $b = p$ holds and we can apply a strategy similar to that for nodes of indegree 1. In a case of $p = c$, we can use another node $p'$. It is easy to see that $O(n^2)$ experiments with cost at most 3 are used in total.

Next, we show the lower bound. We use a similar argument as in Proposition 3 though more complicated construction is required.

We assume w.l.o.g. $n = 2^k$ and construct a genetic network with $3n + 1$ nodes as follows (see Fig. 4):

$$V = \{x_0, c\} \cup \{x_1, \ldots, x_n\} \cup \{x_1^0, \ldots, x_n^0\}$$

$$\cup \{x_1^1, \ldots, x_n^{1/2}\} \cup \{x_1^2, \ldots, x_n^{n/4}\} \cup \cdots \cup \{x_1^{k-1}, x_2^{k-1}\} \cup \{x_1^k\},$$

$$F = \{x_0 \land x_i^0 \rightarrow x_i | i = 1, \ldots, n\} \cup \{x_i \rightarrow x_i^0 | i = 1, \ldots, n\}$$

$$\cup \{x_1^{k-1} \land x_2^{k-1} \rightarrow x_{j+1}^h | h = 1, \ldots, k, j = 0, \ldots, 2^{k-h} - 1\}$$

$$\cup \{x_1^0 \land c \rightarrow x_0\} \cup \{x_a \land x_b \rightarrow c\}.$$
In order to activate $c$, $x_a$ and $x_b$ must be activated. In order to activate $x_a$ and $x_b$, either $x_a$ or $x_a^0$ must be overexpressed and either $x_b$ or $x_b^0$ must be overexpressed. Otherwise, there can be a global state which represses $c$. Therefore, as in Proposition 3, $\Omega(n^2)$ experiments are necessary if the maximum cost is bounded by a constant.

Note that the above proof can be modified for a case in which AND is replaced by OR. Although Theorem 9 might be generalized for any constant $D$, we do not investigate it because the genetic network consisting of only activation edges is not realistic.

5. A heuristic strategy

We have shown an $\Omega(n^2)$ lower bound on the number of experiments even for the case of bounded indegree 2. Since $O(n^2)$ experiments are almost impossible even for Yeast (Saccharomyces cerevisiae, $n > 6000$), we cannot expect any single strategy identification method for the genetic network and that the methods in Section 4 should be employed only for determining local network structures. This observation leads us to develop a strategy by which we can identify as many parts as possible using $O(n)$ experiments. In such a case, we should find a set of edges $E'$ such that $E' \subseteq E$. That is, we should find a set of edges not including false positive edges.

Let $J(x)$ denote the set of nodes that are influenced by $\{x\}$, where we let $x \notin J(x)$. That is, $J(x)$ is the set of nodes that take different states under $\langle x \rangle$ and $\langle \neg x \rangle$. Although $J(x)$ is not necessarily determined uniquely, we assume that we can observe a set $J(x)$ satisfying the following conditions: (i) If $y \in J(x)$ and $x \notin U$ ($U$ is the set of incoming nodes to $y$), there is at least one node $z \in U$ such that $z \in J(x)$. (ii) Each node $x$ does not have influence on any node to which there is no directed path from $x$. Since $J(x)$ can be obtained from two experiments $\langle x \rangle$ and $\langle \neg x \rangle$, $O(n)$ experiments are sufficient in order to obtain $J(x)$ for all nodes.

**Proposition 10.** If $J(b) \cup \{b\} = J(a)$ holds and there is no cycle including node $b$, edge $(a, b)$ appears in the genetic network.

**Proof.** Suppose that both conditions hold but edge $(a, b)$ does not appear in the network. Since $b \in J(a)$ holds from $J(b) \cup \{b\} = J(a)$, there must exist a simple path from $a$ to $b$ which includes at least another node $x$ such that $x \in J(a)$. Since there is no cycle including $b$ and there is a path from $x$ to $b$, $x \notin J(b)$ holds, which is a contradiction. 

Note that the condition that there is no cycle including $b$ that cannot be removed from the above proposition. For example, $J(b) \cup \{b\} = J(a)$ holds in both networks in Fig. 5. But, in each case, there does not necessarily exist an edge $(a, b)$. Note that three nodes satisfy $J(b) \cup \{b\} = J(a)$ in case (i), while only one node satisfies this condition in case (ii). Although testing the existence of a cycle may require exponential number of experiments as in Proposition 1, it is expected that such cases as in Fig. 5 (ii)
seldom occur. Therefore, if only one node $b$ satisfies $J(b) \cup \{b\} = J(a)$ for fixed $a$, we may predict that edge $(a, b)$ appears in the network though this relation should be confirmed by further biological experiments. It should be noted that Proposition 10 does not use properties of boolean functions but uses conditions (i) and (ii) only. Therefore, Proposition 10 can be applied to other models satisfying these conditions.

6. Consistency and stability of networks

Along with the identification of the genetic networks, there exist several important problems. Here we consider two of them.

6.1. Consistency

The consistency problem is, given a network $G'(V', F')$, to check whether or not this network coincides with the underlying genetic network $G(V, F)$ (note that $G(V, F)$ is not given explicitly).

**Proposition 11.** Exponential number of experiments are necessary and sufficient for checking the consistency of a given genetic network.

