OPEN ACCESS Freely available online

PLOS ONE

Inference of Vohradský's Models of Genetic Networks by Solving Two-Dimensional Function Optimization Problems

Shuhei Kimura[1]*, **Masanao Sato**[2], **Mariko Okada-Hatakeyama**[3]

1 Graduate School of Engineering, Tottori University, Tottori, Japan, **2** National Institute for Basic Biology, Okazaki Institute for Integrative Bioscience, National Institute for Natural Sciences, Okazaki, Japan, **3** RIKEN Center for Integrative Medical Sciences, Yokohama, Japan

Abstract

The inference of a genetic network is a problem in which mutual interactions among genes are inferred from time-series of gene expression levels. While a number of models have been proposed to describe genetic networks, this study focuses on a mathematical model proposed by Vohradský. Because of its advantageous features, several researchers have proposed the inference methods based on Vohradský's model. When trying to analyze large-scale networks consisting of dozens of genes, however, these methods must solve high-dimensional non-linear function optimization problems. In order to resolve the difficulty of estimating the parameters of the Vohradský's model, this study proposes a new method that defines the problem as several two-dimensional function optimization problems. Through numerical experiments on artificial genetic network inference problems, we showed that, although the computation time of the proposed method is not the shortest, the method has the ability to estimate parameters of Vohradský's models more effectively with sufficiently short computation times. This study then applied the proposed method to an actual inference problem of the bacterial SOS DNA repair system, and succeeded in finding several reasonable regulations.

Citation: Kimura S, Sato M, Okada-Hatakeyama M (2013) Inference of Vohradský's Models of Genetic Networks by Solving Two-Dimensional Function Optimization Problems. PLoS ONE 8(12): e83308. doi:10.1371/journal.pone.0083308

Editor: Alberto de la Fuente, Leibniz-Institute for Farm Animal Biology (FBN), Germany

Received July 8, 2013; **Accepted** November 1, 2013; **Published** December 30, 2013

Funding: This work is supported by the Ministry of Education, Culture, Sports, Science and Technology of Japan under Grant-in-Aid for Young Scientists (B) No. 23700266. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: kimura@ike.tottori-u.ac.jp

Introduction

With the rapid advancement of technologies such as RNA-seq using next generation sequencers, it has become possible to measure the expression levels of thousands of genes. These data implicitly contain enormous amounts of information on biological systems. In order to exploit these high-throughput technologies, we must develop a way of extracting hidden information from the observed data. The inference of genetic networks is considered a promising approach for extracting useful information from these data. In the genetic network inference, the information is extracted by inferring mutual interactions among genes from the time-series of the gene expression levels. The inferred model of the genetic network is conceived of as an ideal tool to help biologists generate hypotheses and facilitate the design of their experiments. Many researchers have thus taken an interest in the inference of genetic networks, and the development of this methodology has become a major topic in the field of bioinformatics and systems biology.

Numerous models for describing genetic networks have been proposed, and numerous algorithms based on individual models have been developed for the inference of genetic networks [1–14]. Among these models, this study focuses especially on sets of differential equations, as they can capture the dynamic behavior of gene expression. When we use the set of differential equations to describe a genetic network, its inference is generally defined as a problem of estimating the model parameters that produce time-series data consistent with the observed gene expression levels.

A linear model is one of the best-studied models based on a set of differential equations. Several inference methods based on the linear model have therefore been proposed [13,15]. The computation times of these methods are reportedly very short. As the linear model requires that the system is operating near a steady state, however, it is unsuitable for analyzing the time-series of gene expression levels [13]. An S-system model is another well-studied model based on a set of differential equations [16,17]. As several fundamental properties of biochemical systems are inherent in this model, a number of inference methods based on it have been proposed [18–28]. However, the number of parameters in the S-system model is larger. The number of the parameters in the linear model is $N(N+1)$, where N is the number of genes contained in the target network. On the other hand, the number of the parameters in the S-system model is $2N(N+1)$. To obtain reasonable results, therefore, we should give the inference methods based on the S-system model a larger amount of the gene expression data.

When trying to infer genetic networks, we should use a mathematical model that has the ability to approximate actual biochemical reactions. As it is generally difficult to measure a sufficient amount of gene expression data, moreover, the model should contain a fewer number of the parameters. Vohradský proposed a model that is capable of representing the process of the gene expression [29]. The number of the parameters of the Vohradský's model, i.e., $N(N+3)$, is comparable to that of the

linear model. Because of its advantageous features, several researchers have proposed the inference methods based on this model [30–32]. However, these methods try to estimate all of the model parameters simultaneously. When inferring genetic networks consisting of many genes, therefore, they must solve high-dimensional non-linear function optimization problems. In order to overcome this high-dimensionality in the canonical methods, this study proposes a new approach that defines the estimation of the model parameters as two-dimensional function optimization problems. Although the defined problems are still non-linear, their low-dimensionality enhances the probability of obtaining reasonable results. Finally, we confirm the effectiveness of the proposed method by applying it to artificial and actual genetic network inference problems.

Methods

Vohradský's model

This study uses a mathematical model proposed by Vohradský [29] to describe genetic networks. The Vohradský's model is a set of differential equations of the form

$$\frac{dX_n}{dt}=\alpha_n f\left(\sum_{m=1}^{N} w_{n,m}X_m+b_n\right)-\beta_n X_n, \quad (n=1,2,\cdots,N), \qquad (1)$$

where

$$f(x)=\frac{1}{1+e^{-x}},$$

and α_n (>0), β_n (>0), b_n and $\mathbf{w}_n=(w_{n,1},w_{n,2},\cdots,w_{n,N})$ $(n=1,2,\cdots,N)$ are model parameters. In the genetic network inference, X_n is the expression level of the n-th gene and N is the number of genes contained in the target network. When we use the Vohradský's model to describe genetic networks, our purpose is to estimate all of the model parameters that produce time-series data consistent with the observed gene expression levels.

The discrete form of the model (1) is equivalent to a recurrent neural network. We can thus use learning algorithms for recurrent neural networks, such as a back-propagation through time [33], in order to estimate the parameters of this model [29]. The canonical inference methods based on the Vohradský's model [30–32] have been designed on the basis of the back-propagation through time. In contrast to these methods, on the other hand, the proposed method estimates the parameters by solving simultaneous equations, as described below.

Parameter estimation

The proposed method divides the inference problem of the Vohradský's model of a genetic network consisting of N genes into N subproblems, each of which corresponds to each gene. By solving the n-th subproblem, our method estimates the parameters corresponding to the n-th gene, i.e., α_n, β_n, b_n and $\mathbf{w}_n=(w_{n,1},w_{n,2},\cdots,w_{n,N})$. This section will describe the method to solve the n-th subproblem.

