

On Observability of Steady States in a Boolean Network

Tatsuya Akutsu[†]Takeyuki Tamura[†]

1 Introduction

The *Boolean network* (BN) is known as a discrete mathematical model of gene regulatory networks [3]. In a BN, each node corresponds to a gene and takes one of two values 0 and 1, where 0 (resp., 1) means that the corresponding gene is inactive (resp., active). The value of a node at a given time step is determined according to a regulation rule, which is a Boolean function of the values of the predecessors of the node at the previous time instant. The values of nodes are updated synchronously, and the (global) *state* of a network at a given time step is the vector of its node values. Beginning from any initial state, the system eventually falls into an *attractor*, which is classified into two types: a *singleton attractor* corresponding to a stable state, and a *periodic attractor* corresponding to a sequence of states that repeats periodically. In some interpretation, attractors are considered as cell types: different attractors correspond to different cell types [3]. Based on this interpretation, extensive studies have been done on distribution of attractors and detection of attractors. Furthermore, extensive studies have recently been done on controllability and observability of BNs [2, 4].

By the way, it is also important for medical diagnosis to identify the type of each cell by observing expression patterns of a few genes (e.g., marker genes). Therefore, in this technical report, we consider the problem of identifying attractors by observing expression levels (0 or 1 in BN) of a small number of genes using the BN model. In particular, we focus on finding the minimum number of genes, by which all given attractors can be discriminated. Since treating periodic attractors is much more difficult, we only consider singleton attractors. Although it is #P-hard to enumerate all singleton attractors, we assume that the set of singleton attractors is given. We can even assume that this set is given independent of a BN because gene expression data for each cell type may be able to be experimentally obtained without knowing the structure of the underlying genetic network.

2 Problem Definition

Let A be an $m \times n$ binary matrix. $A[i, j]$ denotes the element at i -th row and j -th column. $A[i, -]$ and $A[-, j]$ denote the i -th row and j -th column of A , respectively. Let $J = \{j_1, \dots, j_k\}$ be a set of column indices. Then, A_J denotes a submatrix of A consisting of j_1, j_2, \dots, j_k -th columns.

Minimum Observer for Singleton Attractors [MOSA]

[†]Institute for Chemical Research, Kyoto University

Input: A set of singleton attractors represented as an $m \times n$ binary matrix A , where m and n correspond the number of singleton attractors and the number of genes, respectively.

Output: A minimum cardinality set J of columns (i.e., genes) such that $A_J[i_1, -] \neq A_J[i_2, -]$ holds for all $i_1 \neq i_2$.

For example, consider the following matrix A :

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Then, $J = \{2, 3, 5, 6\}$ is a solution of MOSA for A because

$$A_{\{2,3,5,6\}} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

It is to be noted that MOSA is a special case (i.e., binary case) of the minimum key problem [1, 5], and even the special case is known to NP-hard [7].

3 A Simple Dynamic Programming Algorithm

MOSA can be trivially solved in $O(2^n \text{poly}(m, n))$ time by examining all possible subsets of columns. However, n is usually large and m is usually small because n corresponds to the number of genes and m corresponds to the number of attractors (the number of different types of cells). Here we present a simple dynamic programming algorithm that works in $O(m^m \text{poly}(m, n))$ time.

Let \mathbf{s} be an m -dimensional vector of integers between 0 to $m-1$, which we call a *signature vector*. It is to be noted that there are m^m possible signature vectors. $\mathbf{0}$ denotes the signature vector consisting only of 0's. \mathbf{s}_i denotes the i -th element of \mathbf{s} .

If each element of \mathbf{s} is at most $2^k - 1$ for some integer k (i.e., each element of \mathbf{s} is represented using k bits), \mathbf{s} is called a *k-bit signature vector*. Then, we identify such \mathbf{s} with $m \times k$ binary matrix by regarding the i -th element of \mathbf{s} as the i -th row consisting of k bits. Let $M(\mathbf{s})$ denotes such a matrix. Conversely, we can construct a k -bit signature vector \mathbf{s} from a given

$m \times k$ binary matrix by identifying each row with k bit number. However, we use a compact form of \mathbf{s} by renaming the numbers appearing in \mathbf{s} (with keeping the ordering) so that only consecutive numbers beginning from 0 are used. Let the resulting vector be $\mathbf{v}(M)$. For $m \times k_1$ matrix M_1 and $m \times k_2$ matrix M_2 , $M_1 \cdot M_2$ denotes $m \times (k_1 + k_2)$ matrix obtained by concatenating M_1 and M_2 . For example, consider the matrix A given in Section 2. Then, for $\mathbf{s} = (1, 6, 6, 3, 2, 2)$, $M(\mathbf{s})$ is a matrix consisting of the first three columns of $A_{\{2,3,5,6\}}$, and $M(\mathbf{s}) \cdot A[-, 6] = A_{\{2,3,5,6\}}$. While signature vectors corresponding to $A_{\{2,3,5\}}$ and $A_{\{2,3,5,6\}}$ are $(1, 6, 6, 3, 2, 2)$ and $(3, 12, 13, 6, 5, 4)$ respectively, $\mathbf{v}(A_{\{2,3,5\}}) = (0, 3, 3, 2, 1, 1)$ and $\mathbf{v}(A_{\{2,3,5,6\}}) = (0, 5, 6, 3, 2, 1)$.

