JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 21, Number 11, 2014

Pp. 785–798
DOI: 10.1089/cmb.2014.0171

Research Articles

An Efficient Data Assimilation Schema for Restoration and Extension of Gene Regulatory Networks Using Time-Course Observation Data

TAKANORI HASEGAWA,[1] TOMOYA MORI,[1] RUI YAMAGUCHI,[2] SEIYA IMOTO,[2] SATORU MIYANO,[2] and TATSUYA AKUTSU[1]

ABSTRACT

Gene regulatory networks (GRNs) play a central role in sustaining complex biological systems in cells. Although we can construct GRNs by integrating biological interactions that have been recorded in literature, they can include suspicious data and a lack of information. Therefore, there has been an urgent need for an approach by which the validity of constructed networks can be evaluated; simulation-based methods have been applied in which biological observational data are assimilated. However, these methods apply nonlinear models that require high computational power to evaluate even one network consisting of only several genes. Therefore, to explore candidate networks whose simulation models can better predict the data by modifying and extending literature-based GRNs, an efficient and versatile method is urgently required. We applied a combinatorial transcription model, which can represent combinatorial regulatory effects of genes, as a biological simulation model, to reproduce the dynamic behavior of gene expressions within a state space model. Under the model, we applied the unscented Kalman filter to obtain the approximate posterior probability distribution of the hidden state to efficiently estimate parameter values maximizing prediction ability for observational data by the EM-algorithm. Utilizing the method, we propose a novel algorithm to modify GRNs reported in the literature so that their simulation models become consistent with observed data. The effectiveness of our approach was validated through comparison analysis to the previous methods using synthetic networks. Finally, as an application example, a Kyoto Encyclopedia of Genes and Genomes (KEGG)-based yeast cell cycle network was extended with additional candidate genes to better predict the real mRNA expressions data using the proposed method.

Key words: biological simulation, gene regulatory networks inference, time-series analysis.

1. INTRODUCTION

Intracellular systems in cells consist of many genetic and chemical interactions, and gene regulatory networks (GRNs) play a crucial role in sustaining such systems. Although comprehensive

[1]Bioinformatics Center, Institute for Chemical Research, Kyoto University, Gokasho, Uji, Kyoto, Japan.
[2]Human Genome Center, The Institute of Medical Science, The University of Tokyo, Minato-ku, Tokyo, Japan.

understanding of GRNs is still lacking, much data have been recorded in the literature following recent advances in biotechnology, for example, microarray and Chip-Seq. Thus, by integrating these findings, we may be able to reconstruct GRNs and understand the dynamic behavior of gene expression through mathematical simulation models. However, since some unverified interactions are present in the literature, simulation results may not match the observed data, for example, microarray expression data. In this respect, a method for finding candidate networks that are consistent with the data by improving and extending literature-based models is needed to elucidate GRNs (Hasegawa et al., 2011, 2014; Nakajima et al., 2012).

In order to construct simulation models of GRNs, interactions between biomolecules, for example, mRNA and proteins, are firstly collected from the literature and are integrated to construct the networks. Then, mathematical differential or difference equations are given to the constructed networks to simulate the dynamic behavior of these biomolecules. Thus, biologically reliable models, for example, the Michaelis-Menten model (Savageau, 1969) and S-system (Savageau and Voit, 1987), described by differential equations, have been applied in dealing with the limited number of genes (Murtuza Baker et al., 2013; Hasegawa et al., 2011; Liu and Niranjan, 2012; Rogers et al., 2007; Quach et al., 2007). In these approaches, a simulation-based methodology, called data assimilation, was employed for estimating parameter values and evaluating such simulation models (Nagasaki et al., 2006; Nakamura et al., 2009). However, although simulation results generated from these models can be biologically reasonable, evaluation of even one simulation model with estimating optimal parameter values is computationally demanding since parameter estimation must rely on a type of Monte Carlo methodology (Murtuza Baker et al., 2013; Julier and Uhlmann, 1997; Kitagawa, 1998; Koh et al., 2010). Therefore, it is computationally implausible to find appropriate models from a large number of candidate models.