Concept

In the n-th subproblem corresponding to the n-th gene, the proposed method estimates the model parameters, α_n, β_n, b_n and $\mathbf{w}_n=(w_{n,1},w_{n,2},\cdots,w_{n,N})$, by solving the following simultaneous equations.

$$\begin{aligned}
\left.\frac{dX_n}{dt}\right|_{t_1} &= \alpha_n f\left(\textstyle\sum_{m=1}^{N} w_{n,m}\, X_m|_{t_1}+b_n\right)-\beta_n X_n|_{t_1},\\
\left.\frac{dX_n}{dt}\right|_{t_2} &= \alpha_n f\left(\textstyle\sum_{m=1}^{N} w_{n,m}\, X_m|_{t_2}+b_n\right)-\beta_n X_n|_{t_2},\\
&\vdots\\
\left.\frac{dX_n}{dt}\right|_{t_K} &= \alpha_n f\left(\textstyle\sum_{m=1}^{N} w_{n,m}\, X_m|_{t_K}+b_n\right)-\beta_n X_n|_{t_K},
\end{aligned} \qquad (2)$$

where $X_m|_{t_k}$ is the expression level of the m-th gene at time t_k, $\left.\frac{dX_n}{dt}\right|_{t_k}$ is the time derivative of the expression level of the n-th gene at time t_k, and K is the number of measurements. In the proposed approach, $X_m|_{t_k}$'s are measured using technologies such as RNA-seq, and $\left.\frac{dX_n}{dt}\right|_{t_k}$'s are estimated directly from the observed time-series of the gene expression levels using a smoothing technique such as spline interpolation [34], local linear regression [35], neural networks [28], or a modified Whittaker's smoother [36]. Based on an idea similar to the method proposed here, several genetic network inference methods have already been proposed [8,13,23,28,37].

It is not always easy to solve the simultaneous equations (2), since they are non-linear. In order to resolve the difficulty in solving these equations, this study uses a feature arisen from the transformation of them. By rearranging the k-th member of the equations (2), we have

$$Y_k=f\left(\sum_{m=1}^{N} w_{n,m}X_m|_{t_k}+b_n\right), \qquad (3)$$

where

$$Y_k=\frac{\left.\frac{dX_n}{dt}\right|_{t_k}+\beta_n X_n|_{t_k}}{\alpha_n}.$$

By applying $f(x)=\frac{1}{1+e^{-x}}$ to the equation (3), then, we obtain

$$Y_k=\frac{1}{1+\exp\left[-\left(\sum_{m=1}^{N} w_{n,m}X_m|_{t_k}+b_n\right)\right]},$$

$$\frac{1-Y_k}{Y_k}=\exp\left[-\left(\sum_{m=1}^{N} w_{n,m}X_m|_{t_k}+b_n\right)\right]. \qquad (4)$$

By taking the logarithms of both sides of the equation above, we finally have

$$\log\left(\frac{Y_k}{1-Y_k}\right)=\sum_{m=1}^{N} w_{n,m}X_m|_{t_k}+b_n. \qquad (5)$$

Note that, although the transformed equation (5) is non-linear with respect to the parameters α_n and β_n, it is linear with respect to the parameters b_n and $\mathbf{w}_n=(w_{n,1},w_{n,2},\cdots,w_{n,N})$. This fact suggests that, when the parameters α_n and β_n are given, the other parameters b_n and $\mathbf{w}_n$ are easily estimated. The proposed method uses this feature for solving the simultaneous equations (2), as described below.

Solving the simultaneous equations

As mentioned just before, we can easily estimate the parameters b_n and $\mathbf{w}_n$, when the parameters α_n and β_n are given. In this study, the set of algebraic equations (2) is thus solved by estimating the parameters α_n and β_n. As this study uses a least-squares method to solve the simultaneous equations, the estimation of the parameters α_n and β_n is defined as a problem of minimizing the following two-dimensional function.

$$S_n(\alpha_n,\beta_n)=\sum_{k=1}^{K}\left[\left.\frac{dX_n}{dt}\right|_{t_k}-R_k(\alpha_n,\beta_n,b_n^*,\mathbf{w}_n^*)\right]^2, \qquad (6)$$

where

$$R_k(\alpha_n,\beta_n,b_n,\mathbf{w}_n)=\alpha_n f\left(\sum_{m=1}^{N} w_{n,m}\,X_m|_{t_k}+b_n\right)-\beta_n X_n|_{t_k},$$

and b_n^* and $\mathbf{w}_n^*$ are the optimal values for b_n and $\mathbf{w}_n$, respectively, under given α_n and β_n. In the next section, we will describe a method for obtaining b_n^* and $\mathbf{w}_n^*$.

Any function optimization algorithm can be used to minimize the objective function (6). When we used the local search for optimizing this function, however, several local optima were found. As this optimization problem seemed to be multimodal, this study uses an evolutionary algorithm, REX^{star}/JGG (see Supporting Information) [38], to solve it. Because the parameters α_n and β_n are positive, this study searches for them in a logarithmic space.

Estimation of b_n^* and $\mathbf{w}_n^*$

In order to compute a value for the objective function (6), we must provide values for b_n^* and $\mathbf{w}_n^*$. In the proposed method, they serve as the solution of a set of the transformed equations (5) under given α_n and β_n. Note that, when the parameters α_n and β_n are given, these equations are linear with respect to the unknown parameters, i.e., b_n and $\mathbf{w}_n^*$. We can thus easily obtain b_n^* and $\mathbf{w}_n^*$. The proposed method estimates these parameters by optimizing the following constrained function minimization problem.

$$\underset{b_n,\,\mathbf{w}_n,\xi_k^+,\xi_k^-}{\text{minimize}} \sum_{m=1}^{N}|w_{n,m}|+\frac{C}{\Gamma}\sum_{k=1}^{K}\left(\gamma_k^+\,\xi_k^+ +\gamma_k^-\,\xi_k^-\right), \qquad (7)$$

subject to

$$\begin{cases} \sum_{m=1}^{N} w_{n,m}\,X_m|_{t_k}+b_n-L_k\le\xi_k^+, & (k=1,2,\cdots,K),\\ \xi_k^+\ge 0, & (k=1,2,\cdots,K),\\ \sum_{m=1}^{N} w_{n,m}\,X_m|_{t_k}+b_n-L_k\ge-\xi_k^-, & (k=1,2,\cdots,K),\\ \xi_k^-\ge 0, & (k=1,2,\cdots,K), \end{cases}$$

where

$$L_k=\log\left(\frac{Z_k}{1-Z_k}\right),$$

$$Z_k=\begin{cases} Y_k, & (if\ \delta\le Y_k\le 1-\delta),\\ \delta, & (if\ Y_k<\delta),\\ 1-\delta, & (\text{otherwise}), \end{cases}$$

$$\Gamma=\frac{1}{2}\sum_{k=1}^{K}(\gamma_k^+ +\gamma_k^-),$$

ξ_k^+ and ξ_k^- are slack variables, and γ_k^+, γ_k^-, δ and C are constant parameters. In this problem, we treat the parameters α_n and β_n as constants. Note that, whenever trying to compute a value for the objective function (6), we must always solve the problem (7) (see Figure 1).

When the n-th gene is not regulated by the m-th gene, the parameter corresponding to this regulation, i.e., $w_{n,m}$, is zero in the Vohradský's model. Because genetic networks are known to be sparsely connected [39], most of $w_{n,m}$'s should be zero. The first term of the objective function of the problem (7), i.e., $\sum_{m=1}^{N}|w_{n,m}|$, introduces this a priori knowledge into our parameter estimation. The second term of the objective function is, on the other hand, a sum of the differences between the left and right hand sides of the equations (5).