We define a binary table $D[\mathbf{s}, k]$ by: $D[\mathbf{s}, k] = 1$ if and only if there exists J with $|J| = k$ such that $\mathbf{v}(A_J) = \mathbf{s}$. $D[\mathbf{s}, k]$ can be computed by the following dynamic programming procedure. Although it returns only the minimum size of J , such J can be obtained by using the standard *traceback* procedure. All $D[\mathbf{s}, k]$ s are initialized to be 0.

```

Procedure SolveMOSA(A)
  D[0, 0] ← 1;
  for k = 1 to n do
    for all  $\mathbf{s}$  such that  $D[\mathbf{s}, k - 1] = 1$  do
      for all column  $j$  of  $A$  do
         $M' \leftarrow M(\mathbf{s}) \cdot A[-, j]$ ;
         $\mathbf{s}' \leftarrow \mathbf{v}(M')$ ;
         $D[\mathbf{s}', k] \leftarrow 1$ ;
        if  $\mathbf{s}'_{i_1} \neq \mathbf{s}'_{i_2}$  holds for all  $i_1 \neq i_2$  then
          return  $k$ 

```

Theorem 1 *MOSA can be solved in $O(m^m \text{poly}(m, n))$ time.*

Although the time complexity may be slightly improved by identifying identical signatures (e.g., identifying $(0, 1, 2, 2, 3)$ and $(2, 1, 3, 3, 0)$), significant improvement of the time complexity is left as future work.

4 Remark on Feedback Vertex Set

Mochizuki et al. showed that all (singleton and periodic) attractors can be identified by observing states of nodes in an FVS [6]. Here we give a very simple example (see also Fig. 1) showing that the minimum FVS does not necessarily give a solution of MOSA. Let $V = \{v_1, v_2, v_3\}$ be a set of nodes. We define a BN on these nodes by the following regulation rules

$$\begin{aligned} v_1(t+1) &= v_1(t) \vee \overline{v_3(t)} \\ v_2(t+1) &= v_2(t) \wedge v_3(t) \\ v_3(t+1) &= v_1(t) \oplus v_2(t) \end{aligned}$$

where $x \wedge y$, $x \vee y$, and $x \oplus y$ denote logical AND, OR, and XOR, respectively. Then, we can see that $(0, 1, 1)$

and $(1, 0, 1)$ are the singleton attractors (i.e., 0-1 assignments to nodes such that $v_i(t+1) = v_i(t)$ for all $i = 1, 2, 3$). These two attractors can be discriminated by observing the state of v_1 (or v_2). Therefore, the size of the minimum observer node set is 1. On the other hand, the minimum FVS is $\{v_1, v_2\}$. Therefore, we can see that the minimum FVS does not necessarily give a solution of MOSA. Although self loops are included in this example, we can modify it so that there does not exist self loops.

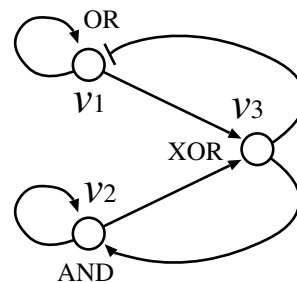


Figure 1: Example of BN showing that the minimum FVS does not necessarily give a solution of MOSA. In this figure, T-type arrow means that the input is negated.

References

- [1] T. Akutsu and F. Bao, Approximating minimum keys and optimal substructure screens, *Proc. COCOON 1996* (LNCS 1090), 290-299, 1996.
- [2] D. Cheng, H. Qi, and Z. Li, *Analysis and Control of Boolean Networks*, Springer, 2011.
- [3] S. A. Kauffman, *The Origins of Order: Self-organization and Selection in Evolution*, Oxford Univ. Press, 1993.
- [4] D. Laschov, M. Margaliot, and G. Even, Observability of Boolean networks: A graph-theoretic approach, *Automatica*, 49:2351-2362, 2013.
- [5] C. L. Licchesi and S. L. Osborn, Candidate keys for relations, *J. Computer and System Sciences*, 17:270-279, 1978.
- [6] A. Mochizuki, B. Fiedler, G. Kurosawa, and D. Saito, Dynamics and control at feedback vertex sets. II: A faithful monitor to determine the diversity of molecular activities in regulatory networks, *Journal of Theoretical Biology*, 335:130-146, 2013.
- [7] R. Motwani and Y. Xu, Efficient algorithms for masking and finding quasi-identifiers, Technical Report, Stanford University, 2007.