# Input-State Approach to Boolean Networks * 

Daizhan Cheng<br>Institute of Systems Science, Chinese Academy of Sciences, Beijing 100080, P.R.China<br>E-mail: dcheng@iss.ac.cn


#### Abstract

This paper investigates the structure of Boolean networks via input-state structure. Using the algebraic form proposed by the author, the logic-based input-state dynamics of Boolean networks, called the Boolean control networks, is converted into an algebraic discrete-time dynamic system. Then the structure of cycles of Boolean control systems is obtained as compounded cycles. Using the obtained input-state description, the structure of Boolean networks is investigated, and their attractors are revealed as nested compounded cycles, called Rolling Gears. This structure explains why small cycles mainly decide the behaviors of cellular networks. Some illustrative examples are presented.


Key Words. Algebraic form, network transition matrix, invariant subspace, input-state structure.

## 1 Introduction

Input-output structure is essential in systems and control theory. How about the cellular networks? It was pointed out by [11] that "Gene-regulatory networks are defined by trans and cis logic. ... Both of these types of regulatory networks have input and output." Ignoring outputs, this paper focuses on input-state structure only.

A Boolean network could be a description of genetic circuits, an explanation of self-organization in organisms, and the structure causing order in the evolution, which leads to life [12]. In Boolean network model gene expression is quantized to only two levels: "T" (True) and " F "(False), or " 1 " and " 0 " respectively, denoted by $D=\{T, F\}$, (or $D=\{1,0\}$ ). We refer to $[7]$ for logical notations, concepts and operators used in this paper, and refer to [2] for some related works in neural networks.

Denote the nodes of a network graph by $A_{1}(t), A_{2}(t), \cdots$. Each node is functionally related to the expression states of some other nodes. If $A_{i}$ is affected directly by $A_{j}$, there is a directed edge from $A_{j}$ to $A_{i}$, and it is said that $A_{j}$ is in the neighborhood of $A_{i}$. It can also be understood as the intracellular signal transduction from $j$-th cellular to $i$-th cellular. Throughout this paper we consider only the networks, which have fixed graph topologies. The actions between genes are described by logical rules, which are described by a logical dynamic equation [6].

$$
\left\{\begin{array}{l}
A_{1}(t+1)=f_{1}\left(A_{1}(t), A_{2}(t), \cdots, A_{n}(t)\right),  \tag{1}\\
A_{2}(t+1)=f_{2}\left(A_{1}(t), A_{2}(t), \cdots, A_{n}(t)\right) \\
\vdots \\
A_{n}(t+1)=f_{n}\left(A_{1}(t), A_{2}(t), \cdots, A_{n}(t)\right),
\end{array}\right.
$$

where $f_{i}, i=1,2, \cdots, n$ are logical functions (also called $n$-ary logical operators[1]).

[^0]We use an example to illustrate the graph and dynamics of a network. It will be used again in the sequel.

Example 1.1 The graph of a Boolean network is depicted in Fig. 1.


Figure 1: Graph of Network

Its dynamic model is assumed to be

$$
\left\{\begin{array}{l}
A(t+1)=A(t)  \tag{2}\\
B(t+1)=A(t) \rightarrow C(t) \\
C(t+1)=B(t) \vee D(t) \\
D(t+1)=\neg B(t) \\
E(t+1)=\neg C(t)
\end{array}\right.
$$

If the number of nodes in a Boolean network is $n$, then it is obvious that the state space considers of $2^{n}$ statues, which is a finite set described as $D^{n}$. So as a dynamic process on $D^{n}$, there must be at least a fixed point or a cycle, and eventually a trajectory starting from any initial state must enter a cycle (a fixed point can be considered as a cycle of length one). So a cycle is also called an attractor. For convenience, we briefly denote by $\Omega$ the set of attractors. For a state $x_{0}$, the smallest number of steps to enter $\Omega$ is called its transient period, denoted by $T_{t}\left(x_{0}\right)$. That is, let $x\left(t, x_{0}\right)$ be the trajectory starting from $x_{0}$ (i.e., $\left.x\left(0, x_{0}\right)=x_{0}\right)$. Then

$$
T_{t}\left(x_{0}\right)=\min \left\{k \mid x\left(k, x_{0}\right) \in \Omega\right\} .
$$

For overall network, the transient period is defined as

$$
T_{t}=\max _{x \in D^{n}} T_{t}(x)
$$

One of the most important issues in investigating a Boolean network is to find its cycles and transient period. These topics have been studied widely, e.g., in [10], [6]. But there was no reported technique, which solves the problem systematically so far.

To investigate the structure of a Boolean network, [5] proposed a way to convert system (1) into an standard discrete-time dynamic system. The key tool for this approach is the matrix expression of logic, based on semi-tensor product of matrices. We give a very brief introduction here and refer to [3, 4] for details. Investigating the structure of a network via its dynamics can also been found in $[8,16,13]$.

Definition 1.2 1. Let $X$ be a row vector of dimension $n p$, and $Y$ be a column vector with dimension $p$. Then we split $X$ into $p$ equal-size blocks as $X^{1}, \cdots, X^{p}$, which are $1 \times n$ rows. Define the semi-tensor product(STP), denoted by $\ltimes$, as

$$
\left\{\begin{array}{l}
X \ltimes Y=\sum_{i=1}^{p} X^{i} y_{i} \in \mathbb{R}^{n},  \tag{3}\\
Y^{T} \ltimes X^{T}=\sum_{i=1}^{p} y_{i}\left(X^{i}\right)^{T} \in \mathbb{R}^{n} .
\end{array}\right.
$$

2. Let $A \in M_{m \times n}$ and $B \in M_{p \times q}$. If either $n$ is a factor of $p$, say $n t=p$ and denote it as $A \prec_{t} B$, or $p$ is a factor of $n$, say $n=p t$ and denote it as $A \succ_{t} B$, then we define the STP of $A$ and $B$, denoted by $C=A \ltimes B$, as the following: $C$ consists of $m \times q$ blocks as $C=\left(C^{i j}\right)$ and each block is

$$
C^{i j}=A^{i} \ltimes B_{j}, \quad i=1, \cdots, m, \quad j=1, \cdots, q,
$$

where $A^{i}$ is $i$-th row of $A$ and $B_{j}$ is the $j$-th column of $B$.
Semi-tensor product of matrices is a generalization of the conventional matrix product, so the notation $\ltimes$ can be omitted. Moreover, all the basic properties of the conventional matrix product remain true for this extension.

Denote by

$$
\Delta_{k}:=\left\{\delta_{k}^{i} \mid i=1,2, \cdots, k\right\}
$$

where $\delta_{k}^{i}$ is the $i$-th column of identity matrix $I_{k}$. In the framework of semi-tensor product, the logical variables are expressed as a vector in $\Delta_{2}$ by identifying $T$ with $[1,0]^{T}$ and $F$ with $[0,1]^{T}$. Then the region $D$ is replaced by

$$
D_{v}=\left\{\left[\begin{array}{l}
1 \\
0
\end{array}\right],\left[\begin{array}{l}
0 \\
1
\end{array}\right]\right\}
$$

In this way, for a logical operator we can always find a matrix, called its structure matrix, and the action of a logical operator on logical variables becomes a (semi-tensor) product of the structure matrix with its arguments' vectors. For instance, consider conjunction $\wedge[7]$, its structure matrix is

$$
M_{c}=\left[\begin{array}{llll}
1 & 0 & 0 & 0 \\
0 & 1 & 1 & 1
\end{array}\right]
$$

Hence, (in vector form) we have

$$
P \wedge Q=M_{c} P Q, \quad P, Q \in D_{v}
$$

In general, we have
Proposition 1.3 Let $f$ be a logical function (operator) of $X_{1}, X_{2}, \cdots, X_{n}$, and

$$
\begin{equation*}
Y=f\left(X_{1}, X_{2}, \cdots, X_{n}\right) \tag{4}
\end{equation*}
$$

Then we can find a matrix, called the structure matrix of $f$ and denoted by $M_{f}$, such that

$$
\begin{equation*}
Y=M_{f} X_{1} X_{2} \cdots X_{n} \tag{5}
\end{equation*}
$$

Note that $X_{i}$ may appear in (4) many times, but in (5) $Y$ is multi-linear with respect to $X_{1}, \ldots, X_{n}$.
Using this vector expression, [5] converts a logical equation of a Boolean network into a discrete time linear system, called its algebraic form as $x(t+1)=L x(t)$. Analyzing the structure of $L$, precise formulas have been obtained to reveal the structure of the network.