In contrast to such approaches, in order to cope with the computational burden known as the curse of dimensionality in applying mathematical models to elucidate GRNs, there exists the other approach to use linear models for dealing with more than a hundred genes. In this approach, many effective methods have been developed, for example, state space models (Beal et al., 2005; Hirose et al., 2008; Rangel et al., 2004) and Bayesian inference (Friedman et al., 2007; Mahdi et al., 2012; Watanabe et al., 2012). For restoring literature-based GRNs, a concept, called network completion, has also been developed (Akutsu et al., 2009; Nakajima et al., 2012). However, these methods could fail in some cases, for example, handling non-equally spaced time-point data, because of simplified abstractions of biological systems. Thus, since these models cannot adequately represent the dynamics of gene expression due to simplified abstractions of biological systems, biologically invalid results might often be obtained. For improving and extending literature-based GRNs, these models are not sufficient because the number of genes is limited and their regulatory relationships are mostly reliable.

Here, applied simulation models should maximally emulate reliable biological dynamics under the constraint that their parameter values can be efficiently estimated. To satisfy the requirements, we developed a new data assimilation schema that applies a simple nonlinear simulation model, termed the combinatorial transcription model (Opper and Sanguinetti, 2010; Wang et al., 2005). As a part of this schema, we applied the unscented Kalman filter (Julier and Uhlmann, 1997, 2004; Chow et al., 2007) to obtain approximate posterior probability distributions of the hidden state and estimated parameter values maximizing prediction ability for observational data by means of EM-algorithm. Then, a novel algorithm was developed to efficiently select and evaluate a candidate network to obtain a network that can best predict the data within a framework of the nonlinear state space model.

To show the effectiveness of the proposed method, we performed a comparison using artificial data in regard to a previously proposed network completion method (Nakajima et al., 2012). For the comparison, synthetic data with equally and non-equally spaced time-points were generated from WNT5A (Kim et al., 2002) and a yeast cell cycle network (Kanehisa et al., 2012). Next, as real data experiments, a yeast cell-cycle network from KEGG database (Kanehisa et al., 2012) and candidate genes from The Saccharomyces Genome Database (SGD) (Cherry et al., 2012), which can have functions related to this network, were integrated to extend the network using real mRNA expression data (Spellman et al., 1998).

2. METHOD

2.1. Combinatorial model for gene regulatory networks

Let $x_i(t)$ be the abundance of the ith ($i = 1, \ldots, p$) gene as a function of time t. As a gene regulatory model, we assume a system in which each gene undergoes synthesis and degradation processes, and its

expression value can be controlled through regulations of its synthesis process by other genes. Thus, $x_i(t)$ is determined by

$$\frac{dx_i(t)}{dt} = f_i(\boldsymbol{x}(t), \boldsymbol{\theta}) \cdot u_i - x_i(t) \cdot d_i + v_{i,t}, \tag{1}$$

where f_i is a function of the regulatory effect on the ith gene, $\boldsymbol{x}(t) = (x_1(t), \ldots, x_p(t))'$, $\boldsymbol{\theta}$ is a tuning parameter, u_i is synthesis coefficient, d_i is a degradation coefficient, and $v_{i,t}$ is a system noise at time t. Typically, f_i is represented by a hill function, such as the Michaelis-Menten model (Savageau, 1969).

Due to its heavy computational cost to estimate parameter values maximizing prediction ability for data, Equation (1) is often approximated as a difference equation. Then, we apply the combinatorial transcription model (Opper and Sanguinetti, 2010; Wang et al., 2005) as

$$x_{i,t+\Delta t} = x_{i,t} + \left(\sum_{j \in \mathcal{A}_i} a_{i,j} \cdot x_{j,t} + \sum_{j \in \mathcal{A}_i} \sum_{k \in \mathcal{A}_i \setminus j} b_{i,(j,k)} \cdot x_{j,t} \cdot x_{k,t} + u_i - x_{i,t} \cdot d_i + v_{i,t} \right) \cdot \Delta t, \tag{2}$$

where $x_{i,t}$ is the amount of the ith gene at time t, $a_{i,j}$ is an individual effect of the jth gene on the ith gene, $b_{i,(j,k)}$ is a combinatorial effect from the jth and the kth genes to the ith gene, $\mathcal{A}_i$ is an active set of genes regulating the ith gene, and Δt is a minute displacement. Here, we set $\Delta t = 1$ (:a minimum observational interval) for simplicity. Figure 1 exemplifies this model.

