Journal of Computational and Applied Mathematics 236 (2012) 4067–4081

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

# On computation of the steady-state probability distribution of probabilistic Boolean networks with gene perturbation

Wen Li[a], Lu-Bin Cui[a], Michael K. Ng[b,*]

[a] School of Mathematical Sciences, South China Normal University, Guangzhou, 510631, China
[b] Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong

ARTICLE INFO

*Article history:*
Received 22 November 2011
Received in revised form 13 February 2012

In memory of Donato Trigiante

*Keywords:*
Structured matrices
Probabilistic Boolean networks
Steady-state probability distribution
Perturbation bound
Iterative methods

ABSTRACT

Given a Probabilistic Boolean Network (PBN), an important problem is to study its steady-state probability distribution for network analysis. In this paper, we present a new perturbation bound of the steady-state probability distribution of PBNs with gene perturbation. The main contribution of our results is that this new bound is established without additional condition required by the existing method. The other contribution of this paper is to propose a fast algorithm based on the special structure of a transition probability matrix of PBNs with gene perturbation to compute its steady-state probability distribution. Experimental results are given to demonstrate the effectiveness of the new bound, and the efficiency of the proposed method.

## 1. Introduction

There has been considerable interest in genomic signal processing recently. Since regulatory decisions within cells utilize numerous inputs, analytical tools are necessary to model multivariate influences on decision-making produced by complex genetic networks. Mathematical modeling and computational study of regulatory interactions between DNA, RNA, proteins and small molecules based on the microarray data are hot topics in bioinformatics and have been studied by a number of researchers [1–3]. There have been many formalisms proposed in the literature to study genetic regulatory networks such as directed graphs, Boolean Networks (BNs) [4,5], Probabilistic Boolean Networks (PBNs) [6,7], multivariate Markov chain models [8–10] and many other mathematical models [11]. Among these models, BNs and PBNs (an extension of BNs) have attracted much attention.

BN was first introduced by Kauffman in [12,13]. We remark that a BN is a deterministic model. Due to the fact that a genetic regulation process exhibits an uncertainty property and microarray data sets have errors due to experimental noise in the complex measurement process, BNs have been extended to PBNs (stochastic models). The network dynamics of a PBN can be studied in a Markov chain framework [6]. Owing to this, the rich theory and numerous tools developed for Markov chains are applicable to the analysis of PBNs as well. PBNs also provide a natural way to quantify the relative influence and sensitivity of genes in their interactions with other genes. Random gene perturbations are introduced into the PBN model in [7], where the perturbation describes random inputs to the network. The effect of introducing random gene perturbations is to make a network stable in the long run. A review on BNs and PBNs can be found in [14,15] and in Section 2, we shall give a brief review of BNs, PBNs and PBNs with gene perturbations.

Given a PBN, an important problem is to study its steady-state probability distribution. It provides the first-order statistical information of a PBN. Based on such information of a PBN, one can understand a genetic network, and identify

[*] Corresponding author.
*E-mail addresses:* liwen@scnu.edu.cn (W. Li), hnzkc@163.com (L.-B. Cui), mng@math.hkbu.edu.hk, 10466231@hkbu.edu.hk (M.K. Ng).

0377-0427/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2012.02.022

the influence of different genes in a network [16,17]. Furthermore, one can figure out how to control some genes in a network, such that the whole system can evolve into a target state or a desirable steady-state probability distribution [18]. In [19], efficient algorithms has been proposed for constructing a sparse probabilistic Boolean network using the steady-state probability distribution.

It is well-known that in Markov chain theory, if a Markov chain is irreducible and aperiodic, the steady-state probability distribution exist and is independent of the initial condition. We remark that in a PBN with random gene perturbations, the underlying transition probability matrix can be shown to be irreducible and aperiodic. In [20], a matrix-based method has been proposed for computing the steady-state probability distribution. Recently, an efficient matrix approximation method is proposed to get the steady-state distribution of a PBN [10]. It is known that the size of a transition probability matrix of a PBN is very huge, $2^n \times 2^n$. Here $n$ refers to the number of genes. In the literature, existing computational methods [17] can handle $n$ to be around 15. Therefore, it is necessary to develop fast algorithms for computing such a steady-state probability distribution of a PBN. The main contribution of this paper is to develop a fast algorithm based on the special structure of transition probability matrix of PBNs with gene perturbation to compute the steady-state probability distribution. Our numerical results show that we can compute the steady-state probability distribution of a genetic network of size $n$ to be 30 within a reasonable time on a desktop computer using MATLAB implementation.

On the other hand, in [7], the sensitivity of the steady-state probability distribution has been successfully analyzed based on an effective construction of the transition probability matrix of a PBN with random perturbation. Recently, Xu et al. [21] proposed a perturbation bound for the steady-state probability distribution of a PBN with gene perturbations. However, this perturbation bound is not practical. The second contribution of this paper is to present a new perturbation bound of the steady-state probability distribution of PBNs with gene perturbation without any additional condition.

The rest of the paper is organized as follows. In Section 2, a brief review of PBNs is presented. In Section 3, we give a new perturbation bound of the steady-state probability distribution. In Section 4, we present a fast algorithm for computing the steady-state probability distribution of PBNs with gene perturbations. In Section 5, numerical experiments are given to demonstrate the effectiveness of the proposed algorithm. Finally, some concluding remarks are given in Section 6.

## 2. The review

### 2.1. Boolean networks

In this subsection, we give a brief review of Boolean Networks (BNs). A BN $G(V, F)$ consists of a set of nodes $V$ and Boolean functions $F$ where

$$V = \{v_1, v_2, \dots, v_n\} \quad \text{and} \quad F = \{f_1, f_2, \dots, f_n\}.$$

Let $v_k(t)$ be the state of $v_k$ at time $t$, where $v_k = 0$ represents that the gene is unexpressed and $v_k = 1$ means it is expressed. The expression levels of all the genes in the network at the time $t$ is given by the following column vector

$$v(t) = [v_1(t), v_2(t), \dots, v_n(t)]^T.$$

This vector is called the Gene Activity Profile (GAP) of the network at time $t$. We note that when $v(t)$ ranges from $[0, 0, \dots, 0]^T$ to $[1, 1, \dots, 1]^T$, it takes on all the $2^n$ possible states of the $n$ genes. The list of Boolean functions represents the rules of the regulatory interactions among the nodes (genes):

$$v_k(t + 1) = f_k(v(t)), \quad k = 1, 2, \dots, n.$$

Here each gene will update its state according to the states of its input genes in the previous step and its corresponding Boolean function. Thus, a BN is a deterministic dynamical system.

**Example 2.1.** Suppose we are given a BN consisting of two genes $V = (v_1, v_2)$ and the function set $F = f_1, f_2$. The Boolean functions are given in Table 1.

The transition probability matrix is given by the following Boolean matrix:

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

### 2.2. Probabilistic Boolean networks

In a Probabilistic Boolean Network (PBN), for each target gene, it has a number of Boolean functions having equivalent prediction abilities. All these Boolean functions can be selected randomly with some probabilities. We assume that for the $k$th gene, there are $l(k)$ possible Boolean functions:

$$F^{(k)} = \{f_j^{(k)} : \text{ for } j = 1, \dots, l(k)\}$$

**Table 1**
The truth table.

| State | $v_1(t)$ | $v_2(t)$ | $f_{(1)}$ | $f_{(2)}$ |
|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 |
| 2 | 0 | 1 | 1 | 0 |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 1 | 1 | 1 | 0 |

**Table 2**
The truth table of the PBN.

| State | $v_1v_2v_3$ | $f_1^{(1)}$ | $f_2^{(1)}$ | $f_1^{(2)}$ | $f_1^{(3)}$ | $f_2^{(3)}$ |
|---|---|---|---|---|---|---|
| 1 | 000 | 0 | 0 | 0 | 0 | 0 |
| 2 | 001 | 1 | 1 | 1 | 0 | 0 |
| 3 | 010 | 1 | 1 | 1 | 0 | 0 |
| 4 | 011 | 1 | 0 | 0 | 1 | 0 |
| 5 | 100 | 0 | 0 | 1 | 0 | 0 |
| 6 | 101 | 1 | 1 | 1 | 1 | 0 |
| 7 | 110 | 1 | 1 | 0 | 1 | 0 |
| 8 | 111 | 1 | 1 | 1 | 1 | 1 |
|  | $c_j^{(i)}$ | 0.6 | 0.4 | 1 | 0.5 | 0.5 |

and the probability of choosing function $f_j^{(k)}$ is $c_j^{(k)}$, where $f_j^{(k)}$ is a function with respect to the activity levels of $n$ genes. A PBN is said to be independent if the elements from different $F^{(k)}$ are independent. For an independent PBN of $n$ genes, there are at most

$$N = \prod_{k=1}^{n} l(k) \tag{2.1}$$

different possible BNs. This means that there are totally $N$ possible realizations of the genetic network. Suppose $f_j$ is the $j$th possible realization,

$$f_j = [f_{j_1}^{(1)}, f_{j_2}^{(2)}, \dots, f_{j_n}^{(n)}], \quad 1 \le j_k \le l(k),\ k = 1, 2, \dots, n.$$

The probability to choose the $j$th realization is given by

$$p_j = \prod_{k=1}^{n} c_{j_k}^{(k)}, \quad j = 1, 2, \dots, N. \tag{2.2}$$

If the joint probability distribution of $F^{(1)}, F^{(2)}, \dots, F^{(n)}$ cannot be factorized as the product of $F^{(k)}$, then it is a dependent PBN. For a dependent PBN, one can still use the same notations as those for independent PBNs.