This paper considers the Boolean control networks, which have input-state structure. We first propose a framework for Boolean control networks, and the structure of attractors of the networks is investigated. The input-state approach is then applied to the analysis of the structure of general Boolean networks and a structure of nested compounded cycles is obtained. We call such a structure "rolling gears" structure and will discuss some interesting properties of this kind of structures. We guess it could be used to reveal the hidden order in lives!

The paper is organized as follows: Section 2 reviews the converting technique from logical dynamic equation to algebraic one. An example is used to depict it. In Section 3, a framework of Boolean control networks is proposed and the structure of attractors of input-state type of networks is investigated. In Section 4, the input-state approach is implemented to analyze the structure of general Boolean networks and the structure of nested compounded cycles, called "rolling gears", is revealed. Section 5 contains two illustrative examples. Section 6 is a brief conclusion.

## 2 From Logical Equation to Algebraic Equation

In this section we briefly review the technique, developed in [5], which provides a systematic tool to treat Boolean networks. Assume a logical variable is expressed in a vector form. That is: $A_{i}(t) \in D_{v} \sim D$. (We will use scalar form $D$ and vector form $D_{v}$ alternatively without explanation, and use $D$ for both. From the text, it is very easy to figure out what form is used there.) Consider system (1). Since $f_{i}$, $i=1,2, \cdots, n$ are logical functions, according to Proposition 1.3, we can convert the equation (1) into an algebraic dynamic form as

$$
\left\{\begin{array}{l}
A_{1}(t+1)=M_{1} A_{1}(t) A_{2}(t) \cdots A_{n}(t)  \tag{6}\\
A_{2}(t+1)=M_{2} A_{1}(t) A_{2}(t) \cdots A_{n}(t) \\
\vdots \\
A_{n}(t+1)=M_{n} A_{1}(t) A_{2}(t) \cdots A_{n}(t)
\end{array}\right.
$$

where $M_{i}, i=1, \cdots, n$ are the structure matrices of $f_{i}$. Define

$$
\begin{equation*}
x(t)=A_{1}(t) A_{2}(t) \cdots A_{n}(t) \in \Delta_{2^{n}} . \tag{7}
\end{equation*}
$$

Multiplying $A_{i}(t+1)$ together yields

$$
\begin{equation*}
x(t+1)=\prod_{i=1}^{n}\left[M_{i} A_{1}(t) A_{2}(t) \cdots A_{n}(t)\right] \tag{8}
\end{equation*}
$$

Using the properties of semi-tensor product of matrices, (8) can be converted into an algebraic form as

$$
\begin{equation*}
x(t+1)=L x(t) \tag{9}
\end{equation*}
$$

where $L$ is called the network transition matrix of (1).
The following result reveals all the attractors from $L$.

Theorem 2.1 Consider system (1) with its network transition matrix L.

1. The number of length $s$ cycles, $N_{s}$, is inductively determined by

$$
\left\{\begin{array}{l}
N_{1}=\operatorname{Trace}(L),  \tag{10}\\
N_{s}=\frac{\operatorname{Trace}\left(L^{s}\right)-\sum_{k \in \mathcal{P}(s)} k N_{k}}{s}, \quad 2 \leq s \leq 2^{n},
\end{array}\right.
$$

where $\mathcal{P}(s)$ is the set of proper factors of $s$.
2. The elements on cycles of length $s$, denoted by $\mathcal{C}_{s}$, is

$$
\begin{equation*}
\mathcal{C}_{s}=\mathcal{D}_{a}\left(L^{s}\right) \backslash \cup_{k \in \mathcal{P}(s)} \mathcal{D}_{a}\left(L^{k}\right) \tag{11}
\end{equation*}
$$

where $\mathcal{D}_{a}(L)$ is the set of diagonal nonzero columns of $L$.
Note that $a \in \mathcal{P}(k)$, iff $a \in \mathbb{Z}_{+}, a<k$, and $k / a \in \mathbb{Z}_{+}$. For instance, $\mathcal{P}(8)=\{1,2,4\}, \mathcal{P}(12)=$ $\{1,2,3,4,6\}$, etc.

Denote

$$
r_{0}=\min \left\{k \mid L^{k} \in\left\{L^{k+1}, L^{k+2}, \cdots, L^{2^{2 n}}\right\}\right\}
$$

Then we have
Theorem 2.2 For system (1) the transient period

$$
\begin{equation*}
T_{t}=r_{0}=\min \left\{k \mid L^{k} \in\left\{L^{k+1}, L^{k+2}, \cdots, L^{2^{2 n}}\right\}\right\} . \tag{12}
\end{equation*}
$$

The following theorem provides an easy way to construct the regions of attraction.
Theorem 2.3 Given an $\eta \in \Delta_{2^{n}}$. Denote the columns of $L$, which equal to $\eta$, by $L_{i_{j}}=\eta, \quad j=$ $1,2, \cdots, k$. Then the set of parent points of $\eta$ is

$$
\begin{equation*}
L^{-1}(\eta)=\left\{\delta_{2^{n}}^{i_{1}}, \delta_{2^{n}}^{i_{2}}, \cdots, \delta_{2^{n}}^{i_{k}}\right\} \tag{13}
\end{equation*}
$$

We use an example to illustrate these results.
Example 2.4 Consider Example 1.1. Equation (2) can be converted into algebraic form as

$$
\left\{\begin{array}{l}
A(t+1)=A(t)  \tag{14}\\
B(t+1)=M_{i} A(t) C(t) \\
C(t+1)=M_{d} B(t) D(t) \\
D(t+1)=M_{n} B(t) \\
E(t+1)=M_{n} C(t)
\end{array}\right.
$$

Let $x(t):=A(t) B(t) C(t) D(t) E(t)$. Then

$$
\begin{align*}
x(t+1) & =A(t) M_{i} A(t) C(t) M_{d} B(t) D(t) M_{n} B(t) M_{n} C(t) \\
& =\left(I_{2} \otimes M_{i}\right) M_{r} A C M_{d} B D M_{n} B M_{n} C \\
& =\cdots\left(I_{4} \otimes M_{d}\right) A C B D M_{n} B M_{n} C \\
& =\cdots\left(I_{16} \otimes M_{n}\right) A C B D B M_{n} C \\
& =\cdots A C B W_{[2]} B D M_{n} C  \tag{15}\\
& =\cdots\left(I_{8} \otimes W_{[2]}\right) A C B^{2} D M_{n} C \\
& =\cdots\left(I_{4} \otimes M_{r}\right)\left(I_{16} \otimes M_{n}\right) A C B D C \\
& =\cdots A W_{[2]} B C W_{[2]} C D \\
& =\cdots\left(I_{2} \otimes W_{[2]}\right)\left(I_{8} \otimes W_{[2]}\right) A B C^{2} D \\
& =\cdots\left(I_{4} \otimes M_{r}\right) A(t) B(t) C(t) D(t) .
\end{align*}
$$

Starting from the second row of (15) the front constant coefficient matrix in previous row is replaced in the next row by "..." to save space.

