A Semi-tensor Product Approach for Probabilistic Boolean Networks

Xiaoqing Cheng, Yushan Qiu*, Wenpin Hou and Wai-Ki Ching
Advanced Modeling and Applied Computing Laboratory
Department of Mathematics
The University of Hong Kong, Hong Kong
Emails: mechxqxiaofen@gmail.com, yushanqiu2526374@163.com, whou@connect.hku.hk, wching@hku.hk

***Abstract*—Modeling genetic regulatory networks is an important issue in systems biology. Various models and mathematical formalisms have been proposed in the literature to solve the capture problem. The main purpose in this paper is to show that the transition matrix generated under semi-tensor product approach (Here we call it the probability structure matrix for simplicity) and the traditional approach (Transition probability matrix) are similar to each other. And we shall discuss three important problems in Probabilistic Boolean Networks (PBNs): the dynamic of a PBN, the steady-state probability distribution and the inverse problem. Numerical examples are given to show the validity of our theory. We shall give a brief introduction to semi-tensor and its application. After that we shall focus on the main results: to show the similarity of these two matrices. Since the semi-tensor approach gives a new way for interpreting a BN and therefore a PBN, we expect that advanced algorithms can be developed if one can describe the PBN through semi-tensor product approach.**

Keywords: Boolean Networks (BNs), Semi-tensor Product Approach, Inverse Problem, Probabilistic Boolean Networks (PBNs), Similar Matrices, Steady-state Distribution.

I. Introduction

Modeling genetic regulatory networks is one of the important topics in systems biology [7], [11]. A number of models and mathematical formalisms have been proposed to explain the genetic intersections, including linear models [19], Bayesian networks [16] and its extensions, differential equations model [14], Boolean Networks (BNs) and its extension Probabilistic Boolean Networks (PBNs) [17], [18]. BN and PBN models are some of the most attractive models. BN was first introduced by Kauffman [12], [13]. In a BN, the expression states of the gene are categorized into levels, either on (1) or off (0). The dynamics of a BN can be viewed as a process that each gene is governed by a function (called Boolean function). BN is called a deterministic model since the target gene only depends on the initial state and the set of Boolean functions. And a BN will eventually enter into an attractor cycle, whose length could be either 1 (singleton attractor) or more than 1 (periodic attractor). Finding the attractor cycles and their features are important topics for a BN. The attractor cycles in a BN may reveal some cancer cells or abnormality in a cell. Thus finding the attractor cycles and their features are of important topics in BN. Other research problems and developments related to BNs can be found in [1], [2], [10], [12], [13].

Shmulevich [17] pointed out that the holistic behavior of the network should be studied because it is believed that genes are not independent of each other. Based on a couple of reasons (e.g. the limitation that BN is a deterministic model, BN may only reveal part of the information while generating to the next state, the desire for an open system and so on), a stochastic version of BNs, namely, Probabilistic Boolean Networks (PBNs) was proposed [17], [18]. It is based on the appealing rule-based property of BN, but it also incorporates with stochastic features. PBN owns a couple of advantages over a BN, for example, it can cope with the uncertainty in the data and the Boolean functions due to its stochastic nature. The proportion of steady-state probabilistic distribution provides a holistic picture of the network. It can also reveal whether the genes are interacting with each other, and how they interact.

Cheng et al. [3], [4], [5] proposed an algebraic approach called the semi-tensor product approach. And they successfully applied their theory to BN problems and BN control problems, see for instance [6]. Based on Cheng's works, Yang and Li also applied semi-tensor approach to PBN control problems [22][23], however, they did not discuss much about the theoretical support of applying semi-tensor product to PBN control problems. In the semi-tensor product approach theory, a mapping is defined from the gene expression state to the column of identity matrix I_2, where "true" equals to the first column and "false" equals to the second column. Therefore there is no logical functions and logical expressions in each iteration step. Then, they define a kind of operation called semi-tensor product, which is based on Kronecker product and primitive product of matrices. The semi-tensor product shares all the appealing properties with the primitive matrix product. This can be easily shown under its definition. Hence BN can be transformed into an algebraic form by multiplying all the BN equations together. The most salient limitation of the semi-tensor approach is it will take much effort in transforming a BN into that form. But the flaws do not detract from the jade's essential beauty. Semi-tensor approach is a powerful mathematical method and it also provides a new way for dealing with genetic regulatory networks.