In order to assimilate a simulation model and observational data, we apply a nonlinear state space model (Asif and Sanguinetti, 2011; Hirose et al., 2008; Kojima et al., 2010; Lillacci and Khammash, 2010; Quach et al., 2007). Let $\boldsymbol{x}_t = (x_{1,t}, \ldots, x_{p,t})'$ be the vector of hidden variables and $\boldsymbol{y}_t$ be the observational data at time t. A state space representation of Equation (2) is given by

$$\boldsymbol{x}_{t+1} = A\boldsymbol{x}_t + B\mathrm{vec}(\boldsymbol{x}_t \boldsymbol{x}_t') + \boldsymbol{u} + \boldsymbol{v}_t, \tag{3}$$

$$\boldsymbol{y}_t = \boldsymbol{x}_t + \boldsymbol{w}_t, \tag{4}$$

where $A \in R^{p \times p}$ is a linear effect matrix, $B \in R^{p \times p^2}$ is a combinatorial effect matrix, $vec(\cdot)$ is a transformation function ($R^{p \times p} \rightarrow R^{p^2}$), $\boldsymbol{u} = (u_1, \ldots, u_p)'$, $\boldsymbol{v}_t \sim N(0, Q)$, and $\boldsymbol{w}_t \sim N(0, R)$ are system noise and observational noise with diagonal covariance matrices, respectively. Note that A and B should be sparse matrices according to $\mathcal{A}_i$, and that $\boldsymbol{d}_i$ is included in A. We define an entire set of time points $\mathcal{T} = \{1, \ldots, T\}$ and the observed time set $\mathcal{T}_{obs}$ ($\mathcal{T}_{obs} \subset \mathcal{T}$).

2.2. Unscented Kalman filter

In Equations (3) and (4), conditional probability densities $P(\boldsymbol{x}_t|Y_{t-1})$, $P(\boldsymbol{x}_t|Y_t)$, and $P(\boldsymbol{x}_t|Y_T)$ can be non-Gaussian forms, where $Y_t = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t)$. Therefore, we applied the unscented Kalman filter (UKF) (Julier and Uhlmann, 1997, 2004; Chow et al., 2007) to approximately obtain these conditional probability densities. The procedure is explained below.

2.2.1. Prediction and filtering steps. Let $\boldsymbol{x}_{t|s}$ and $V_{t|s}$ be the expectation and the covariance matrix, given observational data Y_s at time t. For $t = 0, \ldots, T-1$,

1. Select sigma points $\boldsymbol{x}_{t|t}^{(n)} (n = 0, \ldots, 2p)$ as

$$\boldsymbol{x}_{t|t}^{(n)} = \boldsymbol{x}_{t|t}, \qquad (n = 0), \tag{5}$$

$$\boldsymbol{x}_{t|t}^{(n)} = \boldsymbol{x}_{t|t} + \sqrt{(p+\lambda)\Sigma_{t|t}^{(n)}}, \qquad (n = 1, \ldots, p), \tag{6}$$

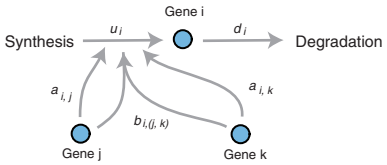

FIG. 1. An example of the combinatorial transcription model regarding the ith gene. A gene undergoes synthesis and degradation processes, and its synthesis process is regulated through individual effects $a_{i,j}$,$a_{i,k}$, and a combinatorial effect $b_{i,(j,k)}$.

$$\boldsymbol{x}_{t|t}^{(n)} = \boldsymbol{x}_{t|t} - \sqrt{(p+\lambda)\Sigma_{t|t}^{(n-p)}}, \qquad (n = p+1, \ \ldots, 2p), \tag{7}$$

where $\Sigma_{t|t}^{(n)}$ is the nth column vector of $\Sigma_{t|t}$ and $\lambda = \alpha^2(p + \kappa) - p$. Here, $\alpha^2 = 3/10$ and $\kappa = 0$ were applied to set $p + \lambda = 3$ (Julier et al., 2000).