Note that now in the left hand side of (15) there is no $E(t)$. To get $x(t)$ we have to add it. We can use a dummy operator [5]

$$
E_{d}=\left[\begin{array}{llll}
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1
\end{array}\right],
$$

which satisfies

$$
E_{d} P Q=Q, \quad \forall P, Q \in \Delta_{2} .
$$

Hence (15) can be converted further as

$$
\begin{align*}
x(t+1) & =\cdots\left(I_{4} \otimes M_{r}\right) A(t) B(t) C(t) E_{d} W_{[2]} D(t) E(t)  \tag{16}\\
& =\cdots\left(I_{8} \otimes\left(E_{d} W_{[2]}\right)\right) x(t) .
\end{align*}
$$

From (16) we have

$$
\begin{align*}
L= & \left(I_{2} \otimes M_{i}\right) M_{r}\left(I_{4} \otimes M_{d}\right)\left(I_{16} \otimes M_{n}\right)\left(I_{8} \otimes W_{[2]}\right)\left(I_{4} \otimes M_{r}\right)  \tag{17}\\
& \left(I_{16} \otimes M_{n}\right)\left(I_{2} \otimes W_{[2]}\right)\left(I_{8} \otimes W_{[2]}\right)\left(I_{4} \otimes M_{r}\right)\left(I_{8} \otimes\left(E_{d} W_{[2]}\right)\right) .
\end{align*}
$$

Note that $L \in M_{32 \times 32}$. It is easy to calculate $L$. We express it into a condensed form as

$$
\begin{aligned}
& L=\delta_{32}(4,4,4,4,11,11,11,11,2,2,6,6,9,9,13,13 \\
& 20,20,20,20,19,19,19,19,18,18,22,22,17,17,21,21)
\end{aligned}
$$

where the $i$-th component $\mu$ means the $i$-th column of $L$ is $\delta_{32}^{\mu}$.
It is easy to calculate that only when $i=1$, or $2, N_{i} \neq 0$, and $\operatorname{Trace}(L)=2$, $\operatorname{Trace}\left(L^{2}\right)=4$. Theorem 2.1 yields that there are 2 fixed points, which are

$$
11100, \text { and } 01100,
$$

and a cycle of length 2 , which is

$$
11010 \rightarrow 10101 \rightarrow 11010 .
$$

We can also check that $r_{0}=4$ and $L^{4}=L^{6}$, so the transient period is $T_{t}=4$.

## 3 Boolean Control Networks

A Boolean control network is defined as

$$
\left\{\begin{array}{l}
A_{1}(t+1)=f_{1}\left(A_{1}(t), A_{2}(t), \cdots, A_{n}(t), u_{1}(t), \cdots, u_{m}(t)\right)  \tag{18}\\
A_{2}(t+1)=f_{2}\left(A_{1}(t), A_{2}(t), \cdots, A_{n}(t), u_{1}(t), \cdots, u_{m}(t)\right) \\
\vdots \\
A_{n}(t+1)=f_{N}\left(A_{1}(t), A_{2}(t), \cdots, A_{n}(t), u_{1}(t), \cdots, u_{m}(t)\right)
\end{array}\right.
$$

where $u_{i}, i=1,2, \cdots m$ are inputs (or controls), which are logical variables satisfying certain logical rule, called the input network, described as

$$
\left\{\begin{array}{l}
u_{1}(t+1)=g_{1}\left(u_{1}(t), u_{2}(t), \cdots, u_{m}(t)\right)  \tag{19}\\
u_{2}(t+1)=g_{2}\left(u_{1}(t), u_{2}(t), \cdots, u_{m}(t)\right) \\
\vdots \\
u_{m}(t+1)=g_{m}\left(u_{1}(t), u_{2}(t), \cdots, u_{m}(t)\right)
\end{array}\right.
$$

In algebraic form a Boolean control network can be expressed as

$$
\left\{\begin{array}{l}
u(t+1)=G u(t), \quad u \in D^{m}  \tag{20}\\
x(t+1)=L(u) x(t), \quad x \in D^{n}
\end{array}\right.
$$

where $L(u)=L u(t)$ is the control-depending network transition matrix.
Example 3.1 Consider Example 1.1 again. It is very natural to take $A(t)$ as input. Ignoring $E(t)$, which is considered as an output, the system can be rewritten as

$$
\left\{\begin{array}{l}
B(t+1)=u(t) \rightarrow C(t)  \tag{21}\\
C(t+1)=B(t) \vee D(t) \\
D(t+1)=\neg B(t)
\end{array}\right.
$$

and the control network is

$$
u(t+1):=A(t+1)=A(t)
$$

Converting this system into algebraic form, we have

$$
\left\{\begin{array}{l}
u(t+1)=u(t)  \tag{22}\\
x(t+1)=L(u) x(t)
\end{array}\right.
$$

where $L(u)$ can be easily calculated as

$$
\begin{aligned}
L(u) & =M_{c} u\left(I_{2} \otimes M d\right)\left(I_{8} \otimes M n\right) W_{[2]} W_{[2,8]} M_{r} \\
& =\left[\begin{array}{cccccccc}
0 & 0 & 0 & 0 & 1 & 0 & 1-\alpha & 0 \\
1 & 1 & 1-\alpha & 1-\alpha & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1-\alpha \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \alpha & 0 \\
0 & 0 & \alpha & \alpha & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right],
\end{aligned}
$$

and $u=(\alpha, 1-\alpha)^{T}$, where $\alpha=0$ or 1 .
Now both $\delta_{2}^{1}$ and $\delta_{2}^{2}$ are fixed points of the control network. Using Theorem 2.1, it is easy to figure out that for $u=\delta_{2}^{1}$, there is a fixed point for the system, which is $x=(01000000) \sim 110$, there is also a cycle of length 2 , which is $101 \rightarrow 010 \rightarrow 101$. While $u=\delta_{2}^{2}$, there is only a fixed point 110 .

In general, we consider the structure of the Boolean control system (18), where the controls are varying, according to its own logical evolution rule (19).

Denote by $U=D^{m}$ the input space, $X=D^{n}$ the state space, and let $W=U \times X$ be the input-state (product) space. Let $w \in W$. It is easy to prove that there exist unique $u \in U$ and $x \in X$, such that $w=u x$. Now assume there is a cycle of length $k$ in the input-state space $W$. Say, it is

$$
C_{w}^{k}: w(0)=w_{0}=u_{0} x_{0} \rightarrow w(1)=w_{1}=u_{1} x_{1} \rightarrow \cdots \rightarrow w(k)=w_{k}=u_{k} x_{k}=w_{0}
$$

First of all, one sees easily that since $u_{0}=u_{k},\left\{u_{0}, u_{1}, \cdots, u_{k}\right\}$ contains, say, $j$ folds of a cycle of length $\ell$, say, $j \ell=k$. Hence $u_{\ell}=u_{0}$. Now let us see what condition the $\left\{x_{i}\right\}$ in the cycle $C_{w}^{k}$ should satisfy. Define a network transition matrix as

$$
\begin{equation*}
\Psi:=L\left(u_{\ell-1}\right) L\left(u_{\ell-2}\right) \cdots L\left(u_{1}\right) L\left(u_{0}\right) \tag{23}
\end{equation*}
$$