The main contribution of this paper is that we proved the probability transition matrix and probability structure matrix are similar matrices. Thus, semi-tensor product theory is

978-1-4799-7294-4/14/$31.00 ©2014 IEEE

applicable to PBN problems. For a given PBN, the transition probability matrix generated from the two ways (the traditional one and using the semi-tensor technique) are different. Actually, the transition matrix of a BN generated by the semi-tensor product approach is called a structure matrix [5]. So we call the probability transition matrix of a PBN generated from the semi-tensor product approach probability structure matrix. Here we try to find the relationship between them. We can show that they are similar matrices, which is one of the main results in this paper. Based on the similarity property of these two matrices, we discuss three important problems in studying a PBN: (i) the dynamics of a PBN, (ii) the steady-state probability distribution and (iii) the inverse problem of constructing a PBN.

The remainder of this paper is structured as follows. Section 2 gives a review on some important concepts of BNs and PBNs. Section 3 presents about the main results on semi-tensor product for PBNs, and we show that the two matrices are similar. We propose three important problems in studying a PBN in Section 4 and discuss their relationships in these two approaches. The final section concludes the paper.

II. Preliminaries

A. Boolean Networks and Probabilistic Boolean Networks

1) Boolean Networks (BNs): BN $G(V, F)$ is a special case of a sequential dynamic system [15], consisting of a set of binary nodes V (also called Boolean variables) such that each of which has a Boolean function assigned to it. Suppose there are n genes in the BN, F is the set of the Boolean functions where

$$F = \{f_1, f_2, \ldots, f_n\}, f_i : \{0,1\}^n \to \{0,1\}.$$

And V is the set of all the vertices, $V = \{v_1, v_2, \ldots, v_n\}$. The value of v_i represents the state of gene i, either 0 (on) or 1 (off). The dynamics of the BN can be expressed as

$$v_i(t+1) = f_i(v_1(t), v_2(t), \ldots, v_n(t)) = f_i(\mathbf{v}(t)).$$

Here $\mathbf{v}(t)$ is called Gene Activity Profile (GAP). Since we know that $v_i \in \{0,1\}$, the value of $\mathbf{v}(t)$ can be taken from

$$S = \{\underbrace{00\ldots0}_{n}, \underbrace{00\ldots1}_{n}, \ \ldots, \ \underbrace{11\ldots1}_{n}\}.$$

The size of set S is 2^n.

2) Probabilistic Boolean Networks (PBNs): BN is a deterministic model, the only randomness comes from its initial state. However, in a biological system, noise and randomness are usually unavoidable, and there always exists noise in experimental data, so a stochastic model is more appropriate. The concept and idea of a Probabilistic Boolean Network (PBN) are introduced in order to capture the stochastic nature of the biological system. A PBN is an open model where the data and the Boolean functions can be changed in different cases. The PBN model shares similar rules with a BN except that more than a BN function is assigned to each gene. Suppose l_i Boolean functions are assigned to gene v_i, denoted by $f_i^1, f_i^2, \ldots, f_i^{l_i}$. And the probability of choosing the jth Boolean function is c_i^j. This implies that

$$\sum_{j=1}^{l_i} c_i^j = 1, \ 0 < c_i^j < 1, \quad \text{for} \ \ i = 1, 2, \ldots, n.$$

If we choose the j_ith Boolean function for gene v_i, then the BN can be expressed as $BN_{j_1 j_2 \ldots j_n}$, where $j_i \in \{1, 2, \ldots, l_i\}$. It can be seen that there are totally $N = \prod_{i=1}^{n} l_i$ BNs. And we assume that it is independent to choose the Boolean function for each gene, so we have the probability of choosing $BN_{j_1 j_2 \ldots j_n}$ given by

$$P\{f_1 = f_1^{j_1}, f_2 = f_2^{j_2}, \ldots, f_n = f_n^{j_n}\} = \prod_{i=1}^{n} c_i^{j_i} = q_{j_1 j_2 \ldots j_n}.$$

We use $A_{j_1 j_2 \ldots, j_n}$ to denote the transition probability matrix for $BN_{j_1 j_2 \ldots j_n}$. And we use the compact form of $j_1 j_2 \ldots j_n$, then the BNs can be denoted by $BN_1, BN_2, \ldots, BN_N$. The probability of choosing the jth BN is q_j and the transition probability matrix for the jth BN is $A_j, j \in \{1, 2, \ldots, N\}$. Then the probability transition matrix for the PBN is given by $A = \sum_{j=1}^{N} q_j A_j$. Since

$$\begin{aligned} & P\{\mathbf{V}(t+1) = \mathbf{a} | \mathbf{V}(t) = \mathbf{b}\} \\ = \ & \textstyle\sum_{j=1}^{N} \{\mathbf{V}(t+1) = \mathbf{a} | \mathbf{V}(t) = \mathbf{b} | \text{the jth network is chosen}\} q_j. \end{aligned}$$