Note that, only when the condition $0<Y_k<1$ is satisfied, we can transform the equation (2) into the equation (5). However, the observed gene expression data are generally polluted by noise. Even when the optimum values are set for α_n and β_n, some Y_k's might not satisfy the condition above. This study thus introduces a threshold parameter δ, and sets its value to 10^{-10}. On the other hand, we should note that, when Y_k approaches 0 or 1, the term $\log\left(\frac{Y_k}{1-Y_k}\right)$ contained in the equation (5) approaches $-\infty$ or $+\infty$, respectively. When Y_k is approximately equal to 0 or 1, therefore, the transformation of the equation (2) into the equation (5) would amplify the noise contained in the measurement data. It is thus inadvisable to rely too much on the equations transformed under this condition. In order to introduce this notion into our estimation, this study sets the constant parameters γ_k^+ and γ_k^- to

$$\gamma_k^+=\begin{cases} \left[1-4\left(Z_k-\frac{1}{2}\right)^2\right]^{\frac{1}{2}}, & (if\ 0\le Z_k\le\frac{1}{2}),\\ 1-4\left(Z_k-\frac{1}{2}\right)^2, & (\text{otherwise}), \end{cases}$$

$$\gamma_k^-=\begin{cases} 1-4\left(Z_k-\frac{1}{2}\right)^2, & (if\ 0\le Z_k\le\frac{1}{2}),\\ \left[1-4\left(Z_k-\frac{1}{2}\right)^2\right]^{\frac{1}{2}}, & (\text{otherwise}). \end{cases}$$

We can transform the optimization problem (7) to a linear programming problem. Thus, the proposed method easily solves this problem by using an interior point method [40].

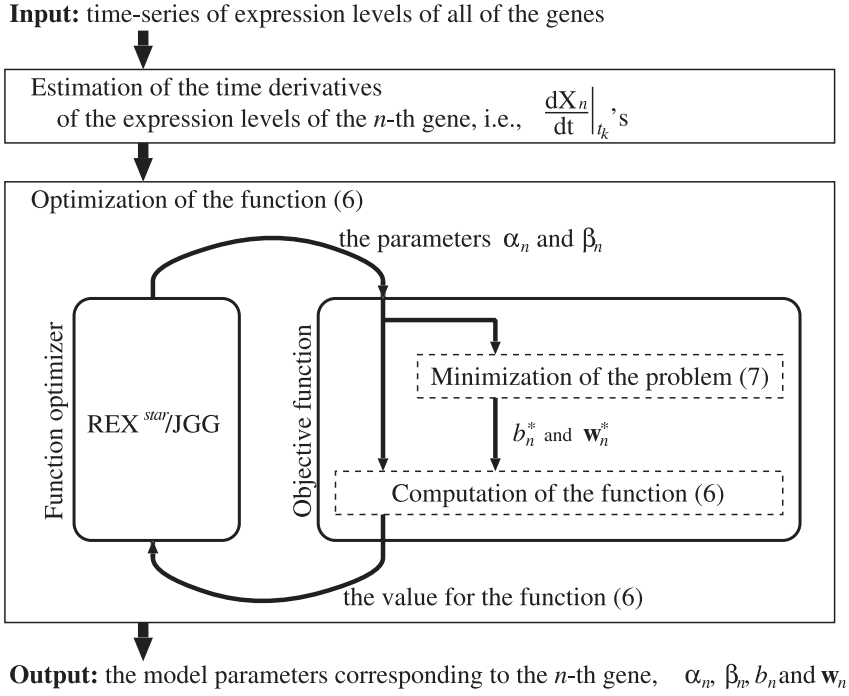

Figure 1. A framework of the algorithm for solving the *n*-th subproblem. Note that the proposed method divides the inference of the Vohradský's model of a genetic network consisting of N genes into N subproblems. By solving the n-th subproblem, we have the model parameters corresponding to the n-th gene, i.e., α_n, β_n, b_n and $\mathbf{w}_n=(w_{n,1},w_{n,2},\cdots,w_{n,N})$.
doi:10.1371/journal.pone.0083308.g001

Remarks

As mentioned before, this study proposes to define the estimation of the model parameters corresponding to the n-th gene as a problem of solving the simultaneous equations (2). The proposed method effectively solves them by minimizing the two-dimensional function optimization problem (6). We can however solve the simultaneous equations simply by using a least-squares method. In this case, for example, the parameters corresponding to the n-th gene are estimated by minimizing

$$T_n(\alpha_n,\beta_n,b_n,\ \mathbf{w}_n)= \sum_{k=1}^{K}\left[\left.\frac{dX_n}{dt}\right|_{t_k}-R_k(\alpha_n,\beta_n,b_n,\ \mathbf{w}_n)\right]^2+C_{lsq}\sum_{m=1}^{N-I}|W_{n,m}|, \quad (8)$$

where $W_{n,m}$'s are given by rearranging $w_{n,m}$'s in descending order of their absolute values, i.e., $|W_{n,1}|\leq|W_{n,2}|\leq\cdots\leq|W_{n,N}|$. C_{lsq} is a constant parameter, and I is a maximum indegree. The maximum indegree determines the maximum number of genes that affect the n-th gene directly. The second term of the objective function (8) is a penalty term that introduces the sparseness into the inferred network. Similar terms have been used in several genetic network inference methods [21,22,24].

As mentioned before, the existing inference methods based on the Vohradský's model try to estimate all of the model parameters simultaneously [30–32]. It is therefore difficult for them to analyze genetic networks with many genes because of the high-dimensionality in the parameter estimation. For the computational simplicity, moreover, they limit the model to being $\alpha_n=\beta_n$. This study thus compared the proposed method chiefly with a method that minimizes the objective function (8). In this study, we refer to the method of optimizing this function as the least-squares approach. As same as the proposed method, the least-squares approach also uses REX^{star}/JGG [38] as a function optimizer. The following recommended values were used for the parameters of REX^{star}/JGG applied in the least-squares approach; the population size n_p is $20s$, the number of children generated per selection n_c is $3s$, and the step-size parameter t is 2.5, where s is the dimension of the search space. Each run was continued until the best objective value did not improved over 1000 generations.

Results and Discussion

This section shows that the proposed method has an ability to estimate parameters of Vohradský's models more effectively.

Inference of a small-scale network

This experiment confirms that the proposed method is capable of estimating reasonable values for the parameters of the Vohradský's model.

Experimental setup: We used the Vohradský's model with 4 genes ($N=4$), that was introduced by the reference [32], as a target network. The model parameters of this system are given in Table 1. Note that, as this network consists of 4 genes, the proposed method solves 4 individual two-dimensional function optimization problems to estimate all of the model parameters.

The observed gene expression patterns, 3 sets of time-series data, each covering 4 genes, were computed from the differential equations (1) on the target model. The sets began from initial values randomly generated in [0.0,1.0], and 50 sampling points for the time-series data were assigned to each gene in each set. The number of observations K is therefore $3\times50=150$. For a practical application, these sets would be obtained by actual biological experiments under different experimental conditions. This experiment simulated no measurement noise in the computed data. The

Table 1. The model parameters for the small-scale target network.

n	$w_{n,1}$	$w_{n,2}$	$w_{n,3}$	$w_{n,4}$	b_n	α_n	β_n
1	20.0	−20.0	0.0	0.0	0.0	0.1	0.1
2	15.0	−10.0	0.0	0.0	−5.0	0.2	0.2
3	0.0	−8.0	12.0	0.0	0.0	0.2	0.2
4	0.0	0.0	8.0	−12.0	0.0	0.2	0.2

doi:10.1371/journal.pone.0083308.t001

time derivatives of the gene expression levels were thus directly computed from the target model. In this study, we estimated the parameters of the target model only from the gene expression levels and their derivatives.