Let $\mathbf{a}$ and $\mathbf{b}$ be any two column vectors with $n$ entries being either 0 or 1, which represent the states of the system at time $t + 1$ and $t$. Then we have

$$Prob\{v(t+1) = \mathbf{a} | v(t) = \mathbf{b}\} = \sum_{j=1}^{n} Prob\{v(t+1) = \mathbf{a} | v(t) = \mathbf{b},\ \text{the } j\text{th BN is selected}\} \cdot p_j. \tag{2.3}$$

Letting $\mathbf{a}$ and $\mathbf{b}$ range from $[0, 0, \dots, 0]^T$ to $[1, 1, \dots, 1]^T$, one can get the transition probability matrix $A$ with size $2^n \times 2^n$. It can be expressed as:

$$A = \sum_{j=1}^{N} p_j A_j$$

where $A_j$ is the transition matrix corresponding to the $j$th BN.

**Example 2.2** ([6]). Suppose we are given a PBN consisting of three genes $V = (v_1, v_2, v_3)$ and the function sets $F^{(1)} = \{f_1^{(1)}, f_2^{(1)}\}$, $F^{(2)} = \{f_1^{(1)}\}$ and $F^{(3)} = \{f_1^{(3)}, f_2^{(3)}\}$. Let the functions be given in Table 2. The state transition probability matrix is then given by

$$A = \begin{pmatrix} 1 & 0 & 0 & 0.2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.3 & 0 & 0 & 0.5 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 1 \end{pmatrix}.$$

### 2.3. PBNs with gene perturbations

In this subsection, we recall the PBN model with gene perturbations [7]. Random gene perturbation is the description of the random input from the outside due to external stimuli. This is meaningful in an open genome system. The effect of random gene perturbations is to make genes flip from state 1 to state 0 or vice versa. It makes the underlying Markov chain of the PBN ergodic, and therefore all the $2^n$ states in the system are communicated. When random gene perturbation is included, the transition probability matrix $\tilde{A}$ is

$$\tilde{A} = (1-p)^n A + \tilde{P}_n \equiv \hat{A} + \tilde{P}_n, \tag{2.4}$$

where $\tilde{P}_n$ is the perturbation matrix (see [7]). Xu et al. [21] studied the properties of such perturbation matrix, and gave a useful form of the perturbation matrix.

**Theorem 2.1** ([21]). *Let $\tilde{P}_n$ be the $2^n \times 2^n$ perturbation matrix of a PBN with $n$ genes, then we have for $n = 1, 2, \ldots$*

$$\tilde{P}_n = Q_n - (1-p)^n I_{2^n}, \tag{2.5}$$

*where*

$$Q_n = \underbrace{Q_1 \otimes Q_1 \otimes \cdots \otimes Q_1}_{n\ terms}$$

$$Q_1 = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}.$$

*Here $\otimes$ refers to the Kronecker product of two matrices.*

From (2.5), we see that the transition matrix $\tilde{A}$ is the sum of the transition matrix without perturbation $A$ multiplied by $(1-p)^n$ and the perturbation matrix $\tilde{P}_n$. We know the perturbation matrix $\tilde{P}_n$ depends on the number of genes and the random gene perturbation probability. When the number of genes and the gene perturbation probability in different PBNs are the same, the perturbation matrices are the same. If the perturbation probability $p = 0$, then $\tilde{A} = A$. If $p = 1$, then $\tilde{A} = \mathbf{0}$, which is of no significance. Hence we always assume that $0 < p < 1$ throughout the paper without further illustration.

According to Theorem 2.1, we have

$$\sigma_{\max}(\tilde{P}_n) = 1 - (1-p)^n, \tag{2.6}$$

where $\sigma_{\max}(\cdot)$ denotes the largest singular value of a matrix.

## 3. The new perturbation bound

A matrix $A$ is called nonnegative, semi-positive and positive if each entry of $A$ is nonnegative, nonnegative but at least a positive entry and positive, respectively. We denote them respectively by $A \geq 0$, $A > 0$ and $A \gg 0$.

**Definition 3.1** ([22]). A real $n \times n$ matrix $A = [a_{i,j}]$ with $a_{ij} \leq 0$ for all $i \neq j$ is an nonsingular $M$-matrix if $A = sI - B$, $B \geq 0$ and $s > \rho(B)$; a Stieltjes matrix if $A$ is symmetric and positive definite.

**Lemma 3.1** ([23,22]).

(1) *If $A$ is a Stieltjes matrix, then it is also a nonsingular $M$-matrix.*
(2) *A matrix $A$ is a nonsingular irreducible $M$-matrix (for the definition of an irreducible matrix, e.g., see* [23]*) if and only if $A^{-1} \gg 0$.*
(3) *If $A$ is a singular irreducible $M$-matrix, then each principal submatrix of $A$ other than $A$ itself is a nonsingular $M$-matrix.*

In this section, we give a new perturbation bound for the steady-state probability distributions of PBNs. Let $x$ and $\tilde{x}$ be the steady-state probability distribution of $A$ and $\tilde{A}$, the transition probability matrix with and without gene perturbation, respectively, i.e., $Ax = x$ and $\tilde{A}\tilde{x} = \tilde{x}$.

Let $H = (1-p)^n A - I$. Since $A$ is a transition probability matrix, 1 is the spectral radius of $A$, and thus $H$ is nonsingular for any nonzero $p$. This implies that

$$\sigma^+_{\min}(H) = \sigma_{\min}(H),$$

where $\sigma^+_{\min}(\cdot)$ and $\sigma_{\min}(\cdot)$ denote the smallest positive singular value and smallest singular value of a matrix respectively. Xu et al. in [21] presented a bound for $\|x - \tilde{x}\|$ under the assumption that

$$\sigma^+_{\min}(H) \geq \sigma_{\max}(\tilde{P}_n).$$

However, this condition is not practical by the following property.

**Property 3.1.** *In the notation above, then we always have*

$$\sigma_{\min}^{+}(H) \le \sigma_{\max}(\tilde{P}_n),$$

*for all* $0 < p \le 1$.

**Proof.** It is noted that $-H = I - (1-p)^n A$ is a nonsingular matrix with $\rho((1-p)^n A) < 1$, where $\rho(\cdot)$ denotes the spectral radius of a matrix. Hence

$$\begin{aligned}(-H)^{-1} &= (I - (1-p)^n A)^{-1} \\ &= I + (1-p)^n A + (1-p)^{2n} A^2 + \cdots.\end{aligned}$$

Since $A$ is a nonnegative matrix with the spectral radius 1 (noting that the spectral radius of a nonnegative matrix is equal to its maximum eigenvalue), we have

$$\begin{aligned}\rho((-H)^{-1}) &= 1 + (1-p)^n + (1-p)^{2n} + \cdots \\ &= (1 - (1-p)^n)^{-1}.\end{aligned} \tag{3.1}$$