Example 1: This is an example of a PBN and its BNs. The truth table of the PBN is given by

	State	f_1^1	f_2^1	f_1^2	f_2^2
1	00	0	0	1	1
2	01	1	0	0	1
3	10	0	1	0	1
4	11	1	1	0	0
c_j^i		0.4	0.6	0.1	0.9

Based on the truth table, we have four BNs and they are listed as follows:

$$A_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad A_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad A_4 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

And we have $q_1 = c_1^1 c_1^2 = 0.04$, $q_2 = c_1^1 c_2^2 = 0.36$, $q_3 = c_2^1 c_1^2 = 0.06$, $q_4 = c_2^1 c_2^2 = 0.54$, so the transition probability matrix A is given by

$$A = \sum_{i=1}^{4} q_i A_i = \begin{pmatrix} 0 & 0.06 & 0.04 & 0 \\ 1 & 0.54 & 0.36 & 0 \\ 0 & 0.04 & 0.06 & 1 \\ 0 & 0.36 & 0.54 & 0 \end{pmatrix}.$$

978-1-4799-7294-4/14/$31.00 ©2014 IEEE

B. Semi-tensor product

The semi-tensor product is defined in the following.

Definition 1: (Cheng et al. [5]) Given an $m \times n$ matrix A and a $p \times q$ matrix B, the semi-tensor product of A and B is given by

$$A \ltimes B = (A \otimes I_{l/n})(B \otimes I_{l/p})$$

where l is the least common multiple of n and p. And for any matrix M and N

$$M \otimes N = \begin{pmatrix} m_{11}N, & m_{12}N, & \dots, & m_{1s}N \\ m_{21}N, & m_{22}N, & \dots, & m_{2s}N \\ \vdots & \vdots & \vdots & \vdots \\ m_{t1}N, & m_{t2}N, & \dots, & m_{ts}N \end{pmatrix}$$

The size of matrix M is $t \times s$.

By deleting the largest common factor of n and p, the size of the semi-tensor product matrix can be determined. For example, if the size of A is $m \times ax$, the size of B is $ay \times q$, and a is the largest common factor of ax and ay, then the size of $A \ltimes B$ is $my \times xq$. The proof can be easily derived from the definition. The following two definitions are usually adopted in semi-tensor product theory.

Definition 2: (Cheng et al. [5]) δ_n^j was defined as the jth column of matrix I_n.

This is the definition of logical matrix.

Definition 3: (Cheng et al. [5]) A matrix $L \in \mathcal{M}_{n\times m}$ is called a logical matrix if each column of L is in the form of $\delta_n^j,\ j \in \{1, 2, \dots, n\}$.

It is obvious that the transition probability matrix of any BN is a logical matrix. Cheng et al. [5] proposed a mapping from $\{0, 1\}$ to set $\{\delta_2^1, \delta_2^2\}$ in the semi-tensor approach, then they can forbid using logical expressions in the coming steps. The mapping is defined as follows:

$$T \sim 1 \sim \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad F \sim 0 \sim \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

After giving the most important concepts in semi-tensor theory, we have one of the most important theorems in semi-tensor product theory.

Theorem 1: (Cheng et al. [5]) Given a logical function $f(p_1, p_2, \dots, p_r)$, there exist a unique 2×2^r matrix M_f, such that

$$f(p_1, p_2, \dots, p_r) = M_f p_1 p_2 \dots p_r.$$

Moreover, M_f is a logical function.

The following example shows us how to express the logical function into the algebraic form. Before the example, some of the most important matrices are given as follow:

Structure matrix	logical function	algebraic form
$M_c = \delta_2[1, 2, 2, 2]$	$f(p, q) = p \wedge q$	$f(p, q) = M_c pq$
$M_d = \delta_2[1, 1, 1, 2]$	$f(p, q) = p \vee q$	$f(p, q) = M_d pq$
$M_n = \delta_2[2, 1]$	$f(p) = \neg p$	$f(p) = M_n p$
$M_r = \delta_4[1, 4]$	$f(p) = p^2$	$f(p) = M_r p$
$W_{[2]} = \delta_4[1, 3, 2, 4]$	$f(p, q) = qp$	$f(p, q) = W_{[2]} pq$