Starting from $w_{0}=u_{0} x_{0}$, we have $x$ component of the cycle $C_{w}^{k}$ as

$$
\begin{align*}
& x_{0} \rightarrow x_{1}=L\left(u_{0}\right) x_{0} \rightarrow x_{2}=L\left(u_{1}\right) L\left(u_{0}\right) x_{0} \rightarrow \cdots \rightarrow x_{\ell}=\Psi x_{0} \rightarrow \\
& x_{\ell+1}=L\left(u_{0}\right) \Psi x_{0} \rightarrow x_{\ell+2}=L\left(u_{1}\right) L\left(u_{0}\right) \Psi x_{0} \rightarrow \cdots \rightarrow x_{2 \ell}=\Psi^{2} x_{0} \rightarrow \\
& \vdots  \tag{24}\\
& x_{(j-1) \ell+1}=L\left(u_{0}\right) \Psi^{j-1} x_{0} \rightarrow x_{(j-1) \ell+2}=L\left(u_{1}\right) L\left(u_{0}\right) \Psi^{j-1} x_{0} \rightarrow \cdots \rightarrow x_{j \ell}=\Psi^{j} x_{0}=x_{0} .
\end{align*}
$$

We conclude that $x_{0} \in D^{n}$ is a fixed point of the equation (with $j>0$ being the smallest one)

$$
\begin{equation*}
x(t+1)=\Psi^{j} x(t) \tag{25}
\end{equation*}
$$

Conversely, if $x_{0} \in D^{n}$ is a fixed point of (25) and $u_{0}$ is a point on a cycle of control space $C_{u}^{\ell}$. Then it is obvious that we have the cycle (24).

Summarizing above arguments yields

Theorem 3.2 Consider the Boolean control network (20). A set $C_{w}^{k} \subset D^{k(n+m)}$ is a cycle of the control system with length $k$, iff for any point $w_{0}=u_{0} x_{0} \in C_{w}^{k}$, there exists an $\ell \leq k$ as a factor of $k$, such that $u_{0}, u_{1}=G u_{0}, u_{2}=G^{2} u_{0}, \cdots, u_{\ell}=G^{\ell} u_{0}=u_{0}$ is a cycle in the control space, and $x_{0}$ is a fixed point of the equation (25)(with $j>0$ being the smallest one).

Theorem 3.2 shows how to find all the cycles in the input-state space. First, we can find the cycles in the input space. Pick a cycle in the input space, say $C_{u}^{\ell}$, then for each point $u_{0} \in C_{u}^{\ell}$ we can construct an auxiliary system

$$
\begin{equation*}
x(t+1)=\Psi x(t) \tag{26}
\end{equation*}
$$

Now, say, $C_{u}^{\ell}=\left(u_{0}, u_{1}, \cdots, u_{\ell}=u_{0}\right)$ is a cycle in $U$, and $C_{x}^{j}=\left(x_{0}, x_{1}, \cdots, x_{j}=x_{0}\right)$ is a cycle of (26). Then a cycle $C_{w}^{k}, k=\ell j$, can be constructed by

$$
\begin{align*}
& w_{0}=u_{0} x_{0} \rightarrow w_{1}=u_{1} L\left(u_{0}\right) x_{0} \rightarrow w_{2}=u_{2} L\left(u_{1}\right) L\left(u_{0}\right) x_{0} \rightarrow \cdots \rightarrow \\
& w_{\ell}=u_{0} x_{1} \rightarrow w_{\ell+1}=u_{1} L\left(u_{0}\right) x_{1} \rightarrow w_{\ell+2}=u_{2} L\left(u_{1}\right) L\left(u_{0}\right) x_{1} \rightarrow \cdots \rightarrow \\
& \vdots  \tag{27}\\
& w_{(j-1) \ell}=u_{0} x_{(j-1)} \rightarrow w_{(j-1) \ell+1}=u_{1} L\left(u_{0}\right) x_{(j-1)} \rightarrow w_{(j-1) \ell+2}=u_{2} L\left(u_{1}\right) L\left(u_{0}\right) x_{(j-1)} \rightarrow \\
& w_{j \ell}=u_{0} x_{j}=u_{0} x_{0}=w_{0} .
\end{align*}
$$

We call this $C_{w}^{k}$ the compounded cycle of $C_{u}^{\ell}$ and $C_{x}^{j}$, denoted by $C_{w}^{k}=C_{u}^{\ell} \circ C_{x}^{j}$.
Note that from a cycle in the input space, $C_{u}^{\ell}$, we can choose any point as the starting point $u_{0}$. Then in equation (26) we have different $\Psi$, which produces different $C_{x}^{j}$. It is reasonable to guess that the final $C_{w}^{k}=C_{u}^{\ell} \circ C_{x}^{j}$ is independent of the choice of $u_{0}$. Otherwise, the picture will be mess. In the following we will prove this is true.

Definition 3.3 Let $C_{w}^{k}=\{w(t) \mid t=0,1, \cdots, k\}$ be a cycle in the input-state space, and $C_{u}^{\ell}$ be a cycle in the input space. Splitting $w(t)=u(t) x(t)$, we said that $C_{w}^{k}$ is attached to $C_{u}^{\ell}$ at $u_{0}$, if $w(0)=u_{0} x_{0}$, and

1. $u(t) \in C_{u}^{\ell}$, with $u(0)=u_{0}$;
2. $x(0)=x_{0}$ is a fixed point of (25) with $j=\frac{k}{\ell} \in \mathbb{Z}_{+}$.

Remark 3.4 According to Theorem 3.2, each cycle $C_{w}^{k}$ in the input-state space must be attached precisely to one cycle in the input space. In fact, the following argument shows that $C_{w}^{k}$ attaches $C_{u}^{\ell}$ at $u_{0}$ at moment $t=0$ (and the attaching point of $C_{w}^{k}$ is $w_{0}=u_{0} x_{0}$ ) and will attach it at $u_{1}$ at moment $t=1$ (with the attaching point of $C_{w}^{k}$ is $w_{1}=u_{1} x_{1}$ ) and so on. So, $C_{w}^{k}$ and $C_{u}^{\ell}$ are moving as two assembled gears.

Proposition 3.5 The sets of cycles in the input-state space, attached to any point of a given cycle $C_{u}^{\ell}$ are the same.

Proof. Let $C_{u}^{\ell}=\left\{u_{0}, u_{1}, \cdots, u_{\ell}=u_{0}\right\}$ be the cycle we concern. $S_{0}, S_{1}, \cdots, S_{\ell-1}$ be the set of cycles attached to $u_{0}, u_{1}, \cdots, u_{\ell-1}$ respectively. First, we show that

$$
S_{0} \subset S_{i}, \quad i=1,2, \cdots, \ell-1
$$

Let $C_{k}^{0}=\left\{w_{0}, w_{1}, \cdots, w_{k}\right\} \in S_{0}$, i.e., it is a cycle attached to $C_{u}^{\ell}$ at $u_{0}$. Using the elements of a control cycle, we can define

$$
L_{i}:=L\left(u_{i}\right), \quad i=0,1, \cdots, \ell-1
$$

Then we construct $\ell$ system matrices as

$$
\left\{\begin{array}{l}
\Psi_{0}:=L_{\ell-1} L_{\ell-2} \cdots L_{0} \\
\Psi_{1}:=L_{0} L_{\ell-1} L_{\ell-2} \cdots L_{1} \\
\vdots \\
\Psi_{\ell-1}:=L_{\ell-2} L_{\ell-3} \cdots L_{0}, L_{\ell-1}
\end{array}\right.
$$