Clearly,

$$\rho((-H)^{-1}) \le \|(-H)^{-1}\|_2 = 1/\sigma_{\min}(H),$$

which together with (3.1) gives that

$$\sigma_{\min}(H) \le 1 - (1-p)^n.$$

Then the result follows from (2.6). □

Next we show the perturbation bound without assuming an additional condition. Let $B \in \mathbb{R}^{n\times n}$. $B_{i:j}$ $(1 \le i < j \le n)$ and $B(i)$ denote the submatrix of $B$ whose rows are taken from $i$th to $j$th of $B$ and the submatrix of $B$ by deleting the $i$th row of $B$, respectively. Let $P^{(i)}$ be a $(2^n+1) \times (2^n+1)$ matrix with the following form:

$$P^{(i)} = \begin{pmatrix} P_1^{(i)} \\ \vdots \\ P_{2^n}^{(i)} \\ P_{2^n+1}^{(i)} \end{pmatrix}, \quad i = 1, 2, \ldots, 2^n,$$

where

$$P_k^{(i)} = \begin{cases} (1, \ldots, 1, 0), & k = i; \\ (\overbrace{0, \ldots, 0, 1}^{k}, 0, \ldots, 0), & \text{otherwise}. \end{cases}$$

Let $\mathbf{1}_{2^n} = (1, \ldots, 1)$ and $\mathbf{0}_{2^n} = (0, \ldots, 0)$ be row vectors with dimension $2^n$. By (2.4) and (2.5) we have $\tilde{A} = \hat{A} + Q_n - (1+p)^n I$. Let

$$H_i = P^{(i)} \begin{pmatrix} I - \tilde{A} \\ \mathbf{1}_{2^n} \end{pmatrix}, \qquad K_i = P^{(i)} \begin{pmatrix} Q_n - I \\ \mathbf{0}_{2^n} \end{pmatrix}, \quad i = 1, 2, \ldots, 2^n. \tag{3.2}$$

**Lemma 3.2.** *The matrix* $\begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_{2^n} \end{pmatrix}$ *is nonsingular for* $i = 1, \ldots, 2^n$.

**Proof.** Otherwise, there is a nonzero vector $u \in R^{2^n}$ such that

$$\begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_{2^n} \end{pmatrix} u = 0. \tag{3.3}$$

Without loss of generality we assume that $i = 2^n$. Let $\tilde{A}_{2^n}$ be a principal submatrix of $\tilde{A}$ by deleting row $i$ and column $i$ of $\tilde{A}$. Then $(I - \tilde{A})(i) = (I - \tilde{A}_{2^n}, \alpha)$, where $\alpha$ is a nonpositive vector in $R^{2^n-1}$. Since $Q_n$ is positive, $\alpha \ne 0$. Let $u = (\widetilde{u}^T, u_{2^n})^T$. Then by (3.3) we have

$$(I - \tilde{A}_{2^n})\widetilde{u} + u_{2^n}\alpha = 0, \tag{3.4}$$

$$\mathbf{1}_{2^n-1}\widetilde{u} + u_{2^n} = 0. \tag{3.5}$$

Since $\tilde{A}$ is an irreducible and positive transition probability matrix, $I - \tilde{A}$ is a singular irreducible $M$-matrix. So, by Lemma 3.1, $I - \tilde{A}_{2^n}$ is a nonsingular $M$-matrix. By (3.4) we have

$$\widetilde{u} = -u_{2^n}(I - \tilde{A}_{2^n})^{-1}\alpha,$$

which is substituted to (3.5) gives

$$(1 - \mathbf{1}_{2^n-1}(I - \tilde{A}_{2^n})^{-1}\alpha)u_{2^n} = 0.$$

Assume that $u_{2^n} = 0$, then $\widetilde{u} = -u_{2^n}(I - \tilde{A}_{2^n})^{-1}\alpha = 0$, which contradicts that $u \neq 0$. Hence $u_{2^n} \neq 0$. This implies that $1 - \mathbf{1}_{2^n-1}(I - \tilde{A}_{2^n})^{-1}\alpha = 0$. However $\mathbf{1}_{2^n-1}(I - \tilde{A}_{2^n})^{-1}\alpha \leq 0$, and hence $1 - \mathbf{1}_{2^n-1}(I - \tilde{A}_{2^n})^{-1}\alpha \geq 1$, which is a contradiction. This proves the lemma. □

Now we present a bound between steady-state probability distributions of $A$ and $\tilde{A}$.

**Theorem 3.3.** *For an ergodic PBN of $n$ genes, let $x$ and $\tilde{x}$ be the steady-state probability distributions of the PBN and the PBN with gene perturbation, respectively. Then*

$$\|x - \tilde{x}\|_2 \leq \min_{1\leq i\leq 2^n}\{\sigma_{\max}(K_i)/\sigma^{+}_{\min}(H_i)\}\|x\|_2, \tag{3.6}$$

*where $H_i$ and $K_i$ are given by (3.2).*

**Proof.** The Markov chain is ergodic if and only if its transition probability matrix is irreducible and aperiodic. Then the eigenvalue 1 of the transition probability matrix is single. So the geometric multiplicity of eigenvector $x$ corresponding to eigenvalue 1 is one. Since the vector $x$ is the steady-state distribution of $A$, we have $\sum_{i=1}^{2^n} x_i = 1$. Then the following linear systems (3.7) and (3.8) have a unique solution $x$ and $\tilde{x}$, respectively.

$$\begin{cases} Ax = x, \\ \sum_{i=1}^{2^n} x_i = 1, \end{cases} \tag{3.7}$$

$$\begin{cases} \tilde{A}\tilde{x} = \tilde{x}, \\ \sum_{i=1}^{2^n} \tilde{x}_i = 1. \end{cases} \tag{3.8}$$

Let $\Delta x = \tilde{x} - x$. Then we have

$$\begin{aligned}(I - \tilde{A})\Delta x &= (Q_n + (1-p)^n(A - I) - I)x \\ &= (Q_n - I)x.\end{aligned}$$

On the other hand, by (3.7) and (3.8) we have

$$\sum_{i=1}^{2^n} \tilde{x}_i = \sum_{i=1}^{2^n} x_i + \sum_{i=1}^{2^n} \Delta x_i = 1,$$

thus

$$\sum_{i=1}^{2^n} \Delta x_i = 0.$$

Therefore, we obtain the following linear system:

$$\begin{pmatrix} I_{2^n} - \tilde{A} \\ \mathbf{1_2^n} \end{pmatrix} \Delta x = \begin{pmatrix} Q_n - I \\ \mathbf{0}_{2^n} \end{pmatrix} x. \tag{3.9}$$

Because $\tilde{A}$ is a transition probability matrix, the summation of every column of $\tilde{A}$ is one, and so is $Q_n$. By the definition of $H_i$ and $K_i$ we have

$$H_i = P_i \begin{pmatrix} I - \tilde{A} \\ \mathbf{1_2^n} \end{pmatrix} = \begin{pmatrix} (I - \tilde{A})_{1:i-1} \\ \mathbf{0}_{2^n} \\ (I - \tilde{A})_{i+1:2^n} \\ \mathbf{1_2^n} \end{pmatrix}$$

and

$$K_i = P_i \begin{pmatrix} Q_n - I \\ \mathbf{0}_{2^n} \end{pmatrix} = \begin{pmatrix} (Q_n - I)_{1:i-1} \\ \mathbf{0}_{2^n} \\ (Q_n - I)_{i+1:2^n} \\ \mathbf{0}_{2^n} \end{pmatrix}.$$

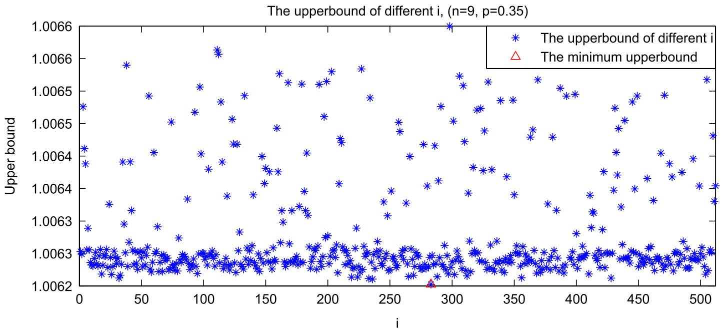

**Fig. 1.** The upper bounds for different $i$.

It is easy to see that $P_i$ is a nonsingular matrix. Left-multiplying by $P_i$ on the Eq. (3.9) leads to the following equivalent equation:

$$H_i\Delta x = K_i x.$$

Deleting the $i$th row of both matrices $H_i$ and $K_i$ gives the following equation:

$$\bar{H}_i\Delta x = \bar{K}_i x, \tag{3.10}$$

where $\bar{H}_i$ and $\bar{K}_i$ are square matrices with the block form:

$$\bar{H}_i = \begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_2^{\mathbf{n}} \end{pmatrix} \quad \text{and} \quad \bar{K}_i = \begin{pmatrix} (Q_n - I)(i) \\ \mathbf{0}_{2^n} \end{pmatrix}.$$