2. Predict the next state of the generated sigma points $\boldsymbol{x}_{t|t}^{(n)}$ as $\boldsymbol{x}_{t+1|t}^{(n)}$ using the system equation of Equation (3) without adding the system noise.
3. Calculate $\boldsymbol{x}_{t+1|t}$ and $V_{t+1|t}$ as

$$\boldsymbol{x}_{t+1|t} = \sum_{n=0}^{2p} \mathcal{W}_1^{(n)} \boldsymbol{x}_{t+1|t}^{(n)}, \tag{8}$$

$$\Sigma_{t+1|t} = \sum_{n=0}^{2p} \mathcal{W}_2^{(n)} (\boldsymbol{x}_{t+1|t}^{(n)} - \boldsymbol{x}_{t+1|t})(\boldsymbol{x}_{t+1|t}^{(n)} - \boldsymbol{x}_{t+1|t})' + Q, \tag{9}$$

$$\mathcal{W}_1^{(0)} = \frac{\lambda}{p+\lambda}, \tag{10}$$

$$\mathcal{W}_2^{(0)} = \frac{\lambda}{p+\lambda} + 1 - \alpha^2 + \beta, \tag{11}$$

$$\mathcal{W}_1^{(n)} = \mathcal{W}_2^{(n)} = \frac{1}{2(p+\lambda)}, \ \ (n = 1, \ \ldots, 2p), \tag{12}$$

where β is set 2 (Julier, 2002).

4. In the combinatorial model, the observational equation of Equation (4) is a linear function. Then, we can apply the general Kalman filter algorithm (Kojima et al., 2010; Kalman, 1960) to obtain the optimal conditional expectation and covariance matrix as follows

$$\boldsymbol{x}_{t+1|t+1} = \boldsymbol{x}_{t+1|t} + \Sigma_{t+1|t+1} R^{-1} (\boldsymbol{y}_{t+1} - \boldsymbol{x}_{t+1|t}), \tag{13}$$

$$\Sigma_{t+1|t+1} = (R^{-1} + \Sigma_{t+1|t}^{-1})^{-1}. \tag{14}$$

More details can be referred to in Julier and Uhlmann (1997, 2004).

2.2.2. Smoothing step. In order to obtain the conditional expectation and covariance matrix of the hidden state given full observational data Y_T, we apply the Rauch-Tung-Striebel (RTS) smoother for UKF (Sarkka, 2008). The formulation of the RTS smoother is described as follows:

$$\boldsymbol{x}_{t|T} = \boldsymbol{x}_{t|t} + K_t (\boldsymbol{x}_{t+1|T} - \boldsymbol{x}_{t+1|t-1}), \tag{15}$$

$$\Sigma_{t|T} = \Sigma_{t|t} + K_t (\Sigma_{t+1|T} - \Sigma_{t+1|t-1}) K_t', \tag{16}$$

$$K_t = C_t \Sigma_{t+1|t}^{-1}, \tag{17}$$

$$C_t = \sum_{n=0}^{2p} \mathcal{W}_2^{(n)} (\boldsymbol{x}_{t|t-1}^{(n)} - \boldsymbol{x}_{t|t-1})(\boldsymbol{x}_{t+1|t}^{(n)} - \boldsymbol{x}_{t+1|t})'. \tag{18}$$

Since we have $\boldsymbol{x}_{T|T}$ and $\Sigma_{T|T}$ after prediction and filtering steps, the above equations are recursively applied for $t = T-1, \ \ldots, 0$.

2.3. Parameter estimation using EM-algorithm

Let $X_T = \{\boldsymbol{x}_0, \ \ldots, \boldsymbol{x}_T\}$ be the set of state variables, and $\boldsymbol{\theta} = \{A, B, \boldsymbol{u}, Q, R, \boldsymbol{\mu}_0\}$ be the parameter vector. The log-likelihood of observational data is given by

$$\log L = \log \int P(\boldsymbol{x}_0) \prod_{t \in \mathcal{T}} P(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}) \prod_{t \in \mathcal{T}_{obs}} P(\boldsymbol{y}_t | \boldsymbol{x}_t) d\boldsymbol{x}_1 \ldots d\boldsymbol{x}_T, \tag{19}$$

where $P(\boldsymbol{x}_0)$ is a probability density of N-dimensional Gaussian distributions $N(\boldsymbol{\mu}_0, \Sigma_0)$, $P(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ and $P(\boldsymbol{y}_t|\boldsymbol{x}_t)$ can be probability densities of N-dimensional non-Gaussian distributions approximated by Equations (3) and (4) in section 2.1 and the unscented transformation in section 2.2.

In this article, we attempted to estimate the parameter vector $\boldsymbol{\theta}$ by maximizing Equation (19) using the EM-algorithm (Dempster et al., 1977). Thus, the conditional expectation of the joint log-likelihood of the complete data (X_T,Y_T) at the lth iteration,

$$q(\boldsymbol{\theta}|\boldsymbol{\theta}_l) = E[\log P(Y_T, X_T|\boldsymbol{\theta})|Y_T, \boldsymbol{\theta}_l], \tag{20}$$

is iteratively maximized with respect to $\boldsymbol{\theta}$ until convergence.