We performed 10 trials, each with different sets of gene expression data. We considered the model parameters to be successfully estimated only when the value of the objective function (6) dropped to less than 1.0×10^{-6}. As the parameters α_n and β_n of our objective function (6) are both positive, this study searched for them in the logarithmic space. Their search area was set to $[-3.0,3.0]^2$. Based on the preliminary experiments, we set the constant parameter C contained in the constrained function minimization problem (7) to 2000. This study used the following recommended values for the parameters of the optimization algorithm, REX^{star}/JGG (see Supporting Information) [38]: the population size n_p is 40, the number of children generated per selection n_c is 6, and the step-size parameter t is 2.5. Each run of REX^{star}/JGG was continued until the maximum number of the generation alternation reached 250. All of the computation were carried out on personal computers using Linux (Fedora release 12). The program was written in C++, and the compiler was gcc 4.4.2.

Results: The proposed method succeeded in estimating the parameter values with precision in 7 trials. Even in the rest of the trials, most of the parameters were correctly estimated. Table 2 shows a sample of the model parameters estimated in one of the failed trials. As mentioned before, the proposed method divided the parameter estimation problem of the target network here into 4 subproblems, each of which is defined as a two-dimensional function optimization problem. In this experiment, our method therefore solved $4\times10=40$ subproblems and failed to find the optimum solutions for only 3 of these 40 subproblems. While the averaged objective value (6) of the 3 failed subproblems was $1.590\times10^{-3}\pm2.120\times10^{-3}$, that of the other subproblems was $4.607\times10^{-9}\pm1.622\times10^{-8}$. In order to estimate all of the model parameters for this network, our method took about 10.4 ± 0.1 minutes on a single-CPU personal computer (Pentium IV 2.8 GHz).

Table 2. A sample of the parameters erroneously estimated by the proposed method in the experiment using the small-scale network.

n	$w_{n,1}$	$w_{n,2}$	$w_{n,3}$	$w_{n,4}$	b_n	α_n	β_n
1	2.745	−0.662	0.090	−0.189	−1.667	0.730	0.457
2	15.002	−10.002	0.000	0.000	−5.000	0.200	0.200
3	0.043	−8.095	11.958	0.072	0.017	0.200	0.200
4	0.000	0.000	8.000	−12.000	0.000	0.200	0.200

Note that the proposed method succeeded in estimating the parameter values with precision in 7 of the 10 trials.

doi:10.1371/journal.pone.0083308.t002

The discrete form of the Vohradský's model can be viewed as a recurrent neural network. The existing inference methods [30–32] have therefore designed on the basis of the learning algorithm for the recurrent neural network, i.e., the back-propagation through time [33]. For the computational simplicity, however, these methods limit the search space to $\alpha_n=\beta_n$. For making a fair comparison, thus, this study constructed two inference methods based on the back-propagation through time, i.e., BPTTLS and BPTTGA, that do not limit the search space. As function optimization algorithms, BPTTLS and BPTTGA used a local search, i.e., the conjugate gradient method [34], and an evolutionary algorithm, i.e., REX^{star}/JGG [38], respectively. In Supporting Information, readers can find more detailed information on these inference methods. We then compared the proposed method with BPTTLS and BPTTGA. The computational costs of these methods were both lower in the small-scale problem described here. They were however unable to estimate the model parameters with precision. In order to estimate all of the model parameters, BPTTLS and BPTTGA required about 0.06 seconds and 9.3 minutes, respectively, on the single-CPU personal computer (Pentium IV 2.8GHz). The averaged objective values of BPTTLS and BPTTGA were $6.381\times10^{0}\pm5.520\times10^{0}$ and $5.987\times10^{-3}\pm1.062\times10^{-2}$, respectively. The objective values of BPTTLS were much worse than those of BPTTGA. Note here that, when we set the model parameters to their optimal values, its objective value is better than those of BPTTGA. These facts indicate that the objective function defined by the back-propagation through time has a lot of local optima. A typical sample of the model parameters estimated by BPTTGA was shown in Table 3. Although the existing inference methods [30–32] limit the search space, on the other hand, they were reportedly still unable to estimate the model parameters with precision.

Inference in noisy environment

Next, we checked the performance of the proposed method in a real-world setting by conducting an experiment with noisy data.

Experimental setup: In the second experiment, we used the Vohradský's models consisting of 10, 20 and 30 genes ($N=10$, 20 and 30) as target networks. As the inference ability of our method might depend on the structure of the target network, we generated the target networks of different structures by changing the model parameters. When trying to determine the model parameters corresponding to the n-th gene, we randomly chose an integer k from a power-law distribution with a cutoff of 5. Then, k genes

Table 3. A typical sample of the parameters estimated by BPTTGA.

n	$w_{n,1}$	$w_{n,2}$	$w_{n,3}$	$w_{n,4}$	b_n	α_n	β_n
1	20.559	−19.907	0.396	−0.309	0.079	0.098	0.100
2	13.422	−8.879	−0.633	1.296	−4.593	0.190	0.189
3	−7.843	0.037	−2.923	−1.354	−0.728	3.000	0.000
4	−0.079	−0.016	7.516	−10.387	0.022	0.161	0.188

BPTTGA is the parameter estimation method of Vohradský's models, that is designed on the basis of the back-propagation through time [33]. As a function optimizer, BPTTGA used an evolutionary algorithm, REX^{star}/JGG [38]. Readers can find more detailed information on BPTTGA in Supporting Information.

doi:10.1371/journal.pone.0083308.t003

were randomly selected from all of the genes contained in the network. The weight parameters $w_{n,m}$'s corresponding to the regulations of the n-th gene from the selected genes were randomly chosen from $[-10.0,-5.0]\cup[5.0,10.0]$, and the rest of the weight parameters were set to 0.0. We also randomly selected the parameters α_n and β_n from [1.0,3.0]. The parameter b_n was set to $-\sum_{m=1}^{N} w_{n,m}$. This study changed the network structure on every trial.

As the performance of the inference method might also depend on the amount of time-series data given, different numbers of time-series datasets were used for the experiments. The time-series datasets were obtained by solving the differential equations (1) on the target networks. The initial values of these sets were selected randomly from [0.0,3.0]. Each dataset consisted of the expression levels at 11 time points with 0.2 time intervals. The measurement noise was simulated by adding 10% Gaussian noise to the computed time-series data. To estimate the time derivatives of the gene expression levels from the given time-series datasets, this experiment used the local linear regression [35], an interpolation technique.

In order to check the performance of the proposed method, this study constructed and then solved 10 genetic network inference problems of each available size with each available number of time-series datasets. In this experiment, we set the constant parameter C to 20. All of the other experimental conditions were the same as those used in the previous experiment.