By Lemma 3.2 $\bar{H}_i$ is nonsingular, and hence

$$\Delta x = \bar{H}_i^{-1}\bar{K}_i x.$$

Taking the 2-norm in the both side of the above equality gives

$$\|\Delta x\|_2 \le \|\bar{H}_i^{-1}\bar{K}_i\|_2\|x\|_2 \le \|\bar{H}_i^{-1}\|_2\|\bar{K}_i\|_2\|x\|_2 = \sigma_{\max}(\bar{K}_i)/\sigma_{\min}(\bar{H}_i)\|x\|_2.$$

Let $\Theta(H_i)$ and $\Theta(\bar{H}_i)$ denote the singular values set of $H_i$ and $\bar{H}_i$, respectively. Then we get $\Theta(H_i) = \Theta(\bar{H}_i) \cup \{0\}$, implying

$$\sigma_{\min}(\bar{H}_i) = \sigma_{\min}^{+}(H_i).$$

By the definition of $\bar{K}_i$ we have $\sigma_{\max}(\bar{K}_i) = \sigma_{\max}(K_i)$. Hence

$$\|x - \tilde{x}\|_2 \le \min_{1\le i\le 2^n}\{\sigma_{\max}(K_i)/\sigma_{\min}^{+}(H_i)\}\|x\|_2.$$

This proves the theorem. □

**Remark 3.1.** By using Example 2.2 of $n = 9$, we consider the gene perturbation probability to be $p = 0.35$ and $p = 0.01$. In Figs. 1 and 2 we show the upper bounds of the perturbation bound in Theorem 3:

$$\sigma_{\max}(K_i)/\sigma_{\min}^{+}(H_i)\|x\|_2, \quad i = 1, \ldots, 2^n.$$

We see from the figures that these upper bounds are concentrated on their minimum value. Also the difference between their maximum and minimum values is very small, namely, $3.9 \times 10^{-4}$ and $3.2 \times 10^{-3}$ in Figs. 1 and 2 respectively.

Next we show a corollary to obtain the upper bound of $\|\Delta x\|_2$ more quickly and effectively.

By the proof of Theorem 3.3 we have

$$\sigma_{\min}(\bar{H}_i) = \left\| \begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_2^{\mathbf{n}} \end{pmatrix}^{-1} \right\|_2 \quad \text{and} \quad \sigma_{\max}(\bar{K}_i) = \sigma_{\max}[(Q_n - I)(i)].$$

Hence we have the following result.

**Corollary 3.4.** *In the notation of Theorem 3.3, we have*

$$\|x - \tilde{x}\|_2 \le \|(Q_n - I)(i)\|_2 \left\| \begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_2^{\mathbf{n}} \end{pmatrix}^{-1} \right\|_2 \|x\|_2, \tag{3.11}$$

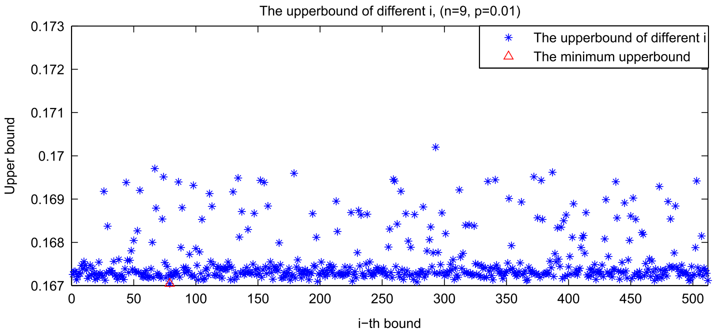

**Fig. 2.** The upper bounds of different $i$, $(n = 9, p = 0.01)$.

*and*

$$\|x-\tilde{x}\|_2/\|x\|_2 \le \|(Q_n - I)(i)\|_2 \left\| \begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_{\mathbf{2}}^{\mathbf{n}} \end{pmatrix}^{-1} \right\|_2, \tag{3.12}$$

*for every* $i = 1, 2, \ldots, 2^n$.

In the following we give a simple bound.

**Corollary 3.5.** *In the notation of Theorem 3.3, we have*

$$\|x-\tilde{x}\|_2/\|x\|_2 \le \begin{cases} [1-(1-2p)^n] \left\| \begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_{\mathbf{2}}^{\mathbf{n}} \end{pmatrix}^{-1} \right\|_2, & 0 < p \le \dfrac{1}{2} \\ 2p \left\| \begin{pmatrix} (I-\tilde{A})(i) \\ \mathbf{1}_{\mathbf{2}}^{\mathbf{n}} \end{pmatrix}^{-1} \right\|_2, & \dfrac{1}{2} < p < 1, \end{cases} \tag{3.13}$$

for $i = 1, \ldots, 2^n$.

**Proof.** Notice that $\|(Q_n - I)(i)\|_2 \le \|(Q_n - I)\|_2$. By the definition of $Q_n$ we know that its singular values are

$$1, (1-2p), \ldots, (1-2p)^n.$$

Hence we have

$$\|(Q_n - I)\|_2 = \sigma_{\max}[(Q_n - I)] = \begin{cases} 1-(1-2p)^n, & 0 < p \le \dfrac{1}{2} \\ 2p, & \dfrac{1}{2} < p < 1. \end{cases}$$

Then the result follows from Corollary 3.4. □

Now we present a bound which is independent of $i$.

**Corollary 3.6.** *In the notation of Theorem 3.3, let* $B = \begin{pmatrix} I-\tilde{A} \\ \mathbf{1}_{2^n} \end{pmatrix}$, *then*

$$\|x-\tilde{x}\|_2/\|x\|_2 \le \begin{cases} \dfrac{1-(1-2p)^n}{\sqrt{\theta}\sigma_{\min}^{+}(B)}, & 0 < p \le 1/2, \\ \dfrac{2p}{\sqrt{\theta}\sigma_{\min}^{+}(B)}, & 1/2 < p < 1, \end{cases} \tag{3.14}$$

*where* $\frac{(2^n+1)-\sqrt{(2^n+1)^2-4}}{2} \le \theta \le \frac{(2^n+1)+\sqrt{(2^n+1)^2-4}}{2}$.

**Proof.** Let $H_i$ be as in the proof of Theorem 3.3. Then by the Ostrowski theorem, we get

$$\sigma^2(H_i) = \lambda(H_i H_i^T) = \lambda(P_i B B^T P_i^T) = \theta(P_i P_i^T)\lambda(BB^T),$$

where

$$\lambda_{\min}(P_i P_i^T) \le \theta(P_i P_i^T) \le \lambda_{\max}(P_i P_i^T).$$

Since $P_iP_i^T$ is similar with $P_1P_1^T$, $P_iP_i^T$ has the same spectrum for $i = 1, 2, 3, \ldots, 2^n$. It is easy to see that the eigenvalues of $P_1P_1^T$ are 1, $\frac{(2^n+1)\pm\sqrt{(2^n+1)^2-4}}{2}$. Hence

$$\frac{(2^n+1)-\sqrt{(2^n+1)^2-4}}{2} \le \theta(P_iP_i^T) \le \frac{(2^n+1)+\sqrt{(2^n+1)^2-4}}{2}$$

for $i = 1, 2, 3, \ldots, 2^n$. Notice that $P_iBB^TP_i^T$ and $BB^T$ have the same number of positive eigenvalues. Then

$$\sigma_{\min}^{+}(H_i) = \sqrt{\theta(P_iP_i^T)}\sigma_{\min}^{+}(B).$$

Then the result follows from Corollary 3.5. □

**Remark 3.2.** (1) For an ergodic PBN model, our bound in (3.6) is given without any restriction.

(2) Let $B = \widetilde{A} - I$ and $P_{B^*}^{\perp}$ be the projection complementary to $P_{B^*}$. In [21] the authors obtained that $\tilde{x} - x = Qx + P_{B^*}^{\perp}z$, where $\mathbf{z} \in \mathbb{R}^{2^n}$ and $Q = B^{\dagger}((I-(1-p)^n)I - \widetilde{P}_n)$, $B^{\dagger}$ is the Moore–Penrose inverse of $B$. In their numerical tests, $\mathbf{z}$ is taken to be zero, which leads to $\tilde{x} - x = Qx$, but it is very difficult to judge whether $\sum_{i=1}^{2^n}(Qx)_i = 0$ or not. If $\sum_{i=1}^{2^n}(Qx)_i \neq 0$, then $\tilde{x}$ is not a steady distribution of the PBN with gene perturbation.

