Exploring Observability of Attractor Cycles in Boolean Networks for Biomarker Detection

YUSHAN QIU1, YULONG HUANG2, SHAOBO TAN3, DONGQI LI3, ADA CHAEI VAN DER ZUJP-TAN2, GLEN M. BORCHERT4, HAO JIANG5, AND JINGSAN HUANG3

1College of Mathematics and Statistics, Shenzhen University, Shenzhen 518000, China
2College of Allied Health Professions, University of South Alabama, Mobile, AL 36688, USA
3School of Computing, University of South Alabama, Mobile, AL 36688, USA
4College of Medicine, University of South Alabama, Mobile, AL 36688, USA
5School of Mathematics, Renmin University of China, Beijing 100872, China

Corresponding authors: Hao Jiang (jiangh@ruc.edu.cn) and Jingshan Huang (huang@southalabama.edu)

This work was supported in part by the Natural Science Foundation of SZU under Grant 000393, in part by the Natural Science Foundation of Shenzhen under Grant JCYJ 20170817100950436, and in part by the National Natural Science Foundation of China NSFC under Grant 11901575, Grant 91730301, and Grant 61472428.

ABSTRACT Boolean Network (BN) is a simple and popular mathematical model that has attracted significant attention from systems biology due to its capacity to reveal genetic regulatory network behavior. In addition, observability, as an important network feature, plays a vital role in deciphering the underlying mechanisms driving a genetic regulatory network and has been widely investigated. Prior studies examined observability of BNs and other complex networks. That said, observability of attractor, which can serve as a biomarker for disease, has not been fully examined in the literature. In this study, we formulated a new definition for singleton or cyclic attractor observability in BNs and developed an effective methodology to resolve the captured problem. We also showed complexity is of \( O(P^m n) \), when the maximal period of cyclic attractor is \( P \), the number of attractor is \( m \) and the number of genes is \( n \). Importantly, we have confirmed our method can faithfully predict the expression pattern of segment polarity genes in Drosophila melanogaster and showed it can effectively and efficiently deal with the captured observability problem.

INDEX TERMS Attractor, Boolean networks, observability.
Even though the authors in [13] have reported that the attractor control problem is $\Sigma_2^P$-hard, an ILP-based methodology can be applied to moderate size BNs. Each of these studies, however, has simply examined randomly generated, simple BNs (i.e., one cell type). It is therefore critically necessary to develop strategies capable of analyzing multiple BNs (i.e., various cell types). In the genuine cellular context, there are various kinds of cells, and it is therefore simply more realistic to perform attractor analysis for multiple BNs. Motivated by this, Qiu et al. [14] recently examined this problem and employed the ILP-based method to solve it.

The control of Boolean networks is also a challenging problem. According to control theory, dynamic systems are controllable if they can be driven from any initial state to any desired final state with a suitable choice of inputs [15]. Several studies have been conducted on this problem [16], [17], and when a random Boolean network is examined, the principle interest lies on the stationary distribution of the system. The reachability problem as in control theory becomes a common concern for the deterministic network [18].

Observability, as a dual of controllability, is a significant concept which needs further exploration. Several studies have recently examined such observability problems. Cheng and Qi [19], [20] adopted a semi-tensor strategy to examine observability and related problems. While their design was effective for formulating and constructing the problems, the application of their method to large-scale BNs is also impractical. Of note, Liu et al. [21] proposed a novel view of complex networks observability. They showed the aim of attractors observability was to determine the minimum consecutive nodes necessary to discriminate distinct attractor cycles from each other in the system. This work provided a great opportunity for one to diagnose disease types since different attractors represent different cell types. That said, we suggest here that the problem proposed represents a particular case, binary alphabet of the minimal key problem examined in [12], [22], [23]. It has been showed that the minimal key problem is NP-hard even when dealing with the binary alphabet [23]. Cheng et al. [24] have proposed to apply the ILP-based method to solve the observability problem which was limited to singleton attractors. However, cyclic attractors exist in the real-world and the issue of observability for cyclic attractors has not yet been fully addressed. Qiu et al. [25] have proposed a method to study the observability of singleton and cyclic attractor. However, the analysis on the sensitivity of number of genes and attractor nodes need to be further explored. In addition, the application of our method to the real model requires further discussion. Thus, the problem we try to address in this study is to identify the minimum set of contiguous nodes such that we can discriminate the singleton or cyclic attractors. Furthermore, we analyze the distribution of gene number and attractor number for distinguishing the cell types, respectively. We show that our proposed method is efficient and can effectively solve large-size problems in $O(P^m n)$ for the worst case, where $P$ is the maximum number of period of cyclic attractor, $m$ is the number of (singleton or cyclic) attractors and $n$ is the number of genes (nodes) in a BN.