Correspondingly, we then construct $\ell$ auxiliary systems as

$$
\begin{equation*}
x(t+1)=\Psi_{i} x(t), \quad i=0,1,, \cdots, \ell-1 \tag{28}
\end{equation*}
$$

Since $w_{0}=u_{0} x_{0} \in C_{w}^{k} \in S_{0}$, then $x_{0}$ satisfies

$$
\begin{equation*}
\left(\Psi_{0}\right)^{j} x_{0}=x_{0} \tag{29}
\end{equation*}
$$

Note that $w(1):=w_{1}=u_{1} L_{0} x_{0}$. To see that $w_{1} \in C_{k}^{1} \in S_{1}$ we have to show that $L_{0} x_{0}$ satisfies

$$
\begin{equation*}
\left(\Psi_{1}\right)^{j} L_{0} x_{0}=L_{0} x_{0} \tag{30}
\end{equation*}
$$

This is true because

$$
\begin{aligned}
L_{0} x_{0} & =L_{0}\left(\Psi_{0}\right)^{j} x_{0} \\
& =L_{0}\left(L_{\ell-1} \cdots L_{0}\right)^{j} x_{0} \\
& =L_{0} \underbrace{\left(L_{\ell-1} \cdots L_{0}\right) \cdots\left(L_{\ell-1} \cdots L_{0}\right)}_{j} x_{0} \\
& =\underbrace{\left(L_{0} L_{\ell-1} \cdots L_{1}\right) \cdots\left(L_{0} L_{\ell-1} \cdots L_{1}\right)}_{j} L_{0} x_{0} \\
& =\left(L_{0} L_{\ell-1} \cdots L_{1}\right)^{j} L_{0} x_{0} \\
& =\Psi_{1}^{j} L_{0} x_{0} .
\end{aligned}
$$

Similarly, we can show that

$$
u_{s} L_{s-1} L_{s-2} \cdots L_{0} x_{0} \in C_{k}^{s} \in S_{s}, \quad s=1,2, \cdots \ell-1
$$

Note that, precisely speaking, (30) can only assure there is a cycle of length $\ell \times j^{\prime}$ attached the cycle at $u_{1}$, where $j^{\prime}$ is a factor of $j$. But since the above definition of $\left\{\Psi_{i}\right\}$ is on a rotating style, staring from a point $w_{0}=u_{1} x_{0}^{\prime}$, same argument shows $j \leq j^{\prime}$. So $j^{\prime}=j$.

Example 3.6 Revisit Example 3.1. Now we may change the control to

$$
A(t+1)=\neg A(t)
$$

We have an obvious control cycle: $0 \rightarrow 1 \rightarrow 0$. Then we can easily calculate (using condensed form)

$$
\begin{aligned}
L(0) & =\delta_{8}(2,2,2,2,1,3,1,3) \\
L(1) & =\delta_{8}(2,2,6,6,1,3,5,7)
\end{aligned}
$$

Then we consider auxiliary system

$$
\begin{equation*}
x(t+1)=\Psi x(t) \tag{31}
\end{equation*}
$$

where

$$
\Psi=L(1) L(0)=\delta_{8}(2,2,2,2,2,6,2,6) .
$$

The routine calculation shows: (1) non-trivial power of $\Psi$ is 1 and $\operatorname{Trace}\left(\Psi^{1}\right)=2$. So there are two fixed points, which are $(0,1,0,0,0,0,0,0) \sim 110$ and $(0,0,0,0,0,0,1,0) \sim 010$. The overall cycles are depicted in Fig. 2, where the dash lines show the duplicated cycles. Overall, we have a cycle in the input space and two compounded cycles of length 2 in the input-state space. Fig. 2 shows the cycles of this control network.


Figure 2: Circles of a Control System

Finally, we consider the transient period of compounded cycles. Assume $C_{u}^{i}, i=1, \cdots, p$ are the cycles in control space. For a fixed $C_{u}^{i}$, which has length $\ell_{i}$, we can construct $\Psi_{i}$ and find the smallest $r^{i}$ such that

$$
\left(\Psi_{i}\right)^{r^{i}}=\left(\Psi_{i}\right)^{r^{i}+T_{i}}
$$

then it is clear that if a point will eventually enter the cycles attached to this cycle, then after $r^{i}$ (compounded) steps the second component will enter the rotating cycle. Note that $\Psi_{i}$ is a compounded mapping, consisting of $\ell_{i}$ steps. Taking the first part ( $C_{u}^{\ell}$ ) into consideration, it is easily seen that the transient period for cycles attached to $C_{u}^{i}$, denoted by $T_{t}\left(C_{u}^{i}\right)$, satisfies

$$
\begin{equation*}
\max \left\{r_{0}, \ell_{i}\left(r^{i}-1\right)\right\} \leq T_{t}\left(C_{u}^{i}\right) \leq \max \left\{r_{0}, \ell_{i}\left(r^{i}\right)\right\}, \quad i=1, \cdots, p \tag{32}
\end{equation*}
$$

Define

$$
\begin{aligned}
V_{i} & :=\max \left\{r_{0}, \ell_{i}\left(r^{i}-1\right)\right\} \\
U_{i} & :=\max \left\{r_{0}, \ell_{i}\left(r^{i}\right)\right\}, \quad i=1, \cdots, p .
\end{aligned}
$$

Then the following is obvious:
Proposition 3.7 The transient period of the control network satisfies

$$
\begin{equation*}
\max _{1 \leq i \leq p}\left\{V_{i}\right\}<T_{t} \leq \max _{1 \leq i \leq p}\left\{U_{i}\right\} \tag{33}
\end{equation*}
$$

## 4 Cascaded Boolean Networks

The input-state structure proposed in previous section is very useful in analyzing the structure of Boolean networks with cascading structure.

Definition 4.1 Consider system (1), where $x \in X=D^{n}$. A subspace $V=D^{k} \subset X$ is called an invariant subspace, if $x_{0} \in V$ implies $x\left(t, x_{0}\right) \in V, \forall t>0$.

From last section one sees easily that the control space is an invariant subspace of the control-state (product) space. Conversely, an invariant subspace can also be considered as a control space.

To testify if a subspace is a control invariant subspace, we can use either network graph or network equation. We use the following two examples to illustrate this.

Let $\left\{A_{i_{1}}, \cdots, A_{i_{s}}\right\}$ be a subset of the nodes of a network. $V=\operatorname{Span}\left\{A_{i_{1}}, \cdots, A_{i_{s}}\right\}$ means $V \subset X$ is the subspace of the states of the subnetwork with nodes $\left\{A_{i_{1}}, \cdots, A_{i_{s}}\right\}$ and edges between them, inherited from original graph.

Example 4.2 Consider the Fig. 3. One sees easily that $V_{1}=\operatorname{Span}\{A\}$ and $V_{2}=\operatorname{Span}\{A, B, C, D\}$ are two invariant subspaces. We have the nested invariant subspaces as

$$
V_{1} \subset V_{2} \subset X
$$



Figure 3: Invariant Subspaces

Note that $V=\operatorname{Span}\{A, B, C\}$ is not an invariant subspace, because it will be affected by $D$. (If you are familiar with graph theory, it is easy to see that a subspace is invariant iff the subgraph has in-degree zero.)

The structure of nested invariant subspaces can also be discovered from network equations. Consider the following example.