(3) In the literature, Wei [24] gave a relative perturbation bound of a singular linear system with index one by group inverse. Here we give a randomly chosen example to compare two perturbation bounds. We consider the following transition probability matrix of 3 genes PBN:

$$A = \begin{pmatrix}
0.2417 & 0.1117 & 0.0195 & 0.1916 & 0.1698 & 0.2687 & 0.1480 & 0.1066 \\
0.0588 & 0.1557 & 0.0439 & 0.0155 & 0.1255 & 0.0213 & 0.0358 & 0.1569 \\
0.1701 & 0.1943 & 0.1735 & 0.1361 & 0.1887 & 0.1324 & 0.0770 & 0.1541 \\
0.1820 & 0.0065 & 0.1524 & 0.1732 & 0.0948 & 0.3083 & 0.1315 & 0.0595 \\
0.1920 & 0.1745 & 0.1724 & 0.0688 & 0.0199 & 0.0025 & 0.1572 & 0.1717 \\
0.0129 & 0.1794 & 0.1162 & 0.1148 & 0.1989 & 0.0211 & 0.1035 & 0.1535 \\
0.0857 & 0.0994 & 0.2045 & 0.1855 & 0.0896 & 0.1066 & 0.1729 & 0.0627 \\
0.0568 & 0.0784 & 0.1177 & 0.1144 & 0.1127 & 0.1390 & 0.1739 & 0.1349
\end{pmatrix}$$

and set the perturbation probability is $p = 0.1$. We find that the relative bound in [24] is 1.5679, while our bound is 0.6691.

## 4. The computation of the steady-state probability distribution

In this section, we propose a fast algorithm to compute the steady-state probability distribution of a PBNs with gene perturbation. For convenience, some notations, definitions and results which will be used in sequel are given below.

**Definition 4.1** ([25]). Let $A$ be an $n \times n$ matrix. The splitting $A = M - N$ is called:

(a) weak regular if $M^{-1} \ge 0$ and $M^{-1}N \ge 0$;
(b) regular if $M^{-1} \ge 0$ and $N \ge 0$;
(c) an $M$-splitting if $M$ is a nonsingular $M$-matrix and $N \ge 0$.

**Lemma 4.1** ([25]). *Let $A$ be an irreducible singular $M$-matrix, and let $A = M - N$ be a weak regular splitting. Then $B = I - M^{-1}N$ is a singular $M$-matrix and $\text{ind}_0(B) = 1$, where $\text{ind}_\lambda(B)$ denotes the index of the eigenvalue $\lambda$ for $B$; that is, the size of the largest Jordan block of $B$ associated with $\lambda$.*

**Lemma 4.2** ([23]). *Let $A$ be singular but the linear system $Ax = b$ be consistent. The stationary iteration*

$$x_{k+1} = Tx_k + c, \quad k = 1, 2, \ldots, x_0 \in \mathbb{C}^n$$

*where $A = M - N$, $T = M^{-1}N \equiv I - M^{-1}A$, $c \equiv M^{-1}b$, semi-converges to a solution of $Ax = b$ if and only if the following conditions are satisfied:*

(a) $\rho(T) = 1$,
(b) $\vartheta(T) \equiv \max\{|\lambda \,\|\, \lambda \in \sigma(T), \lambda \neq 1\} < 1$,
(c) *all elementary divisors associated with the eigenvalue 1 of $T$ are linear, i.e., $\text{ind}_1(T) = 1$.*

### 4.1. Semi-convergent splittings

In this subsection, a semi-convergent splitting of $I-\tilde{A}$ is proposed. By Theorem 2.1 we get,

$$I-\tilde{A}=(1+(1-p)^n)I-Q_n-\hat{A}. \tag{4.1}$$

Since $Q_n$ has Kronecker product structure and $Q_n^T=Q_n$, all the eigenvalues of $Q_n$ are real. Furthermore, the summation of each column of $Q_n$ is 1 and $Q_n \geq 0$, then $\rho(Q_n)=1$. Let $M=(1+(1-p)^n)I-Q_n$, $N=\hat{A}$. Then $M$ is a Stieltjes matrix. If $0<p<1$, then $Q_n \gg 0$. Hence $M$ is a positive Stieltjes matrix. It follows from Lemma 3.1 that $M$ is a nonsingular $M$-matrix with $M^{-1} \gg 0$. Then the splitting $(I-\tilde{A})=M-N$ is an $M$-splitting, and hence we have $\rho(M^{-1}N)=1$ and $ind_1(M^{-1}N)=1$ by Lemma 4.1. On the other hand, since $M^{-1} \gg 0$, $N \geq 0$, and every column of $N$ is nonzero, we have $M^{-1}N \gg 0$, and thus, for every $\lambda \in \sigma(M^{-1}N)$ and $\lambda \neq 1$, we have $|\lambda| < \rho(M^{-1}N)$, i.e., $\vartheta(M^{-1}N) < 1$. By the argument as above we have the following theorem:

**Theorem 4.3.** *Let $M=(1+(1-p)^n)I-Q_n$, $N=\hat{A}$, then the splitting $I-\tilde{A}=M-N$ is a semi-convergent splitting.*

### 4.2. The algorithm

Let $\tilde{x}$ be the steady-state distribution of PBN with gene perturbation, i.e., $\tilde{A}\tilde{x}=\tilde{x}$. Then $(I-\tilde{A})\tilde{x}=0$, that is

$$(M-N)\tilde{x}=0. \tag{4.2}$$

By Theorem 4.3, the following iteration is semi-convergent for every initial approximation:

$$\tilde{x}^{k+1}=M^{-1}N\tilde{x}^k, \quad k=1,2,\ldots.$$

To avoid to compute the inverse of $M$, we use the following iteration

$$M\tilde{x}^{k+1}=N\tilde{x}^k, \quad k=1,2,\ldots. \tag{4.3}$$

In this iteration, we must solve the linear equation

$$M\tilde{x}^{k+1}=y, \tag{4.4}$$

in every iteration step. In order to reduce the computation cost, the special structure of perturbation matrix $Q_n$ must be considered.

Since $Q_1$ is a $2\times 2$ symmetric matrix, it can be decomposed as follows:

$$Q_1=\begin{pmatrix}1-p & p\\ p & 1-p\end{pmatrix}=P_1\Lambda_1P_1,$$

where

$$\Lambda_1=\begin{pmatrix}1 & 0\\ 0 & 1-2p\end{pmatrix}, \qquad P_1=\sqrt{2}/2\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}.$$

$P_1$ has the $LU$ decomposition:

$$P_1=L_1U_1, \quad L_1=\begin{pmatrix}1 & 0\\ 1 & 1\end{pmatrix}, U_1=\sqrt{2}/2\begin{pmatrix}1 & 1\\ 0 & -2\end{pmatrix}. \tag{4.5}$$

Hence we have

$$\begin{aligned}Q_n &= \overbrace{Q_1\otimes Q_1\otimes\cdots\otimes Q_1}^{n\ trems}\\ &= (P_1\Lambda_1P_1)\otimes\cdots\otimes(P_1\Lambda_1P_1)\\ &= (P_1\otimes\cdots\otimes P_1)(\Lambda_1\otimes\cdots\otimes\Lambda_1)(P_1\otimes\cdots\otimes P_1)\\ &= P_n\Lambda_nP_n,\end{aligned} \tag{4.6}$$

where $P_n=\overbrace{P_1\otimes\cdots\otimes P_1}^{n\ terms}$, $\Lambda_n=\overbrace{\Lambda_1\otimes\cdots\otimes\Lambda_1}^{n\ terms}$. Furthermore, $P_1$ is a orthogonal and symmetric matrix, then $P_n$ is also orthogonal and symmetric.

$$M=P_n((1+(1-p)^n)I-\Lambda_n)P_n.$$

By considering the Kronecker structure of $P_n$ and the $LU$ decomposition of $P_1$ we get

$$\begin{aligned} P_n &= (L_1U_1) \otimes \cdots \otimes (L_1U_1) \\ &= (L_1 \otimes \cdots \otimes L_1)(U_1 \otimes \cdots \otimes U_1) \\ &= L_nU_n. \end{aligned}$$