Results: In the noisy environment, the proposed method was unable to estimate the parameter values with precision. In this experiment, therefore, we only checked whether or not our method infers the structures of the target networks correctly. Note that the Vohradský's model represents the positive and negative regulations from the m-th gene to the n-th gene as positive and negative values, respectively, of the weight parameter $w_{n,m}$. On the other hand, when the m-th gene has no influence on the n-th gene, the value of the parameter $w_{n,m}$ is zero. This study thus extracted the structures of the networks from the estimated model parameters according to the following rules: when $w_{n,m} \geq T_n$ and $w_{n,m} \leq -T_n$, we conclude that the m-th gene positively and negatively, respectively, regulates the n-th gene, where T_n is a threshold; otherwise, we infer no regulation from the m-th gene to the n-th gene. This study set the threshold T_n to

$$T_n = 0.05 \times \max\{|w_{n,1}|,|w_{n,2}|,\cdots,|w_{n,N}|,10^{-3}\}.$$

Figure 2 (a), (b) and (c) show the recalls, the precisions and the specificities of the proposed method on the experiments of solving the inference problems for different sizes with different amounts of gene expression data. The recall, the precision and the specificity are defined as

$$\text{recall} = \frac{TP}{TP+FN}, \quad \text{precision} = \frac{TP}{TP+FP},$$
$$\text{specificity} = \frac{TN}{FP+TN},$$

where TP, FN, FP and TN are the numbers of true-positive, false-negative, false-positive and false-negative regulations, respectively. The figures show that the performances of the proposed method improved with increasing the amount of the given data. A similar experiment has been performed to confirm the performance of the inference method based on the S-system model [23].

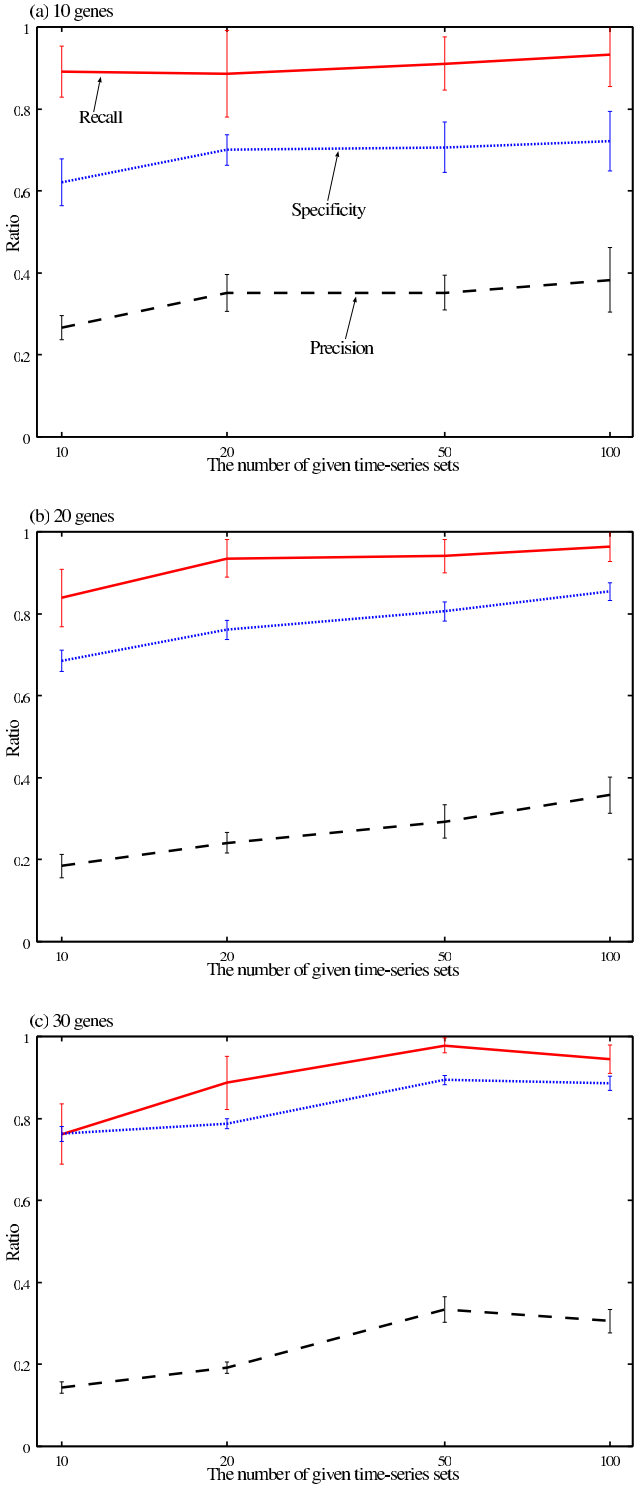

Figure 2. The performances of the proposed method on the experiments of genetic networks consisting of (a) 10 genes, (b) 20 genes, and (c) 30 genes, respectively. Solid, dotted and dashed lines represent the recall, the specificity and the precision, respectively.
doi:10.1371/journal.pone.0083308.g002

These results indicate that the proposed method has the ability to infer a more reasonable network even with a smaller amount of gene expression data. This advantageous feature is due to the smaller number of the parameters of the Vohradský's model. On

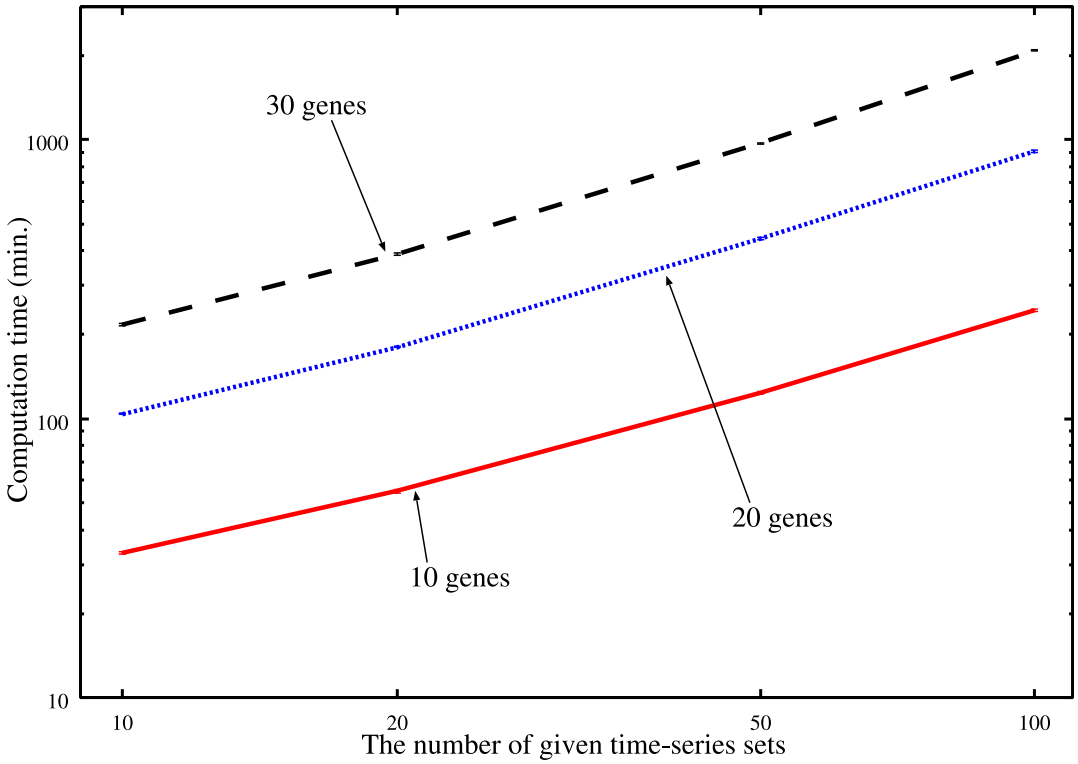

Figure 3. The computation times of the proposed method for noisy experiments. Solid, dotted and dashed lines represents the averaged computation times required for solving the inference problems for 10, 20 and 30 genes, respectively.
doi:10.1371/journal.pone.0083308.g003

the other hand, the computation time required by the proposed method was not always short. The computation times of our method on the single-CPU personal computer (Pentium IV 2.8 GHz) are shown in Figure 3.