This paper has two main contributions. First, we formulate a new biological problem based on Boolean networks to discriminate the cell type with minimum consecutive nodes. Second, we propose an effective and efficient method, (works in $O(P^m n)$) to solve the captured problem. By integrating the expression pattern of the segment polarity genes in Drosophila melanogaster, we have demonstrated that our approach can efficiently identify the minimum contiguous genes or proteins to discriminate the attractors. Furthermore, experimentation confirms that our proposed methodology is extremely efficient and can effectively solve the problem in seconds.

The remainder of the paper is structured as follows. In the “Problem formulation” section, we introduce some background and formulate the problem. “Methodology” presents a novel method for our captured problem and give two propositions. “Numerical Experiments” describes the materials and experimentation results to validate the efficiency and effectiveness of our method. And finally, conclusions and future directions are described in the last section.

II. PROBLEM FORMULATION

This section (1) introduces the BN model and the attractor detection problem which are closely related to attractor observability, and (2) then formulates the attractor observability problem in BNs.

A BN $G(V, F)$ consists of a set of nodes $V$ used to represent multiple genes $V = \{v_1, v_2, \ldots, v_n\}$ and a set of Boolean functions $F = \{f_1, f_2, \ldots, f_n\}$ where $f_i : \{0, 1\}^n \rightarrow \{0, 1\}$. Each node (e.g., a gene) is assumed to take either 1 (active) or 0 (inactive) as its state value. And $v_i(t)$ is denoted as the state of node $i$ at time $t$, after which the state of node $i$ at time $t + 1$ is given as follow:

$$v_i(t + 1) = f_i(v(t)), \quad i = 1, 2, \ldots, n$$

where

$$v(t) = [v_1(t), \ldots, v_n(t)].$$

This indicates the gene state $v_i$ at time $t + 1$ is determined by the $h_i$ gene states at previous time $t$, where $h_i$ represents input node number called indegree of $v_i$. Moreover, the maximum indegree of BN is denoted by $H = \max_i(h_i)$. We note $x \lor y, x \land y, \neg x, x \oplus y$ is logical OR of $x$ and $y$, logical AND of $x$ and $y$, logical NOT of $x$ and exclusive OR of $x$ and $y$, respectively. Overall gene expressions at time $t$ in a BN is determined by

$$gap(t) = [v_1(t), v_2(t), \ldots, v_n(t)]$$

and called the network Gene Activity Profile (GAP) at time $t$. Note that gap ranges from [0, 0, \ldots, 0] to [1, 1, \ldots, 1], therefore we have a total of $2^n$ possible global states. The following details an example of a BN.
The truth table.

| State | \(v_1(t)\) | \(v_2(t)\) | \(v_3(t)\) | \(v_1(t+1)\) | \(v_2(t+1)\) | \(v_3(t+1)\) |
|-------|-----------|-----------|-----------|-------------|-------------|-------------|
| 1     | 0         | 0         | 0         | 1           | 1           | 0           |
| 2     | 0         | 0         | 1         | 1           | 1           | 0           |
| 3     | 0         | 1         | 0         | 0           | 1           | 0           |
| 4     | 0         | 1         | 1         | 0           | 1           | 0           |
| 5     | 1         | 0         | 0         | 1           | 0           | 1           |
| 6     | 1         | 0         | 1         | 1           | 1           | 1           |
| 7     | 1         | 1         | 0         | 0           | 0           | 1           |
| 8     | 1         | 1         | 1         | 0           | 1           | 1           |

Each gene \(v_i\) is regulated by a Boolean function \(f_i\). A state transition diagram and the dynamics of a BN are shown in Figure 1. The BN truth table is showed in Table 1. For example, the fourth row of the table shows that if the state of BN is \([0, 1, 1]\) at time \(t\), then the state at time \(t+1\) will be \([0, 1, 0]\). Similarly, the arc from \([0, 1, 1]\) to \([0, 1, 0]\) indicates that when the BN state is \([0, 1, 1]\) at time \(t\), the state is \([0, 1, 0]\) at time \(t+1\). In total, there are \(2^n\) potential states in a BN containing \(n\) nodes. Thus, the state transition table contains \(2^n\) rows, and the corresponding state transition diagram has \(2^n\) vertices.

The gene state is determined by its associated regulatory function. Given the initial state of a network, a BN will ultimately enter into a certain set of global states (i.e., a directed cycle as depicted in state transition diagram). We denote such a set as an attractor. When an attractor corresponds to one global state (i.e., a single fixed point), it is classified as a singleton attractor. In all other events, it would be classified as a cyclic attractor, and we denote \(p\) as the period of cyclic attractor if it consists of \(p\) global states. We can see from Figure 1 that the network will eventually evolve into one of two attractors: either a singleton attractor \([0, 1, 0]\) or a cyclic attractor of period 2, \([1, 1, 0]\) \(\rightarrow\) \([0, 0, 1]\).