Finally, we have

$$M = L_nU_n((1 + (1 - p)^n)I - \Lambda_n)L_nU_n. \tag{4.7}$$

From (4.7), we know that the matrices $L_n$ and $U_n$ are independent of the gene perturbation probability $p$, so we need not to compute these matrices every time, just only need to store the diagonal elements of $\Lambda_n$ as a vector. So we can solve the linear system (4.4) by solving four triangular equations. Furthermore, the matrix $L_n$ is a lower triangular matrix with all lower triangular elements are 1 or 0, so, when we solve the system $L_ny = z$, we need not to compute the product between the entries of $L_n$ and $y$, just need to substitute the entries of $y$ to the system and do some additions and subtractions. So the equations $L_ny = z$ can be solved more quickly. The transition probability matrix $A$ of a PBN is sparse, so the matrix $N = (1 - p)^nA$ is. A sparse matrix multiplies by a vector, i.e., $Nx^k$, can reduce the computation cost, too. According to the discussion above, we have the algorithm as follows:

```
(1) Algorithm 1:
(2) $k = 0, x^0 = \mathbf{1}./2^n$;
(3) Do
    (a) (a) $y = Nx^k$;
    (b) (b)
    (c) solve the lower triangular equation $L_nx = y$;
    (d) solve the upper triangular equation $U_ny = x$;
    (e) $x = diag(1 + (1 - p)^nI - \Lambda_n)^{-1}. * y$;
    (f) solve the lower triangular equation $L_ny = x$;
    (g) solve the upper triangular equation $U_nx = y$;
    (h) (c)
    (i) $k = k + 1$;
    (j) $\tilde{x}^k = x$;
    (k) residual$(k) = \|\tilde{A}\tilde{x} - \tilde{x}\|$;
(4) Until residual(k) < tol.
```

We also get $\tilde{x}^{k+1}$ from $\tilde{x}^{k+1} = M^{-1}y$, where

$$M^{-1} = P_n^{-1}((1 + (1 - p)^n)I - \Lambda_n)^{-1}P_n^{-1},$$

furthermore, $P_nP_n = I$, hence we have

$$\begin{aligned} M^{-1} &= P_n((1 + (1 - p)^n)I - \Lambda_n)^{-1}P_n \\ &= L_nU_n((1 + (1 - p)^n)I - \Lambda_n)^{-1}L_nU_n, \end{aligned}$$

and

$$L_n = \begin{pmatrix} L_{n-1} & 0 \\ L_{n-1} & L_{n-1} \end{pmatrix}, \qquad U_n = \begin{pmatrix} \sqrt{2}/2U_{n-1} & \sqrt{2}/2U_{n-1} \\ 0 & -\sqrt{2}U_{n-1} \end{pmatrix}.$$

Then

$$L_n * x = \begin{pmatrix} L_{n-1} * x_{1:2^{n-1}} \\ L_{n-1} * x_{1:2^{n-1}} + L_{n-1} * x_{1+2^{n-1}:2^n} \end{pmatrix},$$

$$U_n * x = \begin{pmatrix} \sqrt{2}/2U_{n-1} * x_{1:2^{n-1}} + \sqrt{2}/2U_{n-1} * x_{1+2^{n-1}:2^n} \\ -\sqrt{2}U_{n-1} * x_{1+2^{n-1}:2^n} \end{pmatrix}.$$

In order to compute both $L_n * x$ and $U_n * x$ it need only to compute $L_{n-1} * x_{1:2^{n-1}}$, $L_{n-1} * x_{1+2^{n-1}:2^n}$ and $U_{n-1} * x_{1:2^{n-1}}$, $U_{n-1} * x_{1+2^{n-1}:2^n}$, respectively. By this way we reduce half of the computation cost. We change the step (b) to the following step $(b')$ and give **Algorithm 2**.

```
(1) $(b')$
(2) $x_1 = U_{n-1} * y_{1:2^{n-1}}$;  $x_2 = U_{n-1} * y_{1+2^{n-1}:2^n}$;
(3) $x = \begin{pmatrix} \sqrt{2}/2x_1 + \sqrt{2}/2x_2 \\ -\sqrt{2}x_2 \end{pmatrix}$;
(4) $x_1 = L_{n-1} * x_{1:2^{n-1}}$; $x_2 = L_{n-1} * x_{1+2^{n-1}:2^n}$;
(5) $y = \begin{pmatrix} x_1 \\ x_1 + x_2 \end{pmatrix}$;
```

```
 (6) $x = diag(1 + (1 - p)^n)I - \Lambda_n^{-1} .* y$;
 (7) $x_1 = U_{n-1} * x_{1:2^{n-1}}$; $x_2 = U_{n-1} * x_{1+2^{n-1}:2^n}$;
 (8) $y = \begin{pmatrix} \sqrt{2}/2x_1 + \sqrt{2}/2x_2 \\ -\sqrt{2}x_2 \end{pmatrix}$;
 (9) $x_1 = L_{n-1} * y_{1:2^{n-1}}$; $x_2 = L_{n-1} * y_{1+2^{n-1}:2^n}$;
(10) $x = \begin{pmatrix} x_1 \\ x_1 + x_2 \end{pmatrix}$.
```

We note that Algorithms 1–2 must store the matrix $P_n$ or $L_n$ and $U_n$and the method in (2.11) can reduce the storage of algorithm. So we may use the method in (2.11) to avoid to store these matrices. Let $M_1 = \Pi_{2,2^n}^T (I_{2^{n-1}} \otimes P_1)$. Then

$$P_n = P_1 \otimes \cdots \otimes P_1 = \underbrace{M_1 \cdots M_1}_{n\ terms}. \tag{4.8}$$

The vector $M^{-1} * y$ can be computed by the following step ($b''$). Changing the step (b) to ($b''$) gives Algorithm 3. This algorithm need only to store a 2 × 2 real matrix $P_1$, and hence can reduce much storage.

```
(1) ($b''$)
(2) For $i = 1 : n$
    (a) $y = (P_1 \cdot reshape(y, 2, 2^{n-1}))^T$;
(3) end
(4) $y = reshape(y, 2^n, 1)$;
(5) $x = diag(1 + (1 - p)^n)I - \Lambda_n^{-1} .* y$;
(6) For $i = 1 : n$
    (a) $x = (P_1 \cdot reshape(x, 2, 2^{n-1}))^T$;
(7) end
(8) $x = reshape(x, 2^n, 1)$.
```

As we know, the generalized minimal residual (GMRES) method is widely used for solving a general linear system

$$Ax = b, \quad A \in \mathbb{R}^{n \times n}, \ b \neq \mathbf{0}.$$

From the proof of Theorem 3.3, the following system of linear Eq. (4.9) have a unique solution $\tilde{x}$:

$$\bar{A}\tilde{x} = \bar{b}, \tag{4.9}$$

where

$$\bar{A} = \begin{pmatrix} (\tilde{A} - I)_{2:2^n} \\ \mathbf{1}_{2^n} \end{pmatrix}, \qquad \bar{b} = \begin{pmatrix} \mathbf{0}'_{2^n-1} \\ 1 \end{pmatrix}.$$

Hence $\bar{b} \neq \mathbf{0}$, alternatively, we may use GMRES method to solve the linear system (4.9). We denote the method by ‘M-GMRES’. Notice that this method destroys the special structure of the perturbation matrix $\tilde{P}_n$, actually.

The transition probability matrix $A$ of PBN is a sparse matrix, so is $N = \hat{A} = (1 - p)^n A$. The complexity to compute $N\tilde{x}^k$ in (4.3) is $O(m2^n)$, where $m$ is the maximum number of nonzero elements of every row of $A$. The matrices $P_n$, $L_n$ and $U_n$ are fixed for the given $n$, so it is not necessary to compute these matrices every time. In Algorithm 1, we need only to compute the product of a triangular matrix and a vector, and thus the complexity is $O(2^{2n-1})$. By using the above argument, we know that if $m \ll 2^n$, the complexity of each step for Algorithms 1 and 2 is $O(2^{2n-1})$ and $O(2^{2n-3})$, respectively, and the complexity of every step of Algorithm 3 is $O(2^n)$.

If we use the same technique in the power method,

$$\tilde{A}x = \hat{A}x + P_n(\Lambda_n - (1 - p)^n)IP_n x,$$

we could also reduce the complexity of the power method to $O(2^n)$. But the numerical examples show that number of iterations of the power method is more than number of iterations of Algorithm 3.