Example 2: Consider the logical function

$$f(p, q, r) = (p \wedge \neg q) \vee (r \wedge p).$$

We shall rewrite it in the algebraic form and compute the structure matrix M_f. Here f can be expressed as follow:

$$\begin{aligned} f(p,q,r) &= (p \wedge \neg q) \vee (r \wedge p) \\ &= M_d M_c p M_n q M_c r p. \\ &= M_d M_c (I_2 \otimes M_n) pq M_c r p \\ &= M_d M_c (I_2 \otimes M_n)(I_4 \otimes M_c) pqrp \\ &= M_d M_c (I_2 \otimes M_n)(I_4 \otimes M_c) p W_{[2,4]} pqr \\ &= M_d M_c (I_2 \otimes M_n)(I_4 \otimes M_c)(I_2 \otimes W_{[2,4]}) p^2 qr \\ &= M_d M_c (I_2 \otimes M_n)(I_4 \otimes M_c)(I_2 \otimes W_{[2,4]}) M_r pqr \\ &:= M_f pqr \end{aligned}$$

Then, we have

$$\begin{aligned} M_f &= M_d M_c (I_2 \otimes M_n)(I_4 \otimes M_c)(I_2 \otimes W_{[2,4]}) M_r \\ &= \delta_2[1\ 2\ 1\ 1\ 2\ 2\ 2\ 2]. \end{aligned}$$

III. Methods

In this section, we shall introduce an important matrix T_n, which is defined as follows:

$$T_n = \delta_{2^n}[2^n, 2^n - 1, \dots, 1].$$

Proposition 1: We have $T_n^2 = I_{2^n}$

Proof: For any matrix A, we denote A_j the jth column of A. If $A' = A \times T$ then A' can be obtained from this way: $A'_j = A_{n-j}$, where n denotes the total columns in A. Thus obviously we have $T^2 = I_{2^n}$. ■

Given the definition of T_n, we can give the main theorem in this paper in the following. Assume there are n genes in the PBN, we can use T for T_n for simplicity.

Theorem 2: If we denote A the transition probability matrix of a PBN, and A_{semi} denote the probability structure matrix, then we have $A_{semi} = TAT$, where T is defined above. Since we know that $T^2 = I_{2^n}$, we also have $A = TA_{semi}T$. This means that A and A_{semi} are similar matrices.

Proof: First of all, we show that it is true for any BN. Suppose we have

$$A = \sum_{i=1}^{N} q_i A^i$$

q_i denote the probability choosing the ith BN. And if for any i,

$$A^i_{semi} = TA^iT$$

holds, then $A = TA_{semi}T$ is satisfied.

We need to prove that for each BN, $A^i_{semi} = TA^iT$ holds. We can infer from the definition of A^i and A^i_{semi} that their size are the same, namely, $2^n \times 2^n$. We denote the BN state at time t by $\mathbf{V}(t)$, and the BN state at time t obtained from the semi-tensor approach by $\mathbf{V}_{semi}(t)$. In order to show the equality, we have to find the relationship between $\mathbf{V}(t)$ and $\mathbf{V}_{semi}(t)$. In the semi-tensor product, there is a mapping from $\{0, 1\}$ to $\{\delta_2^1, \delta_2^2\}$, where $0 \sim \delta_2^2, 1 \sim \delta_2^1$. And we know that if there are n genes, then there are totally 2^n gene states. In the construction of A_i, a_{ij} denote whether the BN state j

978-1-4799-7294-4/14/$31.00 ©2014 IEEE

can be transferred to state i, where state 1 to state 2^n equals to $\underbrace{00\ldots0}_{2^n}$, $\underbrace{00\ldots1}_{n}$, $\ldots$, $\underbrace{11\ldots1}_{n}$, respectively. However, in the semi-tensor product, the dynamic of a BN can be expressed as $\mathbf{V}_{semi}(t+1) = A_{semi}\mathbf{V}_{semi}(t)$. Similar to A, A_{semi} is also a logical matrix. Thus $A_{semi}\delta_{2^n}^j$ equals to the jth column of A_{semi}. In the semi-tensor product, state 1 to state 2^n equal to $\underbrace{11\ldots1}_{n}$, $\underbrace{11\ldots0}_{n}$, $\ldots$, $\underbrace{00\ldots0}_{n}$. Hence $A_{semi}^i = TA^iT$ is proved. ■

The following example shows how our main theorem works.