The existing inference methods based on the Vohradský's model try to estimate all of the model parameters simultaneously [30–32]. Because of the high-dimensionality in their parameter estimation, it is difficult for them to analyze genetic networks consisting of dozens of genes. When we applied BPTTGA mentioned before to the inference problem of 20 genes with 20 sets of time-series data, therefore, its computation did not finish within 72 hours. Although the computational cost of BPTTLS was still low, on the other hand, its recalls, precisions and specificities were much worse. We thus compared the proposed method only with the least-squares approach described before. Figure 4 shows the precision-recall curves of the proposed method and the least-squares approach on the genetic network inference problems of 20 genes with 20 sets of noisy time-series data. The least-squares approach was performed under different hyper-parameter settings, i.e., $I=0$ and $I=5$. These curves were obtained by changing the hyper-parameter of our method, i.e., C, and that of the least-squares approach, i.e., C_{lsq}. The figure indicates that our method outperforms the least-squares approach with respect to inference ability. However, the computation time of the proposed method was much longer. While the least-squares approach required 20.1 ± 0.6 minutes on the single-CPU personal computer (Pentium IV 2.8 GHz) to infer each network, the proposed method required 182.1 ± 1.2 minutes on the same computer. In the future work, we must therefore develop a way to reduce the computational cost of the proposed method.

In the proposed method, the number of the inferred regulations depends on the value of the hyper-parameter C. Figure 4 indicates, on the other hand, that the quality of the network inferred by the proposed method was quickly degraded with decreasing the number of the inferred regulations. When trying to analyze an actual genetic network, therefore, we should set the hyper-parameter C so that the inferred network contains a larger number of regulations.

Analysis of actual data

We then checked the performance of the proposed method in an experiment using actual gene expression data.

Experimental setup: In this experiment, we analyzed the SOS DNA repair regulatory network in *E.coli* [41]. More than 30 genes are known to be involved in this system. This study however analyzed the expression data of six genes, i.e., *uvrD*, *lexA*, *umuD*, *recA*, *uvrA* and *polB*, which had been measured by Ronen and colleagues [42] ($N=6$). These data have often been used to confirm the performances of the inference methods [7–9,18,19,23,31,32]. The original expression data contain four sets of time-series data. This study however used only two sets (the third and fourth sets), since those two had been measured under the same experimental conditions. Each set of time-series data consisted of 50 measurement values including the initial concentrations of zero. In the experiment, we removed the initial concentrations from both of the sets, as models based on a set of differential equations cannot produce different time-courses from the same initial conditions. The number of measurements K is thus $2 \times 49 = 98$. According to our previous work [9], we normalized the data corresponding to each gene against its maximum expression level. The normalized data were then

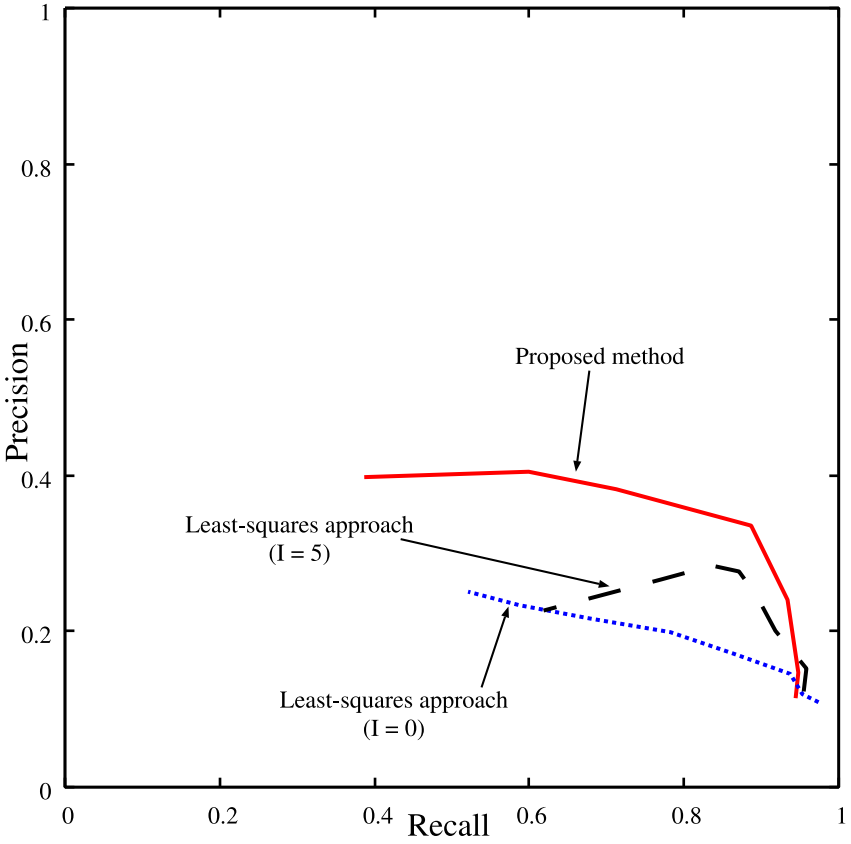

Figure 4. The precision versus the recall for the genetic network inference problems of 20 genes with 20 sets of time-series data. A solid line represents the performances of the proposed method. Dotted and dashed lines represent the performances of the least-squares approach with $I=0$ and $I=5$, respectively.
doi:10.1371/journal.pone.0083308.g004

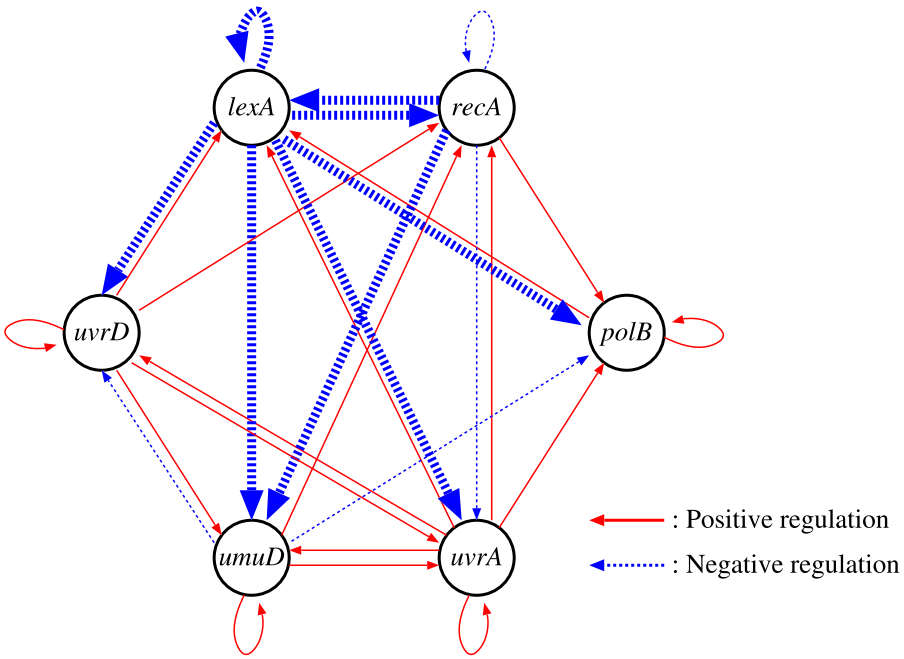

Figure 5. The network structure obtained for the SOS DNA repair system in *E.coli*. Bold lines represent biologically plausible regulations mentioned in the 'Analysis of actual data' section.
doi:10.1371/journal.pone.0083308.g005