### A. OBSERVABILITY OF ATTRACTORS

Once attractors are detected, one can conduct an analysis of the BN attractor control problem [13], [14]. Observability, as a compliment of controllability, has been examined by Cheng and Qi [19]. They developed necessary and sufficient conditions for dealing with the captured observability problem. Cheng et al. [24] proposed an efficient and effective approach for solving singleton attractor observability in BNs.

That said, the problem of observability for cyclic attractor has not yet been resolved. Thus, in this study, we propose a new methodology for addressing the observability problem of (singleton and cyclic) attractors, not restricted to singleton ones in BNs, and therefore representative of the realities of biology. Importantly, when a BN is given, attractors can be further detected. In brief, assume a set of singleton or cyclic attractors is given as \(S = \{S_1, S_2, \ldots, S_m\}\), our method then attempts to identify a tandem gene set of minimum cardinality that can be used to distinguish distinct attractor cycles from each other in a system. For instance, we consider four singleton or cyclic attractors with seven genes which is given as \(S = \{S_1, S_2, S_3, S_4\}\)

\[
S_1 = \{(0, 0, 1, 0, 1, 0, 1, 0)\};
S_2 = \{(1, 1, 0, 0, 0, 1, 0)\};
S_3 = \{(1, 1, 0, 0, 1, 0, 0)\};
S_4 = \{(0, 0, 1, 0, 0, 1, 1)\};
\]

where \(S_1\) and \(S_2\) are singleton attractors, \(S_3\) and \(S_4\) correspond to cyclic attractors of period 2. Note that there are four attractors in total, it would require at least two nodes to distinguish them. Next, if we observe the third and forth nodes only, we can identify which attractor a system belongs to (i.e., \((1, 1), (0, 0), (0, 1), (1, 0)\) mean \(S_1, S_2, S_3\) and \(S_4\) respectively). As such, the attractor observability problem is formulated as:

**Definition 1:** ATTRACTION OBSERVABILITY (AO)

Instance: List of (singleton or cyclic) attractors in a BN.

Problem: Determine the minimum cardinality of consecutive nodes that can distinguish different attractor cycles from one another in a BN.

### III. METHODOLOGY

In the following section, we describe our attractor observability problem approach.

### A. PROPOSED ALGORITHM FOR AO

Assume a set of (singleton or cyclic) attractors are given as \(S = \{S_1, S_2, \ldots, S_m\}\), where size of each \(S_i, i \in \{1, 2, \ldots, m\}\) is \(L_i\) and they have total of \(L = L_1 \cdot L_2 \cdots L_N\) possible combinations and employ a matrix set \(A = \{A_1, A_2, \ldots, A_L\}\) to describe all possible case combinations. Each matrix \(A_i, i \in \{1, 2, \ldots, L\}\) is therefore of size \(m \times n\) (\(m\): attractor number; \(n\): nodes (genes) number). Then an individual row \(A_i\) corresponds to one global state from singleton attractor or cyclic attractor and individual columns represent gene states. We then develop our algorithm based on matrix \(A\) for AO. The major steps (procedures) for our algorithm are details below.