Example 3: Consider the same PBN in Example 1, we try to solve the structure matrix and apply the above theorem to it. As for the semi-tensor approach, we need to solve out the logical equation for each BN first, it is easy to find out the logical equation for each BN, which is given by,

	f_1	f_2
BN_1	q	$\neg(p \vee q)$
BN_2	q	$\neg(p \wedge q)$
BN_3	p	$\neg(p \wedge q)$
BN_4	p	$\neg(p \vee q)$

Here p and q are logical variables. Next step we need to figure out $A_{semi}^i, i \in \{1,2,3,4\}$. Take $BN1$ as an example, we have

$$\begin{aligned} f(p,q) &= q \ltimes M_n(M_d pq) \\ &= (I_2 \otimes M_n)(I_2 \otimes M_d)qpq \\ &= (I_2 \otimes M_n)(I_2 \otimes M_d)W_{[2]}pq^2 \\ &= (I_2 \otimes M_n)(I_2 \otimes M_d)W_{[2]}(I_2 \otimes M_r)pq \end{aligned}$$

The structure matrix equals to $\delta_4[2,4,2,3]$. And $T_2 = \delta_4[4,3,2,1]$, we can easily confirm that $A_{semi}^1 = T_2A^1T_2$, similarly, we can prove $A = T_2A_{semi}T_2$.

IV. The Three Problems

A. Dynamics of a PBN

Suppose there are n genes and $l_i, i \in \{1,2,\ldots,n\}$ Boolean functions are assigned for gene v_i. So we have total $N = \prod_{i=1}^n l_i$ Boolean network, and q_i is the probability of choosing the ith BN. The dynamic of the PBN can be expressed as

$$\begin{cases} x_1(t) = \begin{cases} f_1^1(x_1(t),x_2(t),\ldots,x_n(t)) & \text{with probability} p_1^1 \\ f_1^2(x_1(t),x_2(t),\ldots,x_n(t)) & \text{with probability} p_1^2 \\ \vdots & \vdots \\ f_1^{l_1}(x_1(t),x_2(t),\ldots,x_n(t)) & \text{with probability} p_1^{l_1} \end{cases} \\ x_2(t) = \begin{cases} f_2^1(x_1(t,x_2(t),\ldots,x_n(t))) & \text{with probability} p_2^1 \\ f_2^2(x_1(t),x_2(t),\ldots,x_n(t)) & \text{with probability} p_2^2 \\ \vdots & \vdots \\ f_2^{l_2}(x_1(t),x_2(t),\ldots,x_n(t)) & \text{with probability} p_2^{l_2} \end{cases} \\ \vdots \quad \vdots \\ x_n(t) = \begin{cases} f_n^1(x_1(t,x_2(t),\ldots,x_n(t))) & \text{with probability} p_n^1 \\ f_1^2(x_1(t),x_2(t),\ldots,x_n(t)) & \text{with probability} p_n^2 \\ \vdots & \vdots \\ f_n^{l_n}(x_1(t),x_2(t),\ldots,x_n(t)) & \text{with probability} p_n^{l_n} \end{cases} \end{cases}$$

Here $x_i(t)$ denotes the ith gene state at time t and f_i^j is the jth Boolean function for gene i as we state in the introduction part, we have

$$\sum_{i=1}^{n}\sum_{j=1}^{l_i} p_i^j = 1.$$

We know that the dynamic of the PBN can be expressed as $\mathbf{V}(t+1) = A\mathbf{V}(t)$, and according to Theorem 2, we have $\mathbf{V}(t+1) = TA_{semi}T\mathbf{V}(t)$, it is obvious that $T\mathbf{V}(t) = \mathbf{V}_{semi}(t)$ (In a BN, $\mathbf{V}(t)$ is in the form of $\delta_{2^n}^j$, where n is the total number of genes and j means it is the jth states from $\underbrace{00\ldots0}_{n}$ to $\underbrace{11\ldots1}_{n}$).

Lemma 1: The dynamic of PBN using probability structure matrix can be expressed as

$$\mathbf{V}_{semi}(t+1) = A_{semi}\mathbf{V}_{semi}(t).$$

B. Steady State Analysis

It is known that there is a limitation of using BN to describe the real biological system. A PBN based on the fundamental idea of a BN can better capture the uncertainty characteristic of the biological system. And it has been found a PBN model is a stochastic process with the Markov property. Stationary distribution is an important factor in Markov Chain. Semi-tensor approach provides a new view for describing the PBN, therefore there may arise a lot of ways dealing with the steady state distribution problems based on the semi-tensor product approach. In this subsection, we shall find the relationship between the steady-state distribution with which is found based on the semi-tensor approach. A stationary distribution is defined as follows:

Definition 4: For a time-homogeneous Markov chain, which means that the Markov chain can be described by a single, time-independent matrix P. Then the stationary distribution $\pi = (\pi_1, \pi_2, \ldots, \pi_n)$ exists if the solution of the equation $P\pi = \pi$ subject to $\sum_{j=1}^n \pi_j = 1$ exists.