Example 4.3 Consider the following system

$$
\left\{\begin{array}{l}
A_{1}(t+1)=f_{1}^{1}\left(A_{1}(t), \cdots, A_{\ell}(t)\right),  \tag{34}\\
\vdots \\
\left.A_{\ell}(t+1)\right)=f_{\ell}^{1}\left(A_{1}(t), \cdots, A_{\ell}(t)\right) ; \\
B_{1}(t+1)=f_{1}^{2}\left(A_{1}(t), \cdots, A_{\ell}(t), B_{1}(t), \cdots, B_{m}(t)\right), \\
\vdots \\
\left.B_{m}(t+1)\right)=f_{m}^{2}\left(A_{1}(t), \cdots, A_{\ell}(t), B_{1}(t), \cdots, B_{m}(t)\right) ; \\
C_{1}(t+1)=f_{1}^{3}\left(A_{1}(t), \cdots, A_{\ell}(t), B_{1}(t), \cdots, B_{m}(t), C_{1}(t), \cdots, C_{n}(t)\right), \\
\vdots \\
\left.C_{n}(t+1)\right)=f_{n}^{3}\left(A_{1}(t), \cdots, A_{\ell}(t), B_{1}(t), \cdots, B_{m}(t), C_{1}(t), \cdots, C_{n}(t)\right) .
\end{array}\right.
$$

Then we have at least two nested invariant subspaces: $V_{1}=\operatorname{Span}\left\{A_{1}, \cdots, A_{\ell}\right\}=D^{\ell}$ and $V_{2}=\operatorname{Span}\left\{A_{1}, \cdots, A_{\ell}, B_{1}, \cdots, B_{m}\right\}=D^{\ell+m}$, and

$$
V_{1} \subset V_{2} \subset X=D^{\ell+m+n}
$$

We consider a cycle, say, $U^{3} \in X$. As we discussed in last section, it must attach to a cycle, say, $U^{2} \in V_{2}$. Similarly, $U_{2}$ must attach to a cycle, say, $U^{1} \in V_{1}$. Now in Fig. 4 we assume two cycles, $U_{1}^{2}, U_{2}^{2} \in V_{2}$ are attached to $U_{1} \in V_{1}$, and $U_{1}^{3}, U_{2}^{3} \in X$ are attached to $U_{1}^{2}$ and $U_{3}^{3}, U_{4}^{3} \in X$ are attached to $U_{2}^{2}$. We call such connected cycles as chaired gears.


Figure 4: Structure of cycles in a Cascaded Boolean Network

Chaired gears have the following properties:

- Each chair of gears, such as $U^{1} \rightarrow U_{1}^{2} \rightarrow U_{1}^{3}$, have multiplicative perimeters (precisely, the numbers of states in cycles), i.e., perimeter of $U_{1}^{3}$ is a multiplier of perimeter of $U_{1}^{2}$, perimeter of $U_{1}^{2}$ is a multiplier of perimeter of $U^{1}$.
- In each chair the smaller gears affect the larger gears, and the larger gears do not affect the smaller gears.
- Smallest gears look like steering gears, which steer the other gears to run.

Kauffman claimed that[12] in a cellular network the tiny attractors decide the vast order. The "rolling gears" structure explained why small cycles decide the order of the whole network. We guess that the structure of "Rolling Gears" may be used to explain the "hidden Order" in lives!

Finally, one may ask: why there should be invariant subspace? In fact, if a large or huge network has small cycles, then the small cycles with the elements in their region of attraction form small invariant subspaces. If there are no such small cycles, the system is in chaos [12]! So an ordered large scale network should have the structure of nested invariant subspaces.

## 5 Two Illustrative Examples

First example is from [14]. It is used for two purposes: (1) showing the standard algorithm; (2) demonstrating that the "small cycles" have decisive importance for the structure of overall network.

Example 5.1 Consider a system with 5 nodes, as

$$
\begin{equation*}
A_{i}=f_{i}\left(A_{j_{1}}, A_{j_{2}}, A_{j_{3}}\right), \quad i=1,2,3,4,5 \tag{35}
\end{equation*}
$$

where the logical functions $f_{i}, i=1, \cdots, 5$, are determined by the following truth table:

|  | $f_{1}$ | $f_{2}$ | $f_{3}$ | $f_{4}$ | $f_{5}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 |  |
| 1 | 1 | 1 | 1 | 0 |  |
|  | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |  |
|  | 1 | 1 | 1 | 0 | 0 |
|  | 1 | 1 | 1 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 |
| $j_{1}$ | 5 | 3 | 3 | 3 | 5 |
| $j_{2}$ | 2 | 5 | 1 | 4 | 4 |
| $j_{3}$ | 4 | 4 | 5 | 4 | 1 |

Tab. 1 Truth Table of (35)

Then the matrix form of the system (35) is

$$
\left\{\begin{array}{l}
A(t+1)=\left[\begin{array}{llllllll}
1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1
\end{array}\right] E(t) B(t) D(t),  \tag{36}\\
B(t+1)=\left[\begin{array}{llllllll}
1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1
\end{array}\right] E(t) E(t) D(t), \\
C(t+1)
\end{array}=\left[\begin{array}{llllllll}
1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1
\end{array}\right] E(t) A(t) E(t),\right.
$$

To get the structure matrix note that the first row of the structure matrix of $f_{i}$ is exactly the same as its values in truth table.

To convert the matrix form back to logical form, $\bmod (2)$ algebra is more convenient. Using mod(2) algebra, system (35) can be expressed as

$$
\left\{\begin{array}{l}
A(t+1)=B(t)+{ }_{2} D(t)+{ }_{2} B(t) \times_{2} D(t)  \tag{37}\\
B(t+1)=D(t)+{ }_{2} E(t)+{ }_{2} C(t) \times_{2} D(t) \times_{2} E(t) \\
C(t+1)=A(t)+{ }_{2} C(t)+{ }_{2} E(t)+{ }_{2} C(t) \times_{2} E(t)+{ }_{2} A(t) \times_{2} C(t) \times_{2} E(t) \\
D(t+1)=D(t) \\
E(t+1)=A(t) \times{ }_{2} D(t) \times_{2} E(t)
\end{array}\right.
$$

It is easy to see that the structure matrix of $\bmod (2)$ times, $\times_{2}$ is $M_{c}$ and $\bmod (2)$ plus, " $+_{2}$ " is easy to get that

$$
M_{+}=\left[\begin{array}{llll}
0 & 1 & 1 & 0 \\
1 & 0 & 0 & 1
\end{array}\right] .
$$

Let $x(t)=A(t) B(t) C(t) D(t) E(t)$. Then

$$
x(t+1)=M_{+}^{2} B D M_{c} B D M_{+}^{2} D E M_{c}^{2} C D E M_{+}^{4} A C E M_{c} C E M_{c}^{2} A C E D M_{c}^{2} A D E
$$

Now there is a normal routine to figure out L. In fact,

$$
\begin{aligned}
& L=\delta_{32}[1,6,4,16,13,2,8,12,1,6,20,32,13,2,24,28, \\
& 2,2,4,12,10,6,4,16,2,2,20,28,10,6,20,32]
\end{aligned}
$$

Then one can check that the non-trivial powers are 1 and 2, and

$$
\operatorname{Trace}(L)=4, \quad \operatorname{Trace}\left(L^{2}\right)=6
$$

We conclude that there are 4 fixed points and 1 cycle of length 2. Using Theorem 2.1, one sees easily that the fixed points are

$$
E_{1}=11111 ; \quad E_{2}=10011 ; \quad E_{3}=00100 ; \quad E_{4}=00000
$$

and the cycle of length 2 is

$$
11110 \rightarrow 11010 \rightarrow 11110
$$

The smallest repeating $L^{k}$ is $L^{3}=L^{5}$, so the transient period $T_{t}=3$.
Finally, we use Theorem 2.3 to get the whole picture of the state space.