**Proof.** Since the sufficiency directly follows from Theorem 2, we consider the necessity.

Let $G(V, F)$ be the same network as in the proof of Proposition 1. Let $G'(V', F')$ be a network having no edges. Since we do not know a topology of the underlying network $G(V, F)$, $2^{n-1} - 1$ experiments are required in order to check whether or not the underlying network coincides with $G'(V, F')$.  

Using a similar discussion, $O(n^{2D})$ upper bound and $\Omega(n^{D})$ lower bound can also be obtained for bounded indegree case.

6.2. Stability

Testing the stability of a given network is also an important computational problem. Recall that $G$ is stable (in a native state) if there exists a global state consistent with all gene regulation rules (i.e., boolean functions). However, this problem is hard in general.
Theorem 12. Testing the stability of a given genetic network is NP-complete.

Proof. Once a global state is assigned, the consistency of all regulation rules can be tested in polynomial time. Therefore, the problem is in NP.

In order to show NP-hardness, we transform 3SAT to the problem. Let \( U = \{u_1, u_2, \ldots, u_n\} \) be the set of variables and \( C = \{c_1, c_2, \ldots, c_m\} \) be the set of clauses. We define a genetic network \( G = (V, F) \) (see Fig. 6). The construction will be made up of several components, which can be partitioned into two separate parts, grouped according to their intended function: “truth-setting” components and “satisfaction-testing” components.

For each variable \( u_i \in U \), there is a truth-setting component \( T_i = (V_i, F_i) \), with \( V_i = \{u_i, w_i\} \) and \( F_i = \{\neg w_i \rightarrow u_i, \neg u_i \rightarrow w_i\} \), that is, each of the two elements, if it is expressed, will repress the other. Since \( w_i \) corresponds to a literal \( \neg u_i \), we identify \( w_i \) with \( \neg u_i \).

Once truth-setting components are defined, satisfaction-testing components can be defined in a straightforward way using similar components. For each clause \( c_j \in C \), let \( x_j, y_j \) and \( z_j \) be the literals in \( c_j \). Then, a satisfaction-testing component \( (V'_j, F'_j) \) is defined by \( V'_j = \{a_j, b_j, c_j, x_j, y_j, z_j\} \) and

\[
F'_j = \{x_j \lor y_j \lor z_j \rightarrow c_j, \neg c_j \land b_j \rightarrow a_j, \neg a_j \rightarrow b_j\}.
\]

Finally, \( G(V, F) \) is defined by \( V = (\bigcup_{j=1}^{m} V_j) \cup (\bigcup_{j=1}^{m} V'_j) \) and \( F = (\bigcup_{j=1}^{m} F_j) \cup (\bigcup_{j=1}^{m} F'_j) \).

Note that any stable global state \( \psi \) of \( G \) must satisfy the condition that one of \( \psi(u_i) \) and \( \psi(w_i) \) is 1 and the other is 0. Note also that \( \psi \) must satisfy the condition that \( \psi(c_j) \) is 1, otherwise either \( a_j \) or \( b_j \) must be inconsistent. Therefore, at least one of \( x_j, y_j \) and \( z_j \) is expressed in any stable state.

It is clear from the construction that \( G = (V, F) \) is stable if and only if \( C \) is satisfiable (see Fig. 6), and this construction can be done in polynomial time.
7. Concluding remarks

Algorithmic strategies for identification of genetic networks using gene disruptions and gene overexpressions are studied in this paper. Although a simple boolean network model is used here, real regulation mechanisms of genes are much more complex. Complex models might be good for simulation of biological behaviours. But it would be quite difficult to identify genetic networks if a complex model were used. Therefore, simple models should be used so that efficient identification strategies and efficient identification algorithms can be developed.

Since the boolean network model might be too simple, it is valuable to consider extensions of the boolean network model. It is natural to try to extend the binary domain (\(\{0,1\}\)) to other discrete domains (e.g., \(\{0,1,\ldots,N-1\}\)). Extension of the results of Theorems 2 and 5 for such domains is straightforward under an assumption that expression level of each gene can be set to an arbitrary value in \(\{0,1,\ldots,N-1\}\).

Another possible extension is a probabilistic version of the boolean network since there exist measurement errors of gene expression levels and biological behaviour might be determined non-deterministically. Based on this idea, we defined a probabilistic version of the standard synchronous boolean network and developed an identification algorithm [3]. In this network model, we allow that each boolean rule does not hold with probability less than a fixed constant \(P\) (i.e., the output of each boolean function can be inverted with low probability). In order to identify these networks, we modified the algorithm presented in [2], where this original version was developed for the identification of the standard synchronous boolean networks. This modified algorithm examines all possible boolean functions of indegree \(D\) and outputs boolean functions whose error rates are less than some threshold. We proved that the order of the sample complexity is the same as that for the noiseless synchronous boolean network (i.e., the sample complexity is \(O(\log n)\)) if \(P\) and \(D\) can be treated as constant numbers.

The Bayesian network model, which was already applied to inference of genetic networks [10], can also be considered as a probabilistic extension of the boolean network model. Since probabilistic models were successfully applied to biological sequence analysis [9], further studies on probabilistic models of genetic networks should be done.

Acknowledgements

We thank Masanori Arita, Satoshi Matsumoto and Ayumi Shinohara for valuable discussions. We also thank Ido Grabinsky for pointing out an error in the previous version of this paper [1].

References