In the expectation step, set conditional expectations of $\boldsymbol{x}_t$ as

$$V_t = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_2^{(n)} \boldsymbol{x}_{t|T}^{(n)} \boldsymbol{x}_{t|T}^{(n)'}, \tag{21}$$

$$V_{lag} = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_2^{(n)} \boldsymbol{x}_{t|T}^{(n)} \boldsymbol{x}_{t-1|T}^{(n)'}, \tag{22}$$

$$V_{t-1} = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_2^{(n)} \boldsymbol{x}_{t-1|T}^{(n)} \boldsymbol{x}_{t-1|T}^{(n)'}, \tag{23}$$

$$\Phi_{lag} = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_2^{(n)} \boldsymbol{x}_{t|T}^{(n)} \mathrm{vec}(\boldsymbol{x}_{t-1|T}^{(n)} \boldsymbol{x}_{t-1|T}^{(n)'})', \tag{24}$$

$$\Phi_{t-1} = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_2^{(n)} \boldsymbol{x}_{t-1|T}^{(n)} \mathrm{vec}(\boldsymbol{x}_{t-1|T}^{(n)} \boldsymbol{x}_{t-1|T}^{(n)'})', \tag{25}$$

$$\Psi_{t-1} = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_2^{(n)} \mathrm{vec}(\boldsymbol{x}_{t-1|T}^{(n)} \boldsymbol{x}_{t-1|T}^{(n)'}) \mathrm{vec}(\boldsymbol{x}_{t-1|T}^{(n)} \boldsymbol{x}_{t-1|T}^{(n)'})', \tag{26}$$

$$\boldsymbol{s}_t = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_1^{(n)} \boldsymbol{x}_{t|T}^{(n)}, \tag{27}$$

$$\boldsymbol{s}_{t-1} = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_1^{(n)} \boldsymbol{x}_{t-1|T}^{(n)}, \tag{28}$$

$$\boldsymbol{s}_{t-1}^2 = \sum_{t\in\mathcal{T}}\sum_{n=0}^{2p} \mathcal{W}_1^{(n)} \mathrm{vec}(\boldsymbol{x}_{t-1|T}^{(n)} \boldsymbol{x}_{t-1|T}^{(n)'})'. \tag{29}$$

In the maximization step, $\boldsymbol{\theta}_l$ is updated to $\boldsymbol{\theta}_{l+1} = \arg\max_{\boldsymbol{\theta}} q(\boldsymbol{\theta}|\boldsymbol{\theta}_l)$. Let $\boldsymbol{v}_{lag,i}$, $\boldsymbol{\phi}_{lag,i}$, and $\boldsymbol{\phi}_{t\text{-}1,i}$ be a transpose of the ith row vector of V_{lag}, Φ_{lag}, and Φ_{t-1}, respectively. Then, $\boldsymbol{\theta}$ is updated as

$$\boldsymbol{a}_i^{\mathcal{A}} = V_{t-1}^{\mathcal{A}^{-1}} (\boldsymbol{v}_{lag,\,i}^{\mathcal{A}} - \phi_{t-1}^{\mathcal{A}\times\mathcal{B}} \boldsymbol{b}_i^{\mathcal{B}} - u_i \boldsymbol{s}_{t-1}^{\mathcal{A}}), \tag{30}$$

$$\boldsymbol{b}_i^{\mathcal{B}} = \Psi_{t-1}^{\mathcal{B}^{-1}} (\phi_{lag,\,i}^{\mathcal{B}} - \phi_{t-1}^{\mathcal{A}\times\mathcal{B}'} \boldsymbol{a}_i^{\mathcal{A}} - u_i \boldsymbol{s}^{2^{\mathcal{B}}}), \tag{31}$$

$$\boldsymbol{u} = \frac{\boldsymbol{s}_t - A\boldsymbol{s}_{t-1} - B\boldsymbol{s}^2}{T}, \tag{32}$$

$$Q = \frac{1}{T}\sum_{t=1}^{T} E[(\boldsymbol{x}_t - A\boldsymbol{x}_{t-1} - B\mathrm{vec}(\boldsymbol{x}_{t-1}\boldsymbol{x}_{t-1}') - \boldsymbol{u}) \cdot (\boldsymbol{x}_t - A\boldsymbol{x}_{t-1} - B\mathrm{vec}(\boldsymbol{x}_{t-1}\boldsymbol{x}_{t-1}') - \boldsymbol{u}')|Y_T], \tag{33}$$