## 5. Numerical examples

In this section we demonstrate the efficiency of the proposed algorithms by some numerical examples. Firstly, we test our algorithms by an example with 3-genes network proposed in [6]. Secondly, we generate some random $2^n \times 2^n$ sparse nonnegative matrix as the transition probability matrix of a PBN to show the effectiveness of the proposed algorithms. Finally, we compare the relative error $d$ computed by the proposed methods and the Xu et al.’s method [21]. All the runs were done in Matlab 7.9.0 on a CPU 2.66 GHz and 3.48 GB memory computer, and the termination tolerance $\varepsilon = 1.0e - 9$. When we use the power method, we will stop calculating as it iterate 5000 times.

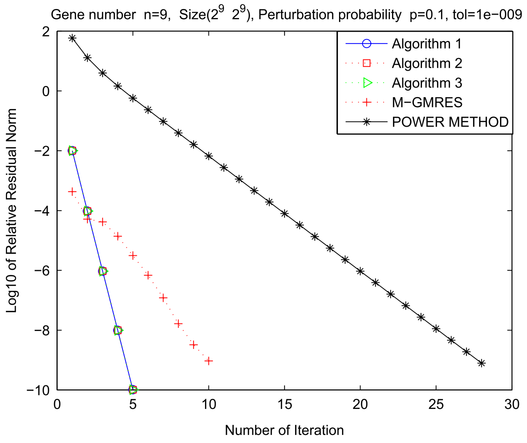

**Fig. 3.** The relative residual norm for $n = 9$ and $p = 0.1$ of different methods.

**Example 5.1** ([6]). The truth table and the state transition probability matrix of the PBN are also given in the Example 2.2. Suppose the perturbation probability $p = 0.2$. Then the steady state distributions $\tilde{x}_p$ and $\tilde{x}$ given by the power method and Algorithm 3 are

$$\tilde{x}_p = \begin{pmatrix} 0.12364045178 \\ 0.06123450558 \\ 0.12683169459 \\ 0.06269248594 \\ 0.12755601798 \\ 0.12010837289 \\ 0.19350277613 \\ 0.18443369507 \end{pmatrix} \quad \text{and} \quad \tilde{x} = \begin{pmatrix} 0.12364045170 \\ 0.06123450557 \\ 0.12683169455 \\ 0.06269248596 \\ 0.12755601796 \\ 0.12010837292 \\ 0.19350277612 \\ 0.18443369517 \end{pmatrix},$$

respectively. Hence the residual errors are

$$\|\tilde{A}\tilde{x}_p - \tilde{x}_p\|_2 = 0.1808 \times 10^{-9} \quad \text{and} \quad \|\tilde{A}\tilde{x} - \tilde{x}\|_2 = 0.1245 \times 10^{-9}.$$

The iteration number of the power method and Algorithm 3 are 31 and 29, respectively, and

$$\|\tilde{x} - \tilde{x}_p\|_2 = 1.3950 \times 10^{-10}.$$

**Example 5.2.** In this example, we show numerically the effectiveness of the proposed algorithm by counting iteration numbers (denoted by 'IT'), elapsed CPU time in seconds (denoted by 'CPU'), and relative residual error (denoted by 'RES') defined by

$$RES = \frac{\|\tilde{A}\tilde{x}^k - \tilde{x}^k\|_2}{\|\tilde{A}\tilde{x}^0 - \tilde{x}^0\|_2}.$$

The transition probability matrix without gene perturbations $A$ is produced by using the Matlab function $A = sprand(2^n, 2^n, density)$ that give a random $2^n \times 2^n$ sparse matrix with approximately $density * 2^n \times 2^n$ uniformly distributed nonzero entries then following by a column normalization. The number of nonzero entries of $A$ around $5.0 \times 10^4$.

Numerical results for $p = 0.10$ and different values of $n$ are given in Table 3. Table 4 gives the numerical results for $n = 10$ and different values of $p$. By Tables 3 and 4, it can be seen that for each gene perturbation probability $p$ or for each gene number $n$, the iteration numbers and most of relative residual errors of the proposed method are smaller than those by the power method. On the other hand, the iteration number of the proposed method is decreasing as the gene perturbation probability $p$ is increased. The relative residual errors of every step of different methods are shown in Fig. 3. Table 5 gives the numerical results for $p = 0.01$ and a large value of $n$. The example shows that the proposed method is numerically effective.

**Example 5.3.** In this example, we compare the relative error computed by the proposed method, by the power method, by M-GMRES and by the approximation method given in [21]. Let $\tilde{x}$ and $\tilde{x}_p$ denote the steady-state probability distributions computed by one of the proposed methods and the power method, respectively, and denote by $d$ as

$$d \doteq \frac{\|\tilde{x} - \tilde{x}_p\|_2}{\|\tilde{x}_p\|_2}.$$

**Table 3**
Numerical results for $p = 0.10$ and different $n$.

| $n$ | $p$ | Algorithm 1 | | | Algorithm 2 | | | Algorithm 3 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | IT | CPU | RES | IT | CPU | RES | IT | CPU | RES |
| 7 | 0.10 | 9 | 0.0174 | 4.43e−010 | 9 | 0.0000 | 4.43e−010 | 9 | 0.0000 | 4.43e−010 |
| 8 | 0.10 | 8 | 0.0573 | 9.13e−011 | 8 | 0.0000 | 9.12e−011 | 8 | 0.0000 | 9.12e−011 |
| 9 | 0.10 | 7 | 0.1391 | 7.21e−011 | 7 | 0.0000 | 7.20e−011 | 7 | 0.0000 | 7.20e−011 |
| 10 | 0.10 | 6 | 0.5328 | 1.47e−010 | 6 | 0.0625 | 1.46e−010 | 6 | 0.0313 | 1.47e−010 |
| 11 | 0.10 | 5 | 2.7607 | 6.80e−010 | 5 | 0.1719 | 6.78e−010 | 5 | 0.1250 | 6.78e−010 |
| 12 | 0.10 | 5 | 18.2859 | 1.03e−010 | 5 | 0.5000 | 7.02e−011 | 5 | 0.3906 | 6.88e−011 |

| $n$ | $p$ | M-GMRES | | | Power method | | |
|---|---|---|---|---|---|---|---|
| | | IT | CPU | RES | IT | CPU | RES |
| 7 | 0.10 | 10 | 0.0000 | 3.84e−009 | 22 | 0.0000 | 5.04e−010 |
| 8 | 0.10 | 10 | 0.0000 | 9.02e−010 | 25 | 0.0000 | 9.47e−010 |
| 9 | 0.10 | 10 | 0.0000 | 3.89e−010 | 29 | 0.0156 | 4.26e−010 |
| 10 | 0.10 | 9 | 0.0625 | 6.94e−010 | 32 | 0.1250 | 8.89e−010 |
| 11 | 0.10 | 9 | 0.1250 | 3.06e−010 | 36 | 0.5938 | 8.26e−010 |
| 12 | 0.10 | 9 | 0.3281 | 1.74e−010 | 40 | 2.5000 | 7.15e−010 |

**Table 4**
Numerical results for $n = 10$ and different $p$.

| $n$ | $p$ | Algorithm 1 | | | Algorithm 2 | | | Algorithm 3 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | IT | CPU | RES | IT | CPU | RES | IT | CPU | RES |
| 10 | 0.08 | 6 | 0.4993 | 5.53e−010 | 6 | 0.0625 | 5.52e−010 | 6 | 0.0313 | 5.52e−010 |
| 10 | 0.18 | 5 | 0.4200 | 6.73e−011 | 5 | 0.0625 | 5.92e−011 | 5 | 0.0313 | 5.91e−011 |
| 10 | 0.28 | 4 | 0.3303 | 6.44e−011 | 4 | 0.0625 | 3.29e−011 | 4 | 0.0313 | 3.29e−011 |
| 10 | 0.38 | 3 | 0.2555 | 2.20e−010 | 3 | 0.0313 | 1.46e−010 | 3 | 0.0313 | 1.46e−010 |
| 10 | 0.48 | 4 | 0.3484 | 9.32e−010 | 3 | 0.0156 | 6.71e−012 | 3 | 0.0156 | 6.49e−012 |

| $n$ | $p$ | M-GMRES | | | Power method | | |
|---|---|---|---|---|---|---|---|
| | | IT | CPU | RES | IT | CPU | RES |
| 10 | 0.08 | 10 | 0.0469 | 2.54e−010 | 28 | 0.1094 | 9.68e−010 |
| 10 | 0.18 | 8 | 0.0625 | 8.55e−010 | 37 | 0.1406 | 7.11e−010 |
| 10 | 0.28 | 7 | 0.0156 | 8.32e−011 | 28 | 0.1094 | 9.57e−010 |
| 10 | 0.38 | 5 | 0.0313 | 4.12e−010 | 18 | 0.0625 | 7.52e−010 |
| 10 | 0.48 | 3 | 0.0156 | 4.26e−010 | 5000 | 17.5156 | 2.42e−009 |