**The Procedure**

Step I: For \(i \in \{1, 2, \ldots, L\}\), assume the size of matrix \(A_i\) is \(m \times n\), where \(m\) is attractor number and \(n\) is gene number. Repeat Step II-VIII.
Step II: Apply arithmetic \( \oplus \) operator for \( A_i \) to generate a new matrix \( biState_i \) which is of size \( C_m^2 \times n \). We then define
\[
biState(t, j) = A(i, j) \oplus A(k, j),
\]
\[
t = (m - \frac{i}{2})(i - 1) + k - i.
\]
\[
1 \leq i < k \leq m, \quad 1 \leq j \leq n, \quad i, j, k, t \in N.
\]
recall that the arithmetic \( \oplus \) is presented as follows:
\[
x \oplus y = (x \land \neg y) \lor (\neg x \land y).
\]
Step III: For an individual row \( t \in C_m^2 \) in matrix \( biState_i \), identify consecutive 0’s with length no less than \( \lceil \log_2(m) \rceil \) and place them in matrix \( B_{con} \) (assume that total entry number in \( B_{con} \) is \( r \)). Another matrix \( B_{index} \) size \( r \times 2 \) is then created to record indices of \( B_{con} \) (e.g., \( B_{index}(i, 1) \) and \( B_{index}(i, 2) \) correspond to the first and last elements of row \( i \) of \( B_{con} \), \( i \in \{1, \ldots, r\} \), respectively).
Step IV: Generate the union of \( B_{con} \) and denote it as \( B_{uni} \).
Step V: Let \( U \) represent a vector which ranges from 1 to \( n \) (i.e., \( U = \{1, \ldots, n\} \)). If \( U \setminus B_{uni} \neq \emptyset \) holds, then we can conclude this represents the desired minimal contiguous nodes and corresponding minimal number for \( A_i \) is \( \lceil \log_2(m) \rceil \) and denoted as \( minL_i \). Therefore, the desired minimal number of consecutive nodes has been identified (i.e., \( \lceil \log_2(m) \rceil \)) and stop. Otherwise, continue.
Step VI: Employ a sorting algorithm (such as bucket sort) to \( B_{index} \). Then rank vector \( B_{index}(:, 1) \) in an ascending order and update the order in \( B_{index}(:, 2) \) accordingly. As an example,
\[
B_{index} = \begin{bmatrix}
1 & 5 \\
4 & 7 \\
3 & 9 \\
3 & 6 \\
2 & 8
\end{bmatrix} \tag{1}
\]
after applying the sorting algorithm, it becomes
\[
B_{index} = \begin{bmatrix}
1 & 5 \\
2 & 8 \\
3 & 6 \\
3 & 9 \\
4 & 7
\end{bmatrix} \tag{2}
\]
Step VII: Compare last elements of rows with identical first element in \( B_{index} \). Discard rows with smaller last elements and keep the rows with the largest last element in a new matrix \( B_{ni} \) of size \( l \times 2 \) as the rows with smaller last elements are subsets of the row with the largest last element. Thus, \( B_{ni} \) is
\[
B_{ni} = \begin{bmatrix}
1 & 5 \\
2 & 8 \\
3 & 9 \\
4 & 7
\end{bmatrix} . \tag{3}
\]
Step VIII: The desired node number for \( A_i \) is
\[
\max \left\{ \min \{B_{ni}(j, 2) - B_{ni}(j + 1, 1) + 3\}, \lceil \log_2(m) \rceil \right\}, \quad j \in \{1, \ldots, l - 1\}.
\]
If \( \text{biState}_{t_3, B_{ni}(j+1,1)} = 1 \), then it is obvious that no less than one 1 exists within column \( B_{ni}(j+1,1) \) to \( B_{ni}(j,2) + 1 \) for row \( t_3 \). Thus, \( \min \{ B_{ni}(j,2) - B_{ni}(j+1,1) + 3 \} \) gives the original solution.

**Case two:** If

\[
\min \{ B_{ni}(j,2) - B_{ni}(j+1,1) + 3 \} < \lceil \log_2(m) \rceil
\]

then it is sufficient to consider this case (i.e., \( \text{biState}_{t_3, B_{ni}(j+1,1)} = 0 \)), due to the fact that for the other cases, \( \min \{ B_{ni}(j,2) - B_{ni}(j+1,1) + 3 \} \) is not less than \( \lceil \log_2(m) \rceil \). Thus, we extend indicated columns to guarantee containing column \( \text{biState}_{t_3, B_{ni}(j+1,1)} \) and also guarantee the length of them is \( \lceil \log_2(m) \rceil \). Then columns of length \( \lceil \log_2(m) \rceil \) will produce the desired solution. Assume

\[
c = \max \left\{ \min \{ B_{ni}(j,2) - B_{ni}(j+1,1) + 3 \}, \lceil \log_2(m) \rceil \right\},
\]

\[j \in \{1, \cdots, l-1\}\]

and since we have shown there exist node set that can differentiate different cell types, we can next prove \( c \) represents the optimal solution. Put another way, there is no set of nodes length \( c' < c \) that can distinguish \( A_l \) attractors. Assume there exist a subset \( C \subseteq \{c_1, \cdots, c_l\} \) with length \( c' \), then it is sufficient to show that \( c' < c \) does not hold. If \( c' \) is less than \( c \), then a row exists such that the set of column from column \( c_1 \) to \( c_1 \) exactly represents a subset of column \( B_{ni}(j,1) \) to \( B_{ni}(j,2) \). This means that a row exists with elements all corresponding to 0’s. Therefore, we can conclude that \( c' \) does not exist such that \( c' < c \), and maximizes \( \min \{ B_{ni}(j,2) - B_{ni}(j+1,1) + 3 \}, \lceil \log_2(m) \rceil \}, j \in \{1, \cdots, l-1\} \) corresponds to the desired solution of \( A_l \). \( \square \)

We give an example to illustrate the first case of our algorithm.