Table 4. A sample of the parameters estimated by the proposed method in the experiment with actual gene expression data.

n	$w_{n,1}$	$w_{n,2}$	$w_{n,3}$	$w_{n,4}$	$w_{n,5}$	$w_{n,6}$	b_n	α_n	β_n
1(*uvrD*)	4.412	**−6.461**	−0.354	0.000	5.900	−0.028	−4.183	0.336	0.055
2 (*lexA*)	6.056	**−14.288**	0.000	**−7.567**	19.773	1.668	−4.769	0.163	0.050
3 (*umuD*)	3.291	**−11.751**	7.892	**−5.101**	9.328	0.475	−4.997	0.604	0.095
4 (*recA*)	4.173	**−14.738**	6.996	−3.642	11.510	0.000	−5.195	0.464	0.069
5 (*uvrA*)	5.230	**−20.042**	9.914	−7.364	19.515	0.180	−4.307	0.259	0.246
6 (*polB*)	−0.368	**−4.866**	−17.080	10.910	15.702	7.342	−2.536	0.094	0.110

The parameters written in boldface type correspond to biologically plausible regulations mentioned in the 'Analysis of actual data' section.
doi:10.1371/journal.pone.0083308.t004

smoothed by the local linear regression [35]. We assigned a value of 10^{-6} to expression levels with values of less than 10^{-6}, as the gene expression levels must not be negative.

In this experiment, we set the hyper-parameter C to 2000. All of the other experimental conditions were the same as those described before.

Results: Figure 5 shows the structure of the network inferred by the proposed method. Although we performed 10 trials in this experiment, all of the inferred networks had the same structure. A sample of the parameters estimated by the proposed method is listed in Table 4. As shown in the figure, our method removed few regulations from all of the candidate regulations. Most of the inferred regulations would therefore be false-positive. However, the negative regulations of all of the genes from *lexA* are reasonable, since LexA is known to repress the SOS genes. The negative regulation of *lexA* from *recA* also appears to be reasonable, as RecA senses the damage of DNA and mediates LexA autocleavage. Moreover, the regulation of *umuD* from *recA*, inferred by the proposed method, has been contained in a network now known [15].

As mentioned above, our method seemed to find a number of false-positive regulations. In a future work, therefore, we should find a way to reduce these erroneous regulations.

Experiments on a DREAM3 network

In the experiments described before, we focused on whether the proposed method has an ability to estimate parameters of the Vohradský's models more effectively. Therefore, our experiments have chiefly used the Vohradský's models as the target networks. In the experiments described here, on the other hand, we applied the proposed method to an inference problem whose target network is described as a set of differential equations of the form different from the Vohradský's model.

Experimental setup: The proposed method was applied to one of the artificial genetic network problems obtained from DREAM3 in silico challenges [43]. These problems have been often used to check the performances of genetic network inference methods. This study analyzed the third network, i.e., Yeast1, which consists of 100 genes.

46 sets of time-series data, that were obtained by solving a set of differential equations of the form different from the Vohradský's model, were given as the observed gene expression levels. The given data were polluted by noise. 21 sampling points for time-series data were assigned on each gene in each set. The number of observations K was therefore $46 \times 21 = 966$.

In this experiment, we checked the performances of the proposed method by changing a value for the hyper-parameter C from 20 to 2000. All of the other experimental conditions were the same as those described before.

Results: The performances of the proposed method on the DREAM3 problem are shown in Table 5. The network inferred by the proposed method tells us whether the regulation of the n-th gene from the m-th gene is positive or negative. As the correct network given by the DREAM3 does not have the information about the types of the regulations, however, we omitted to check the types of the regulations. Compared with the champion algorithm of the DREAM3 challenges [44], the performances of our method were worse. Note however that this study confirmed the effectiveness of the proposed method through the experiments on the actual genetic network inference problem. Thus, the experimental results shown here do not always prove the inability of the proposed method in analyzing actual gene expression data. The Vohradský's model would be unsuitable to capture the features of the DREAM3 network. One of the reasons of the unsuitability would be that, while the model used in the DREAM3 problem considers the effect of the intrinsic noise [45], the Vohradský's model does not consider it. The intrinsic noise is unavoidable in biological processes, and the analysis of it would be important to understand biological systems. However, the current technologies generally measure the gene expression levels averaged over a lot of cells. As we think that the averaged gene expression levels weaken the effect of the intrinsic noise, the method proposed in this study infers genetic networks without considering it.

Table 5. The performances of the proposed method on the third problem of the DREAM3 in silico challenges [43].

C	FP	FN	TP	TN	recall	precision	specisificy
20	15	153	13	9719	0.078	0.464	0.998
50	78	131	35	9656	0.211	0.310	0.992
100	312	112	54	9422	0.325	0.148	0.968
150	632	95	71	9102	0.428	0.101	0.935
200	876	91	75	8858	0.452	0.079	0.910
500	1948	84	82	7786	0.494	0.040	0.800
2000	4009	69	97	5725	0.584	0.024	0.588

The performances were checked by changing the hyper-parameter of the proposed method, C. FP, FN, TP and TN are the numbers of false-positive, false-negative, true-positive, true-negative regulations, respectively.
doi:10.1371/journal.pone.0083308.t005

Conclusion

This study proposed a new method for the inference of Vohradský's models of genetic networks. The proposed method resolves the difficulty in the estimation of the model parameters by defining it as two-dimensional function optimization problems. The experimental results indicated that our method has an ability to estimate reasonable values for the parameters of the Vohradský's model. However, the computation time of the proposed method is not always short. In the future work, therefore, we must develop a technique to reduce its computational cost.

A variety of inference methods based on a variety of mathematical models have been proposed. However, we still do not know which method is the most suitable for the inference of genetic networks. In order to obtain a reliable network, therefore, it will be important to analyze the measurement data using multiple inference methods based on models different from each other. The inference method proposed in this study may be a promising choice for this purpose.

Supporting Information

Text S1 Detailed algorithms of REX^{star}/JGG and the least-squares approach.
(PDF)

Text S2 Time-series data used in the 'Inference in noisy environment' section (compressed by gzip).
(GZ)

Acknowledgments

A software, BPMPD, developed by Dr. C. Mészáros at MTA SZTAKI was used to solve the linear programming problems in this study.

Author Contributions

Conceived and designed the experiments: SK. Performed the experiments: SK. Analyzed the data: SK MS MOH. Contributed reagents/materials/analysis tools: SK. Wrote the paper: SK.