We remark that if the steady-state probability distribution of a PBN exists then it must be the stationary probability distribution but not vice versa. Thus if we denote π as the stationary distribution and π_{semi} as the stationary distribution regarding to probability structure matrix. It is natural that we can define π_{semi} as follows:

Definition 5: Noted that the PBN with A_{semi} as its probability transition matrix is a Markov chain, so we define $\pi_{semi} = (\pi_{semi}^1, \pi_{semi}^2, \ldots, \pi_{semi}^n)^T$ as the stationary distribution, which can be given by

$$\pi_{semi} = A_{semi}\pi_{semi}$$

subject to $\sum_{i=1}^n \pi_{semi}^i = 1$. Here A_{semi} is an $n \times n$ matrix. Then we know that $\pi = A\pi$, which means $\pi = TA_{semi}T\pi$, similarly, we can obtain $\pi = T\pi_{semi}$.

978-1-4799-7294-4/14/$31.00 ©2014 IEEE

Example 4: The transition probability matrix and probability structure matrix are given, respectively, by

$$A = \begin{pmatrix} 0.00 & 0.14 & 0.06 & 0.00 \\ 1.00 & 0.56 & 0.24 & 0.00 \\ 0.00 & 0.06 & 0.14 & 1.00 \\ 0.00 & 0.24 & 0.56 & 0.00 \end{pmatrix}$$

and

$$A_{semi} = \begin{pmatrix} 0.00 & 0.56 & 0.24 & 0.00 \\ 1.00 & 0.14 & 0.06 & 0.00 \\ 0.00 & 0.24 & 0.56 & 1.00 \\ 0.00 & 0.06 & 0.14 & 0.00 \end{pmatrix}.$$

By solving the equations:

$$A\pi = \pi \quad \text{and} \quad A_{semi}\pi_{semi} = \pi_{semi}$$

we have

$$\pi = (0.0667, 0.3333, 0.3333, 0.2667)^T$$

and

$$\pi_{semi} = (0.2667, 0.3333, 0.3333, 0.0667)^T$$

And it is obvious that

$$\pi = T_2\pi_{semi}$$

.

C. The Inverse Problem

The inverse problem is to find a appropriate PBN from a prescribed transition matrix [8], [9], [21]. Suppose A is the given transition probability matrix. Suppose there are l_i nonzero elements in each column, then we have $\prod_{i=1}^{n} l_i$ feasible BNs at most, they are labeled as $BN1, BN2, \ldots, BN_m$, where $m = \Pi_{i=1}^{n} l_i$. And the transition matrix assigned to BN_i is A_i. Thus the inverse problem can be expressed in the form of finding the appropriate set of q_j minimize the following function:

$$f(q_1, q_2, \ldots, q_m) = \left\| \sum_{i=1}^{m} A_i q_i - A \right\|_2^2$$

subject to $\sum_{i=1}^{m} q_i = 1$.

Similarly, we can define the inverse problem of constructing a PBN by semi-tensor product approach.

Definition 6: Given the probability structure matrix as A_{semi}, we try to find an appropriate $q_{semi} = (q^1_{semi}, q^2_{semi}, \ldots)$ and A^i_{semi} according to q^i_{semi}, such that $\sum_i q^i_{semi} A^i_{semi} = A_{semi}$ subject to $\sum_i q^i_{semi} = 1$.

According to the previous definition of q^i_{semi}, it can be easily obtained that $q^i_{semi} = q_i$, which means that all the algorithms solving the inverse problem can be applied to the PBN regarding to a given probability structure matrix A_{semi}.

Example 5: We construct the transition probability matrix and the probability structure matrix as follows:

$$A = \begin{pmatrix} 0.15 & 0.30 & 0.00 & 0.23 \\ 0.85 & 0.00 & 0.50 & 0.00 \\ 0.00 & 0.70 & 0.50 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.77 \end{pmatrix}$$

$$A_{semi} = \begin{pmatrix} 0.77 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.50 & 0.70 & 0.00 \\ 0.00 & 0.50 & 0.00 & 0.85 \\ 0.23 & 0.00 & 0.30 & 0.15 \end{pmatrix}$$

And we use the projection based gradient method to compute the inverse problem [20]. Thus the problem becomes a least squares problem:

$$\min_{q} \|Uq - p\|_2^2 \quad \text{s.t. } \|q\|_1 = 1 \qquad \text{and} \qquad \min_{q_{semi}} \|U_{semi} q_{semi} - p_{semi}\|_2^2 \quad \text{s.t. } \|q_{semi}\|_1 = 1.$$