- Starting from $E_{1}=11111$, we calculate its parent states, its grand parent states, and so on. We have, (in the following $[x]$ is used to show that $x$ is already on the cycle, so remove it from the retrieving chain,)

$$
E_{1}=11111 \sim \delta^{1} \Rightarrow\left[L_{1} \rightarrow \delta^{1}\right], L_{9} \rightarrow \delta^{9} \sim 10111 \Rightarrow \emptyset
$$

$\bullet$

$$
E_{2}=10011 \sim \delta^{13} \Rightarrow\left[L_{13} \rightarrow \delta^{13}\right], L_{5} \rightarrow \delta^{5} \sim 11011 \Rightarrow \emptyset
$$

$\bullet$

$$
\begin{aligned}
& E_{3}=00100 \sim \delta^{28} \Rightarrow\left[L_{28} \rightarrow \delta^{28}\right], L_{16} \rightarrow \delta^{16} \sim 10000 \Rightarrow \\
& L_{4} \rightarrow \delta^{4} \sim 11100 \Rightarrow\left\{\begin{array}{l}
L_{3} \rightarrow \delta^{3} \sim 11101 \Rightarrow \emptyset \\
L_{19} \rightarrow \delta^{19} \sim 01101 \Rightarrow \emptyset \\
L_{23} \rightarrow \delta^{23} \sim 01001 \Rightarrow \emptyset
\end{array}\right. \\
& L_{24} \rightarrow \delta^{24} \sim 01000 \Rightarrow L_{15} \rightarrow \delta^{15} \sim 10001 \Rightarrow \emptyset .
\end{aligned}
$$

- 

$$
\begin{aligned}
& E_{4}=00000 \sim \delta^{32} \Rightarrow\left[L_{32} \rightarrow \delta^{32}\right], L_{12} \rightarrow \delta^{12} \sim 10100 \Rightarrow \\
& L_{20} \rightarrow \delta^{20} \sim 01100 \Rightarrow\left\{\begin{array}{l}
L_{11} \rightarrow \delta^{11} \sim 10101 \Rightarrow \emptyset \\
L_{27} \rightarrow \delta^{27} \sim 00101 \Rightarrow \emptyset \\
L_{31} \rightarrow \delta^{31} \sim 00001 \Rightarrow \emptyset
\end{array}\right. \\
& L_{8} \rightarrow \delta^{8} \sim 11000 \Rightarrow L_{7} \rightarrow \delta^{7} \sim 11001 \Rightarrow \emptyset .
\end{aligned}
$$

- Next, we consider two points on cycle: $C_{1}=11010$ and $C_{2}=11110$. For $C_{1}$ :

$$
\begin{aligned}
& C_{1}=11010 \sim \delta^{6} \Rightarrow \\
& \left\{\begin{array}{l}
{\left[L_{2} \rightarrow \delta^{2} \sim 11110\right],} \\
L_{10} \rightarrow \delta^{10} \sim 10110 \Rightarrow\left\{\begin{array}{l}
L_{21} \rightarrow \delta^{21} \sim 01011 \Rightarrow \emptyset \\
L_{29} \rightarrow \delta^{29} \sim 00011 \Rightarrow \emptyset
\end{array}\right. \\
L_{22} \rightarrow \delta^{22} \sim 01010 \Rightarrow \emptyset \\
L_{31} \rightarrow \delta^{31} \sim 00010 \Rightarrow \emptyset
\end{array}\right.
\end{aligned}
$$

$\bullet$

$$
C_{2}=11110 \sim \delta^{2} \Rightarrow\left\{\begin{array}{l}
{\left[L_{6} \rightarrow \delta^{6} \sim 11010\right]} \\
L_{14} \rightarrow \delta^{14} \sim 11010 \Rightarrow \emptyset \\
L_{17} \rightarrow \delta^{17} \sim 01111 \Rightarrow \emptyset \\
L_{18} \rightarrow \delta^{18} \sim 01110 \Rightarrow \emptyset \\
L_{25} \rightarrow \delta^{25} \sim 00111 \Rightarrow \emptyset \\
L_{26} \rightarrow \delta^{26} \sim 00110 \Rightarrow \emptyset
\end{array}\right.
$$

The following state transition diagram from [14] verifies our conclusion.

Part 1. $D=0$


Part 2. $D=1$


Figure 5: The State-transition Diagram

What is significant in this example is the following observation: There is a smallest "cycle": fixed point D. From Fig. 5 one sees easily that for $D=0$ and $D=1$ the topological structures of the state space graphs are completely different.

Next, we analyze a system, which is used to simulate gene and protein signaling activity patterns [9].

Example 5.2 The network depicted in Fig. 6 and Tab. 2 is presented in [9] to simulate gene and protein signaling activity patterns within a small model Boolean network. For notational brevity, we use A for "Erk", B for "cyclin D1", C for "p27", D for "cyclin E", E for "E2F", F for " $p R b$ ", $G$ for " $S$ phase
genes", U for "growth factors", V for cell"shape(spreading)", and W for "X". We refer to [9] for the biological meanings of the notations.


Figure 6: Gene and Protein Signaling Activity Patterns

| network element | $W$ | $A$ | $B$ | $C$ | $D$ | $E$ | $F$ | $G$ | $H$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| input 1 | $U$ | $U$ | $A$ | $H$ | $E$ | $D$ | $B$ | $F$ | $H$ |
| input 2 | $V$ | $G$ | $C$ | $W$ | $C$ | $F$ | $D$ | $E$ | $G$ |
| boolean function | and | not if | not if | implicate | implicate | not if | nand | not if | not if |

Tab. 2 Logical Relations

Then the logical equation is expressed as

$$
\left\{\begin{array}{l}
A(t+1)=\neg(U(t) \rightarrow G(t))  \tag{38}\\
B(t+1)=\neg(A(t) \rightarrow C(t)) \\
C(t+1)=H(t) \rightarrow W(t) \\
D(t+1)=E(t) \rightarrow C(t) \\
E(t+1)=\neg(D(t) \rightarrow F(t)) \\
F(t+1)=\neg(B(t) \wedge D(t)) \\
G(t+1)=\neg(F(t) \rightarrow E(t)) \\
H(t+1)=\neg(H(t) \rightarrow G(t))
\end{array}\right.
$$

As for control network, we have

$$
\left\{\begin{array}{l}
U(t+1)=g_{1}(U(t))  \tag{39}\\
V(t+1)=g_{2}(V(t)) \\
W(t+1)=g_{3}(U(t), V(t))
\end{array}\right.
$$

In matrix form, we have algebraic equation as

$$
\left\{\begin{array}{l}
A(t+1)=M_{n} M_{i} U(t) G(t)  \tag{40}\\
B(t+1)=M_{n} M_{i} A(t) C(t) \\
C(t+1)=M_{i} H(t) W(t) \\
D(t+1)=M_{i} E(t) C(t) \\
E(t+1)=M_{n} M_{i} D(t) F(t) \\
F(t+1)=M_{n} M_{c} B(t) D(t) \\
G(t+1)=M_{n} M_{i} F(t) E(t) \\
H(t+1)=M_{n} M_{i} H(t) G(t)
\end{array}\right.
$$

As in [9], we first set the control network as

$$
\left\{\begin{array}{l}
U(t+1)=U(t)  \tag{41}\\
V(t+1)=V(t) \\
W(t+1)=U(t) \wedge V(t)
\end{array}\right.
$$

Case 1. $U(0)=V(0)=[1,0]^{T}$, $\sigma_{1}=\sigma_{2}=$ identity, i.e., $U(t)$ and $V(t)$ equal to $[1,0]^{T}$ constantly.
Plugging them into (39) yields the system

$$
L(U(t), W(t))=L\left(\left[\begin{array}{l}
1 \\
0
\end{array}\right],\left[\begin{array}{l}
1 \\
0
\end{array}\right]\right)
$$

In calculation, a control can be treated as a logical operator, so the routine for calculating network transition matrix remain available. Then it is easy to get the following result:

- The only attractor is a fixed point 00110110;
- $L^{10}=L^{11}$ and the transient period is $T_{t}=10$.