**Example 2:** Let \( A \) be given below:

\[
A = \begin{bmatrix}
0 & 1 & 0 & 1 & 0 & 1 \\
1 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 0 \\
\end{bmatrix}.
\]

Then we have

\[
\text{biState} = \begin{bmatrix}
1 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 1 \\
1 & 1 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 & 0 \\
\end{bmatrix}.
\]

and we can see that the matrix \( B_{con} \) will be in the form

\[
B_{con} = \{[1 \ 2 \ 3], [5 \ 6]\}.
\]

It is noted that the union of \( B_{con} \) is \( [1 \ 2 \ 3], [5 \ 6]\) which is a subset of \( U \). Thus any subset of the remaining elements of \( U \setminus B_{con} \) which is of length \( \lceil \log_2(m) \rceil \) (i.e., 2) will be the desired set of nodes. Actually, the elements of \( U \setminus B_{con} \) is 4, all the subset including 4 of length 2 is \( [3,4] \) or \( [4,5] \) which are exactly the desired minimal set of nodes necessary to discriminate attractors. Thus, \( [v_3, v_4] \) or \( [v_4, v_5] \) may be taken to represent the desired minimal consecutive nodes. However, if the union of \( B_{con} \) is exactly equal to \( U \), for instance,

\[
A = \begin{bmatrix}
1 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 \\
\end{bmatrix}.
\]

Then we have

\[
\text{biState} = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 1 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 \\
\end{bmatrix}.
\]

Similarly, we can obtain the matrix \( B_{con} \) which is in this form:

\[
B_{con} = \{[1 \ 2], [5 \ 6], [3 \ 4 \ 5], [1 \ 2]\}.
\]

Since the union of vectors in \( B_{con} \) ranges from 1 to \( n \), it is necessary to apply Step VI. Thus, the matrix \( B_{index} \) is sorted using bucket sort algorithm according to the first column and keep the row with the greatest last element in \( B_{index} \), then \( B_{ni} \) is given by

\[
B_{ni} = \begin{bmatrix}
1 & 2 \\
3 & 5 \\
5 & 6 \\
\end{bmatrix}.
\]

Then applying our proposed algorithm, the minimum set of columns is as follows:

\[
\max \left\{ \min \left\{ B_{ni}(j,2) - B_{ni}(j+1,1) + 3, \lceil \log_2(m) \rceil \right\}, j \in \{1, \cdots, l-1\} \right\}
\]

which is equal to 2. Furthermore, since the row number of \( A \) is 4, it requires at least \( \lceil \log_2(m) \rceil \) (i.e., 2) to distinguish the attractors, and this implies that it is impossible to find any less number for the other combination of attractor set. Thus, we conclude the length of the minimal set of nodes to distinguish the attractors is two. Based on this example, \( (v_2, v_3) \) likely represents the desired minimal consecutive nodes.

In the case where a BN consists of singleton and cyclic attractors, we consider the following example to illustrate our method.

**Example 3:** We assume the steady states are as follows,

\[
S_1 = \{(0, 0, 1, 0, 1, 0, 0)\}; \quad S_2 = \{(0, 0, 1, 0, 1, 0)\}; \quad S_3 = \{(0, 0, 1, 0, 1, 0, 1)\}; \quad S_4 = \{(0, 1, 0, 0, 1, 0, 0)\};
\]

then there are totally \( l_1 \cdot l_2 \cdot l_3 \cdot l_4 = 4 \) combinations. It is easy to see that the combination can be

\[
A_1 = \begin{bmatrix}
0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 \\
\end{bmatrix}.
\]
By applying our algorithm, the minimal length of consecutive nodes for each combination $A_i$ ($i = 1, 2, 3, 4$) are $4, 6, 3$ and $2$, respectively. After comparing the above values, we conclude that the minimal cardinality of consecutive nodes necessary to distinguish different attractor cycles is $2$. And the cyclic attractors are of period $2$, thus the algorithm efficiently works in $O(2^m n)$. Furthermore, it shall be noted that $m$ is typically extremely small, and it is therefore reasonable to utilize our method to singleton or cyclic attractors with large-scale networks.

### B. COMPLEXITY ANALYSIS

We have conducted a complexity analysis for our algorithm. The principle computational costs come from two parts: one comes from generating the matrix $\text{biState}$ and the other comes from the sorting algorithm. The first part clearly requires $O(C^2_m n)$ operations where $C^2_m = m(m - 1)/2$. Note that the number of attractors in a given BN is usually quite small, and as such conclude the computational complexity of this step is $O(n)$. For the sorting algorithm in Step VI, we have adopted bucket sort that works in $O(n)$, and repeat the algorithm for $P^m n$ time such that all possible combinations of singleton or cyclic attractors are considered. Thus, we conclude our algorithm is both effective and efficient for the problem of attractor observability and works in $O(P^m n)$.