**Table 5**
Numerical results for $p = 0.01$ and large $n$.

| $n$ | $p$ | Algorithm 3 | | | Power method | | |
|---|---|---|---|---|---|---|---|
| | | IT | CPU | RES | IT (h) | CPU (h) | RES (h) |
| 24 | 0.01 | 5 | 83.4920 | 5.84e−13 | >5 | >5 | >5 |
| 25 | 0.01 | 6 | 210.0700 | 6.47e−13 | >5 | >5 | >5 |
| 26 | 0.01 | 6 | 436.3000 | 5.89e−11 | >5 | >5 | >5 |
| 27 | 0.01 | 4 | 603.1700 | 1.25e−14 | >5 | >5 | >5 |
| 28 | 0.01 | 4 | 1249.8500 | 4.14e−12 | >5 | >5 | >5 |
| 29 | 0.01 | 4 | 2700.2700 | 4.01e−16 | >5 | >5 | >5 |
| 30 | 0.01 | 3 | 6946.6425 | 8.69e−13 | >5 | >5 | >5 |

In [21] the authors set the number of substitutions of $Q_1$ with $I_2$ to be 2 and set $z = \mathbf{0}$ for testing their approximation method. Here we do the same setting with [21] for checking the relative error about the power method with the approximation method. However, it is not necessary to set it in the proposed methods. It is also noted that the approximation method in [21] needs to compute the Moore–Penrose generalized inverse of a matrix, which is very expensive. Table 6 gives the relative error about the power method with different methods. From the table, it is shown that the proposed method and M-GMRES are better than the approximation method in [21], Algorithm 3 also gives the best results among all the testing methods.

## 6. Discussions

In this paper, we gave a new perturbation bound without any restriction for the steady-state distribution of PBN with gene perturbation, firstly. And then, we have proposed fast algorithms for computing the steady-state probability distribution of PBNS with gene perturbation by considering the special structure of a transition probability matrix of PBNs with gene perturbation. Numerical experiments are given to demonstrate the efficiency of the proposed methods.

**Table 6**
Relative error $d$ about the power method for different methods.

| $n$ | $p$ | Algorithm 3 | M-GMRES | Approximation method [21] |
|---|---|---|---|---|
| 5 | 0.01 | 2.44e−010 | 6.36e−005 | 9.60e−003 |
| 6 | 0.01 | 1.72e−010 | 2.33e−006 | 6.34e−003 |
| 7 | 0.01 | 1.01e−010 | 9.49e−008 | 3.40e−003 |
| 8 | 0.01 | 7.63e−011 | 5.76e−009 | 2.78e−003 |
| 9 | 0.01 | 1.40e−011 | 3.59e−009 | 1.93e−003 |
| 10 | 0.01 | 7.54e−012 | 6.96e−009 | 1.31e−003 |

## Acknowledgments

The work was supported by National Natural Science Foundation (Grant No. 10971075), Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20104407110001) and Guangdong Provincial Natural Science Foundation (Grant No. 9151063101000021).

## References

[1] J.E. Celis, M. KruhØfer, I. Gromova, C. Frederiksen, M. Ostergaard, T. Thykjaer, P. Gromov, J. Yu, H. Plsdttir, N. Magnusson, T.F. Orntoft, Gene expression profiling: monitoring transcription and translation products using DNA microarrays and proteomics, FEBS Lett. 480 (2000) 2–16.
[2] T.R. Hughes, M. Mao, A.R. Jones, J. Burchard, M.J. Marton, K.W. Shannon, S.M. Lefkowitz, M. Ziman, J.M. Schelter, M.R. Meyer, S. Kobayashi, C. Davis, H. Dai, Y.D. He, S.B. Stephaniants, G. Cavet, W.L. Walker, A. West, E. Coffey, D.D. Shoemaker, R. Stoughton, A.P. Blanchard, S.H. Friend, P.S. Linsley, Expression profiling using microarrays fabricated by an ink-jet oligonucleotide synthesizer, Nat. Biotechnol. 19 (2001) 342–347.
[3] K. Murphy, S. Mian, Modelling gene expression data using dynamic Bayesian networks, Technical Report, Berkeley, 1999.
[4] T. Akutsu, S. Miyano, S. Kuhara, Identification of genetic networks from a small number of gene expression patterns under the Boolean network model, Pac. Symp. Biocomput. 4 (1999) 17–28.
[5] T. Akutsu, M. Hayasida, W. Ching, M. Ng, Control of Boolean networks: hardness results and algorithms for tree structured networks, J. Theor. Biol. 244 (2007) 670–679.
[6] I. Shmulevich, E.R. Dougherty, S. Kim, W. Zhang, Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks, Bioinformatics 18 (2002) 261–274.
[7] I. Shmulevich, E.R. Dougherty, W. Zhang, Gene perturbation and intervention in probabilistic Boolean networks, Bioinformatics 18 (2002) 1319–1331.
[8] W. Ching, E. Fung, M. Ng, T. Akutsu, On construction of stochastic genetic networks based on gene expression sequences, Int. J. Neural Syst. 15 (2005) 297–310.
[9] W. Ching, M. Ng, Markov chains: models, algorithms and applications, in: International Series on Operations Research and Management Science, Springer, New York, 2006.
[10] W. Ching, S. Zhang, M. Ng, T. Akutsu, An approximation method for solving the steady-state probability distribution of probabilistic Boolean networks, Bioinformatics 12 (2007) 1511–1518.
[11] P. Smolen, D. Baxter, J. Byrne, Mathematical modeling of gene network, Neuron. 26 (2000) 567–580.
[12] S.A. Kauffman, Metabolic stability and epigenesis in randomly constructed genetic nets, J. Theor. Biol. 22 (1969) 437–467.
[13] S.A. Kauffman, The Origins of Order: Self-organization and Selection in Evolution, Oxford University Press, Oxford, 1993.
[14] T. Akutsu, W. Ching, Analysis and control of deterministic and probabilistic Boolean networks, in: Huma M. Lodhi, Muggleton Stephen (Eds.), Elements of Computational Systems Biology, in: Wiley Book Series on Bioinformatics, John Wiley & Sons, Inc., 2010, pp. 235–256.
[15] I. Shmulevich, E.R. Dougherty, Probabilistic Boolean Networks: The Modeling and Control of Gene Regulatory Networks, SIAM Press, 2009.
[16] M. Brun, E.R. Dougherty, I. Shmulevich, Steady-state probabilities for attractors in probabilistic Boolean networks, Signal Processing 85 (2005) 1993–2013.
[17] I. Shmulevich, I. Gluhovsky, R. Hashimoto, E.R. Dougherty, W. Zhang, Steady-state analysis of genetic regulatory networks modelled by probabilistic Boolean networks, Comp. Funct. Genomics 4 (2003) 601–608.
[18] I. Shmulevich, E.R. Dougherty, W. Zhang, Control of stationary behavior in probabilistic Boolean networks by means of structural intervention, J. Biol. Syst. 10 (2002) 431–445.
[19] L. Cui, W. Li, W. Ching, On construction of sparse probabilistic Boolean networks from a prescribed transition probability matrix, in: Proceedings of The Fourth International Conference on Computational Systems Biology, in: Lecture Notes in Operations Research, vol. 13, 2010, pp. 227–234.
[20] S. Zhang, W. Ching, M. Ng, T. Akutsu, Simulation study in probabilistic Boolean network models for genetic regulatory networks, Int. J. Data Min. Bioinform. 1 (2007) 217–240.
[21] W. Xu, W. Ching, S. Zhang, W. Li, X. Chen, A matrix perturbation method for computing the steady-state probability distributions of probabilistic Boolean networks with gene perturbations, J. Comput. Appl. Math. 235 (2011) 2242–2251.
[22] R. Varga, Martix Iterative Analysis, second ed., Science press, Beijing, 2005.
[23] A. Berman, R.J. Plemmons, Non-Negative Matrices in the Mathematical Sciences, SIAM, Philadelphia, 1994.
[24] Y. Wei, Perturbation analysis of singular linear systems with index one, Int. J. Comput. Math. 74 (2000) 483–491.
[25] H. Schneider, Theorems of M-splittings of a singular $M$-matrix which depend on graph structure, Linear Algebra Appl. 58 (1984) 407–429.