$$\boldsymbol{\mu}_0 = \boldsymbol{x}_{0|T}, \tag{34}$$

$$R = \frac{1}{T}\sum_{t\in\mathcal{T}_{obs}} \{(\boldsymbol{y}_t - \boldsymbol{x}_{t|T})(\boldsymbol{y}_t - \boldsymbol{x}_{t|T})' + \Sigma_{t|T}\}, \tag{35}$$

where $\mathcal{A}$ and $\mathcal{B}$ are active sets of elements for A and B, respectively. For example, $\boldsymbol{a}_i^{\mathcal{A}}$ is an $|\mathcal{A}|$-dimensional vector consisting of elements regulating the ith gene.

2.4. Network restoration algorithm

When an original gene regulatory network $\mathcal{M}_{original}$ is given, the purpose is to find the model $\mathcal{M}_{best}$ that can best predict observational data. Here, the prediction ability of a model $\mathcal{M}$ is evaluated using the Bayesian information criterion (BIC) (Schwarz, 1978) described by

$$BIC = 2 \log L - D \log \nu, \tag{36}$$

where D and v are the number of samples and the nonzero parameters, respectively. Because of the high computational cost involved in estimating the values of the parameters $\boldsymbol{\theta}$ for $\mathcal{M}$, we can not evaluate all candidate models. Therefore, starting from $\mathcal{M}_{original}$, one strategy is to sequentially evaluate candidate models that are constructed by changing a part of the regulatory structure of the current model $\mathcal{M}_{current}$ of which prediction ability is the best among evaluated ones. In this paradigm, we consider three operations, that is, adding, deleting, and replacing a regulation, which are shown in Figure 2, and the constraints add_{max} and del_{max}, which restrict the number of additional and deleted regulations from $\mathcal{M}_{original}$. Then, we propose a novel algorithm, which can efficiently evaluate only highly possible candidates, for improving and extending GRNs to obtain $\mathcal{M}_{best}$ as concluded in Algorithms 1–3 (2 and 3 are sub-algorithms for Algorithm 1). In these algorithms, we consider a function for measuring the possibility of the model $\mathcal{M}$ that added or deleted a regulation to the ith gene from $\mathcal{M}_{current}$ as

$$e(\mathcal{M}, i) = \boldsymbol{a}_i' V_{t-1} \boldsymbol{a}_i - 2\boldsymbol{v}_{lag,\,i} \boldsymbol{a}_i + 2\boldsymbol{b}_i' \phi_{t-1}' \boldsymbol{a}_i + 2u_i \boldsymbol{s}_{t-1}' \boldsymbol{a}_i. \tag{37}$$