Case 2. $U(0)=[1,0]^{T}$ and $V(0)=[0,1]$. In this case, we have same conclusion.
Case 3. $U(0)=[0,1]^{T}$, then we always have $W(t)=[0,1]^{T}, t \geq 1$. The conclusion is

- The only attractor is a fixed point 00110110;
- $L^{6}=L^{7}$ and the transient period is $T_{t}=6 . \quad$ (Taking $W(0)$ into consideration, $T_{t}$ should be 7.)

Next, we assume the control network is as

$$
\left\{\begin{array}{l}
U(t+1)=\neg U(t)  \tag{42}\\
V(t+1)=\neg V(t) \\
W(t+1)=V(t) \wedge W(t)
\end{array}\right.
$$

Then we have two sequence of nested invariant subspaces. We consider them separately. Consider the first chair, which is

$$
V_{1}=\operatorname{Span}\{U\} \subset V_{2}=\operatorname{Span}\{A, B, C, D, E, F, G, H, U, V, W\}
$$

In $V_{1}$ we have an obvious cycle: $0 \rightarrow 1 \rightarrow 0$. For $U=0$, a routine computation shows that there is only a cycle of length 2, which is

$$
0011011010 \rightarrow 0011011000 \rightarrow 0011011010
$$

$L(0)$ is a $1024 \times 1024$ matrix. We omit it here. But we can calculate that $L(0)^{7}=L(0)^{9}$ and $T_{t}=7$. Similarly, for $U=1$, we have the same cycle. $L(1)^{11}=L(1)^{13}$ and $T_{t}=11$.

Finally, let $\Psi=L(1) L(0)$. Then $\Psi$ has only one fixed point 0011011010 . We conclude that overall in $U$ space we have only one cycle $0 \rightarrow 1 \rightarrow 0$; and in whole space we have only one product cycle

$$
0 \times 0011011010 \rightarrow 1 \times 0011011000 \rightarrow 0 \times 0011011010
$$

They are depicted in Fig. 7 (a), where $I-S$ is the overall input-state space.
Next, we consider the second chair, which is

$$
V_{1}=\operatorname{Span}\{V\} \subset V_{2}=\operatorname{Span}\{V, W\} \subset V_{3}=\operatorname{Span}\{A, B, C, D, E, F, G, H, U, V, W\}
$$

First, there is a trivial cycle in $V$ space as: $0 \rightarrow 1 \rightarrow 0$.
Then in $V \times W$ space, it is easy to calculate that

$$
L(0)=\left[\begin{array}{ll}
0 & 0 \\
1 & 1
\end{array}\right], \quad L(1)=\left[\begin{array}{ll}
1 & 0 \\
0 & 1
\end{array}\right]
$$

So

$$
\Psi=L(1) L(0)=\left[\begin{array}{ll}
0 & 0 \\
1 & 1
\end{array}\right]
$$

which has unique fixed point $(0,1) \sim 0$. We conclude that in $V \times W$ space we have only one cycle: $0 \times 0 \rightarrow$ $1 \times 0 \rightarrow 0 \times 0$. Finally, we consider the space $V \times W \times A B C D E F G H U$. Calculating $\Psi=L(0 \times 0) L(1 \times 0)$ it is easy to find that the only cycle is a fixed point: 001101101. We conclude that there is only one cycle of length 2 in the overall product space, which is: $0 \times 0 \times 001101101 \rightarrow 1 \times 0 \times 001101100$. Circles in nested subspaces are depicted in Fig. 7 (b).

## 6 Conclusion

In this paper we first reviewed the main results in [5]: How to convert a logical form of Boolean equation into an algebraic form. Then we proposed a framework for control Boolean networks. The structure of the cycles in the input-state space was obtained. Using the input-state technique, we investigated the general structure of Boolean networks. Then, as a network has cascade structure, a structure of chaired cycles, called Rolling Gears, has been revealed. It was shown that the Rolling Gears structure is the explanation that tiny attractors decide the order of networks.

## Acknowledgment

A toolbox is provided in http://lsc.amss.ac.cn/ dcheng/STP Toolbox for the related computations.
(a). Circles in $U \subset I-S$ spaces.


Figure 7: Chaired Circles

## References

[1] Barnes, D.W., An Algebraic Introduction to Mathematical Logic, Springer-Verlag, New York, 1975.
[2] Beg, A., P.W.C. Prasad, A. Beg, Applicability of feed-forward and recurrent neural networks to Boolean function complexity modeling, Expt. Sys. Appl., Vol. 34, No. 4, 2436-2443, 2008.
[3] Cheng, D., On Logic-based Intelligent Systems, Proc. ICCA 2005, Budapest, 71-76, 2005.
[4] Cheng, D., H. Qi, Matrix Expression of Logic and Fuzzy Control, Proc. 44 th IEEE CDC, Seville, 3273-3278, 2005.
[5] Cheng, D., H. Qi, Linear representation of dynamics of Boolean networks, sumbitted to IEEE TAC (under revision), (preprint: http://lsc.amss.ac.cn/ dcheng).
[6] Farrow, C., J. Heidel, H. Maloney, J. Rogers, Scalar equations for synchronous Boolean networks with biological applications, IEEE Trans. Neural Networks, Vol. 15, No. 2, 348-354, 2004.
[7] Hamilton, A.G., Logic for Mathematicians, revised ed., Cambridge University Press, Cambridge, 1988.
[8] Ho, C. Y.-F., B.W.-K., Ling, H.-K. Lam, and M.H.U. Nasir, Global convergence and limit cycle behavior of weights of perceptron, IEEE Trans. Neural Networks, Vol. 19, No. 6, 938-947, 2008.
[9] Huang, S., I. Ingber, Shape-dependent control of cell growth, differentiation, and apotosis: Switching between attractors in cell regulatory networks, Exp. Cell Res., Vol. 261, 91-103, 2000.
[10] Heidel, J., J. Maloney, J. Farrow, and J. Rogers, Finding cycles in synchronous Boolean networks with applications to biochemical systems, Int. J. Bifurcat. Chaos, Vol. 13, No. 3, 535-552, 2003.
[11] Ideker, T., T. Galitski, L. Hood, A new approach to decoding life: systems biology, Annu. Rev. Genomics Hum. Genet., Vol. 2, 343-372, 2001.
[12] Kauffman, S.A., At Home in the Universe, Oxford Univ. Press, Oxford, 1995.
[13] Lian, J., Y. Lee, S.D. Sudhoff, S.H. Zak, Self-organizing radial basis function for raal-time approximation of continuous-time dynamical systems, IEEE Trans. Neural Networks, Vol. 19, No. 3, 460-474, 2008.
[14] Paszek, E., Boolean Networks, http://cnx.org/content/m12394/latest/, 2008.
[15] Waldrop, M.M., Complexity, Touchstone, New Your, 1992.
[16] Wu, S., T.W.S., Chow, Self-organizing and self-evolving neurons: A new newral netwrok for optimization, IEEE Trans. Neural Networks, Vol. 18, No. 2, 385-396, 2008.


[^0]:    * Supported partly by NNSF 60674022,60736022 , and 60221301 of China.