Here $\|q_{semi}\|_1$ means the L_1 norm of q_{semi} and the elements of q and q_{semi} are nonnegative. Here $p = F(A)$ and $U = [F(A_1), F(A_2), \ldots, F(A_N)]$, where for any matrix B,

$$F(B) = (b_{11}, b_{12}, \ldots, b_{1n}, b_{21}, b_{22}, \ldots, b_{2n}, \ldots, b_{n1}, b_{n2}, \ldots, b_{nn})^T$$

b_{ij} is the (i, j)th element in B and the same definition holds for U_{semi} and p_{semi}. From the given A and A_{semi}, the value of U, U_{semi}, p, p_{semi} are listed as

$$U = \begin{pmatrix} 1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&1&1&1&0&0&0&0&1&1&1&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0 \\ 0&0&1&1&0&0&1&1&0&0&1&1&0&0&1&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1 \end{pmatrix}$$

$$U_{semi} = \begin{pmatrix} 0&1&0&1&0&1&0&1&0&1&0&1&0&1&0&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&0&1&0&1&0&1&0&1&0&1&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&1&0&0&1&1&0&0&1&1&0&0&1&1 \\ 1&1&0&0&1&1&0&0&1&1&0&0&1&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&1&1&1&0&0&0&0&1&1&1&1 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&1&1&1&0&0&0&0&1&1&1&1&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0 \end{pmatrix}$$

$$p = [0.15, 0.85, 0, 0, 0.3, 0, 0.7, 0, 0, 0.5, 0.5, 0, 0.23, 0, 0, 0.77]^T$$
$$p_{semi} = [0.77, 0, 0, 0.23, 0, 0.5, 0.5, 0, 0, 0.7, 0, 0.3, 0, 0, 0.85, 0.15]^T$$

Using the projection-based gradient algorithm, we have

$$q = q_{semi} = [0.1506, 0, 0, 0, 0, 0, 0, 0, 0.0203, 0.0394, 0.0039, 0.0857, 0.0222, 0.2674, 0.0329, 0.3776]^T$$

The graph of the distributions of q and q_{semi} are shown in the following two figures. We can infer from the graphs that they are exactly same.

978-1-4799-7294-4/14/$31.00 ©2014 IEEE

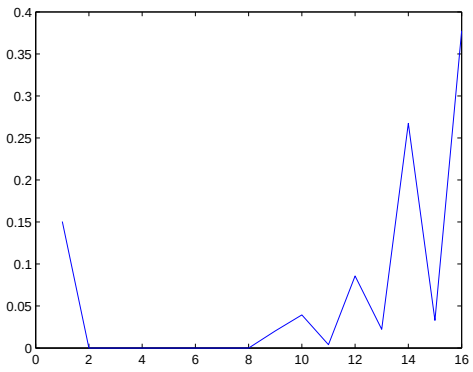

Fig. 1. The Distribution of q

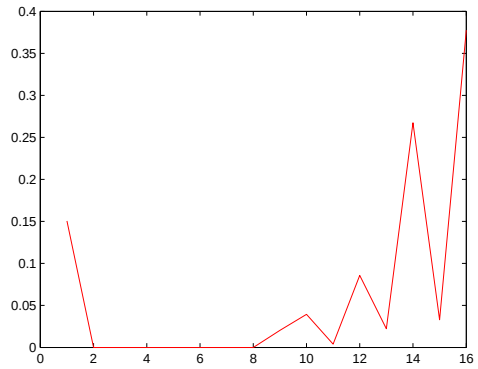

Fig. 2. The Distribution of q_{semi}

V. Conclusions

This paper studies PBNs by using semi-tensor approach. We show the relationship between the probability transition matrix and probability structure matrix. Various of algorithms have been developed to solve the BN problems through semi-tensor approach. This gives a broader fields of vision to dealing with the PBN problems. As we have shown in previous sections, the PBN built from semi-tensor approach and the original one are " equivalent", by "equivalent" here we means they can be transformed to each other under all conditions and they share many same properties. The reason is because A and A_{semi} are similar matrices. Thus all the theories, algorithms for the original PBN can be used for the PBN generalized from semi-tensor approach. And the time complexity and sample complexity for solving the PBN generated from semi-tensor approach is at least no worse than that of the original PBN. The theories for BN generated from semi-tensor approach can be applied to the original BN transition matrix. For example, the theory about singleton attractor. It also provides evidence that the semi-tensor theory and the original theory about BN (or PBN) are equivalent.