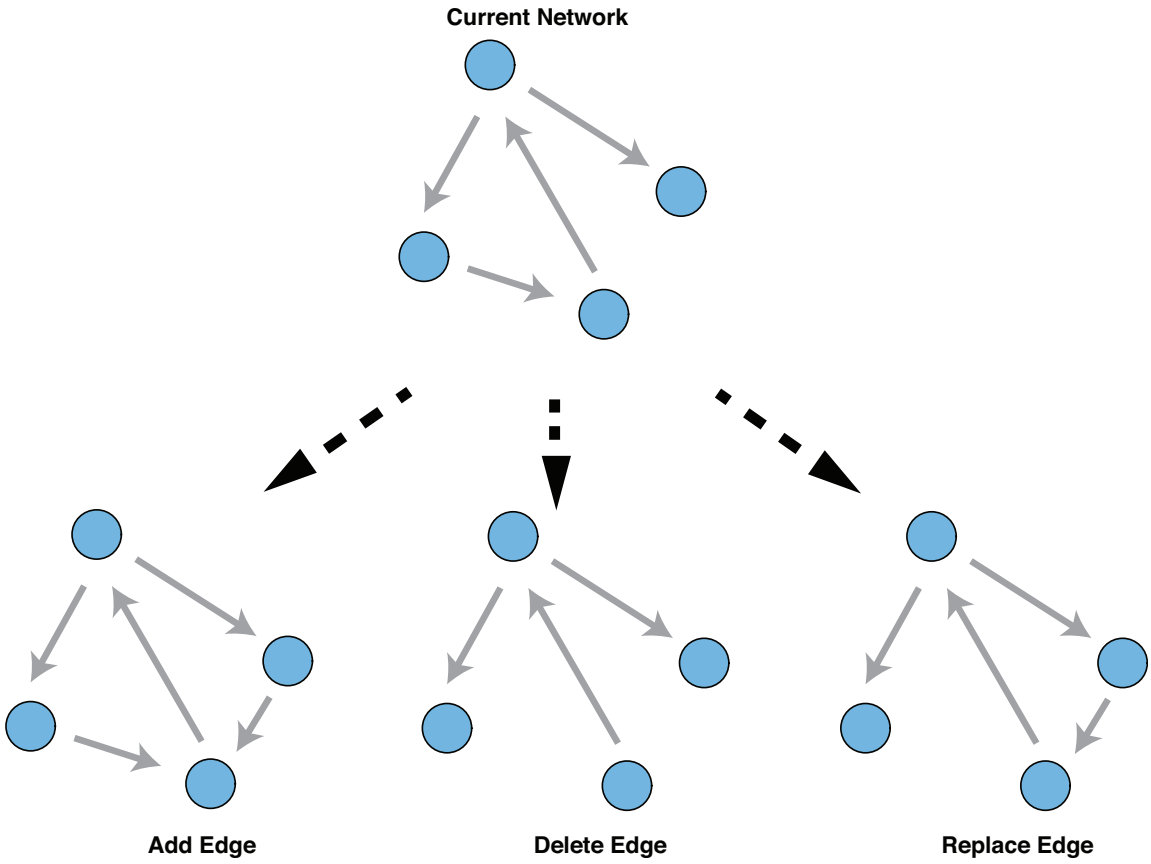

FIG. 2. The operations of changing the current network. We consider the three types of operations for an improvement of gene regulatory network (GRN), that is, ''Add Edge'' (adding), ''Delete Edge'' (deleting), and ''Replace Edge'' (replacing). Under the constraints of add_{max} and del_{max}, these operations are recursively executed until the network cannot be changed through these operations to decrease the Bayesian information criterion (BIC) score.

Algorithm 1. The proposed algorithm for improving GRNs based on the approximate posterior probability

Set r;
$add \leftarrow 0$; $del \leftarrow 0$; $flag \leftarrow 0$
$BIC_{current} \leftarrow$ the BIC score of the original model;
$\mathcal{M}_{current} \leftarrow \mathcal{M}_{original}$;
while $flag < N^2$ **do**
  for $i = 1$ to N **do**
    for $j = 1$ to N **do**
      if $A_{i,j}$ of $\mathcal{M}_{current} = 0$ **then**
        $changed \leftarrow$ Execute subalgorithm 1;
      else
        $changed \leftarrow$ Execute subalgorithm 2;
      end if
      if $changed$ **then**
        $flag \leftarrow 0$;
      else
        $flag \leftarrow flag + 1$;
      end if
      if $flag >= N^2$ **then**
        break;
      end if
    end for
    if $flag >= N^2$ **then**
      break;
    end if
  end for
end while

Algorithm 2. Subalgorithm 1

$changed \leftarrow FALSE$;
Consider $\mathcal{M}_{candidate}$ that is constructed from $\mathcal{M}_{candidate}$ by setting a regulation to the ith gene by the jth gene as an active element;
Estimate the parameter values and obtain the BIC score $BIC_{candidate}$ by the UKF and the EM-algorithm;
if $BIC_{current} > BIC_{candidate}$ and $add_{max} > add$ **then**
  Set $\mathcal{M}_{candidate}$ as $\mathcal{M}_{current}$; $BIC_{candidate} \leftarrow BIC_{current}$;
  $changed \leftarrow TRUE$;
else
  for $i = 1$ to N **do**
    for $k = 1$ to r **do**
      $j \leftarrow$ the kth minimum element with respect to $e(i, j_{col})$ $(j_{col} = 1, \ldots, N)$ of $\mathcal{M}_{candidate}$;
      if $A_{i,j}$ of $\mathcal{M}_{candidate}$ is 0 **then**
        continue;
      end if
      if $A_{i,j}$ of $\mathcal{M}_{original}$ is 1 or $add_{max} > add$ **then**
        continue;
      end if
      Consider $\mathcal{M}_{candidate2}$ that is constructed from $\mathcal{M}_{candidate}$ by setting a regulation to the ith gene by the jth gene as a nonactive set;
      Estimate the parameter values and obtain the BIC score by the UKF and the EM-algorithm;
    end for
  end for
  if $BIC_{current} >$ the minimum BIC score among models calculated above **then**
    Set $\mathcal{M}_{current}$ and $BIC_{current}$ as those of the minimum one;
    $changed \leftarrow TRUE$;
  end if
end if
Set add and del as those of the $\mathcal{M}_{current}$;
return $changed$;