### IV. NUMERICAL EXPERIMENTS

In this section, we performed computational experiments to validate our methodology for attractor observability. Initially, we randomly generated a singleton attractor set and repeated the experiment ten times with different simulated attractors. Then we took the average value as the final result. Notations utilized are as follows.

- $m$: attractor number;
- $n$: node number;
- time: average time (in seconds) for each trial;
- numNode: the desired minimal node number.

We utilize the above notations herein.

As seen in Table 2 our method was effective and efficient in solving the observability problem of singleton attractors for large-scale networks. Although the size of BN was up to 300 nodes, the average elapsed time was less than one second which is much faster than the ILP-based method [24].

Besides, the desired number of nodes to distinguish different attractor cycles was consistently small although the node number was large. For our next analysis, we randomly generated a cyclic attractor set, and set the period of cyclic attractors to two. Computational times are shown in Table 3 which illustrated the efficiency of our method. Running time was around 1 second even the number of network nodes was set to 1000. Therefore, our algorithm can be applied to cyclic attractors, instead of being confined to singleton attractors.

To examine the distribution of the desired number of nodes when the number of attractor was fixed ($m$), we fixed $m$ as $20$ and conducted analysis for different numbers of $n$ ranging from 100 to 1000. The desired number of nodes are shown in Figure 2. Since most of the desired numbers are 6, this indicates that our method is insensitive to the node number in BNs with a fixed number of attractors. Therefore, $n$ variation does not significantly effect desired node number for discriminating the attractor system.

We also examined the distribution of the desired node number when the number of nodes ($n$) was fixed to be 100. We adopted 10 as the step size for $m$ in [10,100]. As illustrated in Figure 3, most of the desired numbers of consecutive nodes are quite small although $m$ is large. Therefore, we can discriminate the set of attractors in large-scale network by identifying the small list of desired nodes.
TABLE 4. Correspondence of variables and gene names.

| compartment 1 | SLP | wg | WG | en | EN | HH | ptc | PTC | PH | SMO | ci | Cl | C1A | C1R |
|---------------|-----|----|----|----|----|----|-----|-----|----|-----|----|----|-----|-----|
| x₁            | x₂  | x₃ | x₄ | x₅ | x₆ | x₇ | x₈  | x₉  | x₁₀|x₁₁ | x₁₂| x₁₃| x₁₄| x₁₅|
| compartment 2 | SLP | wg | WG | en | EN | HH | ptc | PTC | PH | SMO | ci | Cl | C1A | C1R |
| x₁₆ | x₁₇ | x₁₈ | x₁₉ | x₂₀ | x₂₁ | x₂₂ | x₃₂ | x₂₃ | x₂₄ | x₂₅ | x₂₆ | x₂₇ | x₂₈ | x₂₉ | x₃₀ |
| compartment 3 | SLP | wg | WG | en | EN | HH | ptc | PTC | PH | SMO | ci | Cl | C1A | C1R |
| x₃₁ | x₃₂ | x₃₃ | x₃₄ | x₃₅ | x₃₆ | x₃₇ | x₃₈ | x₃₉ | x₄₀ | x₄₁ | x₄₂ | x₄₃ | x₄₄ | x₄₅ |
| compartment 4 | SLP | wg | WG | en | EN | HH | ptc | PTC | PH | SMO | ci | Cl | C1A | C1R |
| x₄₆ | x₄₇ | x₄₈ | x₄₉ | x₅₀ | x₅₁ | x₅₂ | x₅₃ | x₅₄ | x₅₅ | x₅₆ | x₅₇ | x₅₈ | x₅₉ | x₆₀ |

FIGURE 3. Sensitivity analysis on the number of attractors in BNs. Step size 10 is used for $m$ in $[10,100]$ and $n$ is fixed to be 100.