Acknowledgment

The authors would like to thank the referees and the editor for their helpful comments and suggestions. This research work was supported by the Research Grants Council of Hong Kong, under grant No. 17301214 and HKU Hung Hing Ying Physical Research Grant and National Natural Science Foundation of China Grant Nos. 10971075 and S201201009985.

References

[1] R. Albert and A. Barabasi, *Dynamics of complex systems: scaling laws for the period of Boolean networks*, Phys. Rev. Lett, 2000, 84, 5660.
[2] M. Aldana, *Boolean dynamics with scale-free topology*, Physica D, 2003, 185 (1), 45–66.
[3] D. Cheng and H. Qi, *Controllability and observability of Boolean control networks*, Automatica, 2009, 45 (7), 1659–1667.
[4] D. Cheng and H. Qi, *A linear representation of dynamics of Boolean networks*, IEEE Trans. Auto. Contr., 2010, 55 (10), 2251–2258.
[5] D. Cheng, H. Qi and Z. Li, *Analysis and control of Boolean networks: a semi-tensor approach*, Springer, 2011.
[6] D. Cheng, H. Qi and Z. Li, *Model construction of Boolean networks via observed data*, IEEE Trans. Neural Networks, 2011, 22 (4), 525–535.
[7] J. Celis, M. Kruhofferm, I. Gromova, C. Frederiksen, M. Ostergaard and T. Orntoft, *Gene expression profiling: Monitoring transcription and translation products using DNA microarrays and proteomics*, FEBS Letters, 2000, 480(1), 2–16.
[8] X. Chen, W. Ching, X.S. Chen, Y. Cong and N. Tsing, *Construction of probabilistic Boolean networks from a prescribed transition probability matrix: A maximum entropy rate approach*, East Asian Journal of Applied Mathematics, 2011, 1, 132–154.
[9] W. Ching, X. Chen and N. Tsing, *Generating probabilistic Boolean networks from a prescribed transition probability matrix*, IET Systems Biology, 2009, 6 453–464.
[10] D. Drossel, T. Mihaljev, F. Greil, *Number and length of attractors in a critical Kauffman model with connectivity one*, Phys. Rev. Lett., 2005, 94, 088071.
[11] S. Huang, *Gene expression profiling, genetic networks, and cellular states: an integrating concept for Tumorigenesis and drug discovery*, Journal of Molecular Medicine, 1999, 77, 469–480.
[12] *The origins of order: self-organization and selection in evolution*, Oxford University Press, New York, 1993.
[13] *Metabolic stability and epigenesis in randomly constructed genetic nets*, J. Theor. Biol., 1969, 22 (3) 437–467.
[14] T. Mestl, E. Plahte and S. Omholt, *A mathematical frame work for describing and analysing gene regulatory networks*, J. Theor. Biol., 1995, 176 (2), 291–300.
[15] H. Mortveit and C. Reidays, *An introduction to sequential dynamic systems*, Springer, 2008.
[16] K. Murphy and S. Mian, *Modelling gene expression data using dynamic Bayesian networks*, available at http://www-devel.cs.ubc.ca/∼murphyk/Papers/ismb99.pdf, 1999.
[17] I. Shmulevich, E. Dougherty, S. Kim and W. Zhang, *Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory network*, Bioinformatics, 2002, 18 (2) 261–274.
[18] I. Shmulevich and E. Dougherty, *Probabilistic Boolean networks: The modeling and control of gene regulatory networks*, SIAM Press, 2009.
[19] E. van Someren, L. Wessels and M. Reinder, *Linear modeling of genetic networks from experimental data*, ISMB, San Digeo, CA, August 19-23, 2000.
[20] Y. Wen, Z. Zhang, X. Cheng, W. Ching and V. Vassiliadis, *Sparse solution of non-negative Least Squares problems with projection with application in the construction of probabilistic Boolean networks*, submitted, 2014.
[21] S. Zhang, W. Ching, X. Chen and N. Tsing, *Generating probabilistic Boolean networks from a prescribed stationary distribution*, Information Sciences, 2010, 180, 2560–2570.
[22] M. Yang, R. Li, T. Chu, *Optimal control of steady-state probability distributions of probabilistic Boolean network*, Proceedings of the 32nd Chinese Control Conference, July 26-28, Xi'an, China.
[23] F. Li, J. Sun *Stanility and stabilization issue of probabilistic Boolean network* Proceedings of the 30th Chinese Control Conference, July 22-24, 2011, Yantai, China.

978-1-4799-7294-4/14/$31.00 ©2014 IEEE