References

1. Akutsu T, Miyano S, Kuhara S (2000) Inferring qualitative relations in genetic networks and metabolic pathways. Bioinformatics 16: 727–734.
2. Bansal M, di Bernardo D (2007) Inference of gene networks from temporal gene expression profiles. IET Systems Biology 1: 306–312.
3. Chou IC, Voit EO (2009) Recent development in parameter estimation and structure identification of biochemical and genomic systems. Mathematical Biosciences 219: 57–83.
4. D'haeseleer P, Liang S, Somogyi R (2000) Genetic network inference: From co-expression clustering to reverse engineering. Bioinformatics 16: 707–726.
5. Ergün A, Lawrence CA, Kohanski MA, Brennan TA, Collins JJ (2007) A network biology approach to prostate cancer. Molecular Systems Biology 3: 82.
6. Faith JJ, Hayete B, Thaden JT, Mogno I, Wierzbowski J, et al. (2007) Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles. PLoS Biology 5: e8.
7. Kabir S, Noman N, Iba H (2010) Reverse engineering gene regulatory network from microarray data using linear time-variant model. BMC Bioinformatics 11: S56.
8. Kimura S, Sonoda K, Yamane S, Maeda H, Matsumura K, et al. (2008) Function approximation approach to the inference of reduced ngnet models of genetic networks. BMC Bioinformatics 9:23.
9. Kimura S, Nakayama S, Hatakeyama M (2009) Genetic network inference as a series of discrimination tasks. Bioinformatics 25: 918–925.
10. Margolin AA, Nemenman I, Basso K, Wiggins C, Stolovitzky G, et al. (2006) Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. BMC Bioinformatics 7: S7.
11. Tucker W, Kutalik Z, Moulton V (2007) Estimating parameters for generallized mass action models using constraint propagation. Mathematical Biosciences 208: 607–620.
12. Veflingstad SR, Almeida J, Voit EO (2004) Priming nonlinear searches for pathway identification. Theoretical Biology and Medical Modelling 1: 8.
13. Yeung MKS, Tegnér J, Collins JJ (2002) Reverse engineering gene networks using singular value decomposition and robust regression. Proc National Academy of Sciences of USA 99: 6163–6168.
14. Yu J, Smith VA, Wang PP, Hartemink J, Jarvis ED (2004) Advances to bayesian network inference for generating causal networks from observational biological data. Bioinformatics 20: 3594–3603.
15. Gardner TS, di Bernardo D, Lorenz D, Collins JJ (2003) Inferring genetic networks and identifying compound mode of action via expression profiling. Science 301: 102–105.
16. Savageau MA (1969) Biochemical systems analysis i. some mathematical properties of the rate law for the component enzymatic reactions. Journal of Theoretical Biology 25: 365–369.
17. Voit EO (2000) Computational Analysis of Biochemical Systems. Cambridge: Cambridge University Press.
18. Chemmangattuvalappil N, Task K, Banerjee I (2012) An integer optimization algorithm for robust identification of non-linear gene regulatory networks. BMC Systems Biology 6: 119.
19. Cho DY, Cho KH, Zhang BT (2006) Identification of biochemical networks by s-tree based genetic programming. Bioinformatics 22: 1631–1640.
20. Gonzalez OR, Küper C, Jung K, Naval Jr PC, Mendoza E (2007) Parameter estimation using simulated annealing for s-system models of biochemical networks. Bioinformatics 23: 480–486.
21. Kikuchi S, Tominaga D, Arita M, Takahashi K, Tomita M (2003) Dynamic modeling of genetic networks using genetic algorithm and s-system. Bioinformatics 19: 643–650.
22. Kimura S, Ide K, Kashihara A, Kano M, Hatakeyama M, et al. (2005) Inference of s-system models of genetic networks using a cooperative coevolutionary algorithm. Bioinformatics 21: 1154–1163.
23. Kimura S, Araki D, Matsumura K, Okada-Hatakeyama M (2012) Inference of s-system models of genetic networks by solving one-dimensional function optimization problems. Mathematical Biosciences 235: 161–170.
24. Liu PK, Wang FS (2008) Inference of biochemical network models in s-system using multiobjective optimization approach. Bioinformatics 24: 1085–1092.
25. Nakatsui M, Ueda T, Maki Y, Ono I, Okamoto M (2008) Method for inferring and extracting reliable genetic interactions from time-series profile of gene expression. Mathematical Biosciences 215: 105–114.
26. Tsai KY, Wang FS (2005) Evolutionary optimization with data collocation for reverse engineering of biological networks. Bioinformatics 21: 1180–1188.
27. Vilela M, Chou IC, Vinga S, Vasconcelos ATR, Voit EO, et al. (2008) Parameter optimization in s-system models. BMC Systems Biology 2: 35.
28. Voit EO, Almeida J (2004) Decoupling dynamical systems for pathway identification from metabolic profiles. Bioinformatics 20: 1670–1681.
29. Vohradský J (2001) Neural network model of gene expression. FASEB Journal 15: 846–854.
30. Palafox L, Iba H (2012) On the use of population based incrementasl learning to do reverse engineering on gene regulatory networks. In: Proc. 2012 Congress on Evolutionary Computation: 2012; Brisbane. pp. 1865–1872.
31. Xu R, Venayagamoorthy GK, Wunsch II DC (2007) Inference of gene regulatory networks with hybrid differential evolution and particle swarm optimization. Neural Networks 20: 917–927.
32. Xu R, Wunsch II DC, Frank RL (2007) Inference of genetic regulatory networks with recurrent neural network models using particle swarm optimization. IEEE/ACM Transactions on Computational Biology and Bioinformatics 4: 681–692.
33. Williams RJ, Peng J (1990) An efficient gradient-based algorithm for on-line training of recurrent network trajectories. Neural Computation 2: 490–501.
34. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1995) Numerical Recipes in C, 2nd Edition. Cambridge: Cambridge University Press.
35. Cleveland WS (1979) Robust locally weight regression and smoothing scatterplots. Journal of American Statistical Association 79: 829–836.
36. Vilela M, Borges CCH, Vinga S, Vasconcelos ATR, Santos H, et al. (2007) Automated smoother for the numerical decoupling of dynamics models. BMC Bioinformatics 3: 305.
37. Chou IC, Martens H, Voit EO (2006) Parameter estimation in biochemical systems models with alternating regression. Theoretical Biology and Medical Modelling 3: 25.
38. Kobayashi S (2009) The frontiers of real-coded genetic algorithms (in japanese). Transactions of the Japanese Society for Artificial Intelligence 24: 147–162.
39. Thieffry D, Huerta AM, Pérez-Rueda E, Collado-Vides J (1998) From specific gene regulation to genomic networks: a global analysis of transcriptional regulation in escherichia coli. BioEssays 20: 433–440.
40. Mehrotra S (1992) On the implementation of a primal-dual interior point method. SIAM Journal on Optimization 2: 575–601.
41. Sutton MD, Smith BT, Godoy VG, Walker GC (2000) The sos response: Recent insights into umudc-dependent mutagenesis and dna damage tolerance. Annual Review of Genetics 34: 479–497.
42. Ronen M, Rosenberg R, Shraiman BI, Alon U (2002) Assigning numbers to the arrows: Parameterizing a gene regulation network by using accurate expression kinetics. Proc National Academy of Sciences of USA 99: 10555–10560.
43. Dream project. available: http://wiki.c2b2.columbia.edu/dream/index.php/the dream project. Accessed 2013 Nov 13.

44. Prill RJ, Marbach D, Saez-Rodriguez J, Sorger PK, Alexopoulos LG, et al. (2010) Towards a rigorous assessment of systems biology models: The dream3 challanges. PLoS One 5: e9202.

45. Schaffter T, Marbach D, Floreano D (2011) Genenetweaver: in silico benchmark generation and performance profiling of network inference methods. Bioinformatics 27: 2263–2270.