TABLE 5. Steady states of a drosophila model.

| compartment 1 | Wg | En | hh | ptc | PH | SMO | ci | Cl | C1A | C1R |
|---------------|----|----|----|-----|-----|-----|----|----|-----|-----|
| SLP | x₁₉ | x₂₀ | x₂₁ | x₂₂ | x₂₃ | x₂₄ | x₂₅ | x₂₆ | x₂₇ | x₂₈ |

To further verify our strategy, we applied our methodology to a popular genetic regulatory model in Drosophila melanogaster. Previously, in [26], Albert et al. proposed a model to describe embryonic pattern formation in the fruit fly Drosophila melanogaster. Their Boolean model consisted of 60 variables whose steady states were identified by manually solving a system of Boolean equations. After this, authors in [27] developed a strategy for efficiently identifying attractors of the network. To analyze the model, they first standardized their variables using the Boolean rules described in [26] by renaming them, i.e., SLP$_i$ or wg$_i$ to $x_1, \ldots, x_{60}$. The variable $x_i$ and corresponding genes are summarized in Table 4. They next applied their proposed method and obtained the steady states shown in Table 5. Each row in Table 5 corresponds to a stable attractor and each column represents a gene (or protein). Attractors have been denoted as binary values with 1 representing a gene being expressed (or high protein concentration), and 0 representing a gene not being expressed (or low concentration). Based on the established given attractor set, we successfully applied our model to identify the minimum cardinality of contiguous nodes (i.e., 22) necessary to distinguish them. Specifically, $(x_3, \ldots, x_{24})$ are the desired minimum consecutive variables which implies we can use these 22 nodes to discriminate the attractors and the 22 variables correspond to a list of genes or proteins which are shown in Table 6.

V. CONCLUSION

In this work, we addressed a novel problem, observability of singleton or cyclic attractors, which is of value in distinguishing different attractor cycles. We have developed an effective and efficient methodology to solve the captured problem, and complexity analysis shows our methodology works in $O(p^m n)$ time. As the number of attractors ($m$) is typically small, our method can be employed to resolve large-scale networks in addition to medium-size ones. We have also performed computational experiments verifying the efficiency and effectiveness of our novel method, and applied our algorithm to characterize a real-world scenario, i.e., segment polarity gene expression in Drosophila melanogaster. Our results suggest that our strategy may provide a novel tool for use in identifying useful genetic network biomarkers for the detection of disease. In the future work, further investigations will focus on developing more efficient methods to solve the captured problem.

ACKNOWLEDGMENT

The authors are grateful to the anonymous reviewers for their valuable suggestions and remarks which helped to improve the quality of the paper. This paper was presented in part at the workshop session of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM18) (2018) in Madrid, Spain. The full paper has not been published.

REFERENCES

[1] S. A. Kauffman, “Metabolic stability and epigenesis in randomly constructed genetic nets,” J. Theor. Biol., vol. 22, no. 3, pp. 437–467, Mar. 1969.
[2] S. Kauffman, *The Origins of Order: Self-organization and Selection in Evolution*. New York, NY, USA: Oxford Univ. Press, 1993.

[3] S. Kauffman, *At Home in the Universe*. Oxford, U.K.: Oxford Univ. Press, 1995.

[4] R. Albert and A.-L. Barabasi, “Dynamics of complex systems: Scaling laws or the period of Boolean networks,” *Phys. Rev. Lett.*, vol. 84, no. 24, pp. 5660–5663, Jun. 2000.

[5] R. Albert and H. Othmer, “The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in Drosophila melanogaster,” *J. Theor. Biol.*, vol. 225, no. 1, pp. 1–18, Jun. 2003.

[6] B. Drossel, T. Mihaljev, and F. Greil, “Number and length of attractors in a critical Kauffman model with connectivity one,” *Phys. Rev. Lett.*, vol. 94, no. 8, Mar. 2005, Art. no. 088701.

[7] B. Samuelsson and C. Troein, “Superpolynomial growth in the number of attractors in Kauffman networks,” *Phys. Rev. Lett.*, vol. 90, no. 9, Mar. 2003, Art. no. 98701.

[8] V. Devloo, P. Hansen, and E. Labbé, “Identification of all steady states in large networks by logical analysis,” *Bull. Math. Biol.*, vol. 65, no. 6, pp. 1025–1051, Nov. 2003.

[9] D. Irons, “Improving the efficiency of attractor cycle identification in BNs,” *Phys. D, Nonlinear Phenomena*, vol. 217, no. 1, pp. 7–21, May 2006.

[10] S. Q. Zhang, M. Hayashida, T. Akutsu, W. K. Ching, and M. K. Ng, “Algorithms for finding small attractors in Boolean networks,” *EURASIP J. Bioinf. Syst. Biol.*, vol. 2007, Jan. 2007, Art. no. 20180.

[11] T. Akutsu, S. Kuhara, O. Maruyama, and S. Miyano, “A system for identifying genetic networks from gene expression patterns produced by gene disruptions and overexpressions,” *Genome Informat.*, vol. 9, pp. 151–160, Sep. 1998.

[12] T. Akutsu and F. Bao, “Approximating minimum keys and optimal sub-structure screens,” in *Proc. Int. Conf. Combinatorics Conf. (COCOON)*, 1996, pp. 290–299.

[13] T. Akutsu, Y. Zhao, M. Hayashida, and T. Tamura, “Integer programming-based approach to attractor detection and control of Boolean networks,” *IEICE Trans. Inf. Syst.*, vols. E95–D, no. 12, p. 2960, 2012.

[14] Y. Qiu, T. Tamura, W. K. Ching, and T. Akutsu, “On control of singleton attractors in multiple Boolean networks: Integer programming-based method,” *BMC Syst. Biol.*, vol. 8, p. ST Jan. 2014.

[15] D. Luenberger, *Introduction to Dynamic Systems: Theory, Models, Applications*. Hoboken, NJ, USA: Wiley, 1979.

[16] R. Pal, A. Datta, M. L. Bittner, and E. R. Dougherty, “Intervention in context-sensitive probabilistic Boolean networks,” *Bioinformatics*, vol. 21, no. 7, pp. 1121–1218, Apr. 2005.

[17] R. Pal, A. Datta, M. Bittner, and E. R. Dougherty, “Optimal infinite-horizon control for probabilistic Boolean networks,” *IEEE Trans. Signal Process.*, vol. 53, no. 6, pp. 2375–2387, Jun. 2006.

[18] T. Akutsu, M. Hayashida, W.-K. Ching, and M. K. Ng, “Control of Boolean networks: Hardness results and algorithms for tree structured networks,” *J. Theor. Biol.*, vol. 244, no. 4, pp. 670–679, Feb. 2007.

[19] Y. Qiu, T. Tamura, W. K. Ching, and T. Akutsu, “Algorithms for finding small attractors in Boolean networks,” *EURASIP J. Bioinf. Syst. Biol.*, vol. 2007, Jan. 2007, Art. no. 20180.

[20] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabasi, “Observability of complex networks,” *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 52, pp. 21813–21818, Dec. 2009.

[21] Y. Qiu, Y. Huang, S. Tan, D. Li, A. Zijp-Tan, A. Fong, G. Borchert, and Y. Qiu, “Efficient algorithms for masking and finding quasi-identifiers,” in *Proc. Conf. Very Large Data Bases (VLDB)*, Vienna, Austria, Sep. 2007, pp. 83–93.

[22] X. Cheng, Y. Qiu, H. Jiang, M. Yim, and W. K. Ching, “Integer programming-based method for observability of singleton attractors in Boolean networks,” *IET Syst. Biol.*, vol. 11, no. 1, pp. 30–35, Feb. 2017.

[23] Y. Qiu, Y. Huang, S. Tan, D. Li, A. Zijp-Tan, A. Fong, G. Borchert and J. Huang, “Novel method for singleton and cyclic attractor observability in Boolean networks,” in *Proc. IEEE Int. Conf. Bioinf. Bioomed. (BIBM)*, Madrid, Spain, Dec. 2018, pp. 1526–1531.

[24] R. Albert and R. Othmer, “The topology of the regulatory interactions predicts the expression pattern of the segment polarity genes in Drosophila melanogaster,” *J. Theor. Biol.*, vol. 223, no. 1, pp. 1–18, Jul. 2003.

[25] F. Hinkelmann, M. Brandon, B. Guang, R. McNeill, G. Blekherman, A. Veliz-Cuba, and R. Laubenbacher, “ADAM: Analysis of discrete models of biological systems using computer algebra,” *BMC Bioinformatics*, vol. 12, Jul. 2011, Art. no. 295.
GLEN M. BORCHERT received the B.S. degree in biology from the University of Tennessee, in 2000, the Ph.D. degree in interdisciplinary genetics from the University of Iowa, in 2007, and completed Postdoctoral Fellowships at Illinois State and UC Berkeley. He started his own laboratory at the University of South Alabama, in 2012, and the Genetic Mechanisms Cluster of the National Science Foundation named Dr. Borchert, a NSF CAREER Investigator, in 2014. His current research focuses on identifying novel genes and genetic regulators in an array of bacterial, plant, and animal systems through next generation sequencing analysis.

JINGSHAN HUANG received the Ph.D. degree in computer science and engineering from the University of South Carolina, in 2007. He joined the University of South Alabama, in 2009. His research areas are data science and computational life science in general; in particular, the semantics and analysis of data as well as their innovative applications on human genomics and transcriptomics. His research is supported by NIH, NSF, and DOE.

HAO JIANG received the B.Sc. degree from the Harbin Institute of Technology, in 2009, and the Ph.D. degree from The University of Hong Kong, in 2013. She is currently an Assistant Professor with the School of Mathematics, Renmin University of China. Her research interests include mathematical modeling in bioinformatics and matrix analysis-based machine learning methods.