Algorithm 3. Subalgorithm 2

changed ← *FALSE*;
Consider $\mathcal{M}_{candidate}$ that is constructed from $\mathcal{M}_{candidate}$ by setting a regulation to the *i*th gene by the *j*th gene as a non-active element;
Estimate the parameter values and obtain the BIC score $BIC_{candidate}$ by the UKF and the EM-algorithm;
if $BIC_{current} > BIC_{candidate}$ and $del_{max} > del$ **then**
  Set $\mathcal{M}_{candidate}$ as $\mathcal{M}_{current}$; $BIC_{candidate} \leftarrow BIC_{current}$;
  changed ← *TRUE*;
else
  for $i = 1$ to N **do**
    for $k = 1$ to r **do**
      $j \leftarrow$ the kth minimum element with respect to $e(i, j_{col})$ ($j_{col} = 1, \ldots, N$) of $\mathcal{M}_{candidate}$;
      if $A_{i,j}$ of $\mathcal{M}_{candidate}$ is 1 **then**
        continue;
      end if
      if $A_{i,j}$ of $\mathcal{M}_{original}$ is 0 or $add_{del} > del$ **then**
        continue;
      end if
      Consider $\mathcal{M}_{candidate2}$ that is constructed from $\mathcal{M}_{candidate}$ by setting a regulation to the *i*th gene by the *j*th gene as an active set;
      Estimate the parameter values and obtain the BIC score by the UKF and the EM-algorithm;
    end for
  end for
  if $BIC_{current} >$ the minimum BIC score among models calculated above **then**
    Set $\mathcal{M}_{current}$ and $BIC_{current}$ as those of the minimum one;
    changed ← *TRUE*;
  end if
end if
Set *add* and *del* as those of the $\mathcal{M}_{current}$;
return *changed*;

To measure the effectiveness of the candidate models when changing active sets, Equation (37), of which active sets are changed as those of the next candidate, is calculated. Then, only for r top models with respect to $-e(\mathcal{M}, i)$ for each i, the BIC scores are evaluated by estimating the parameter values maximizing prediction ability for observational data using UKF and the EM-algorithm. This procedure is shown in Figure 3. Note that $e(\mathcal{M}, i)$ can be derived when calculating $\arg\arg\max_{\boldsymbol{a}_i} q(\boldsymbol{\theta}|\boldsymbol{\theta}_l)$.

3. RESULTS

3.1. Comparison analysis using synthetic data of WNT5A and yeast network

To show the effectiveness of the proposed algorithm, we used artificial time-course gene expression data from two synthetic networks of WNT5A (Kim et al., 2002) and a yeast cell cycle network (Kanehisa et al., 2012) as illustrated in Figures 4 and 5, respectively. For each network, we generated two time-courses consisting of $\mathcal{T} = \{1, 2, \ldots, 30\}$ and $\{1, 2, \ldots, 10, 12, \ldots, 30\}$ by using Equations (3) and (4). For Equation (3), the values of the parameters were determined between 0 and 1, and the system noise was according to Gaussian distribution with a mean of 0 and a variance of 0.1. In Equation (4), Gaussian observational noise with a mean of 0 and a variance of 0.3 were added to these artificial data. Note that the networks were used for the performance comparison in the previous study (Nakajima et al., 2012).

For this comparison, we applied (a) the proposed method, (b) a regression-based method (DPLSQ) (Nakajima et al., 2012), (c) DPLSQ with BIC (Schwarz, 1978), and (d) Akaike information criterion (AIC) (Akaike, 1974) to the data sets. Here, since DPLSQ is based only on the least-square errors, it may infer many false positives. Then, we modified the algorithm to use BIC and AIC; r in the proposed algorithm is set 3.

For each data set, 10 trials were executed, for each of which the true network of Figures 4 and 5 is randomly modified and given as an original network. Thus, a network obtained by adding and deleting five edges from the true network was given as an original network and then (a)–(d) were applied to obtain the

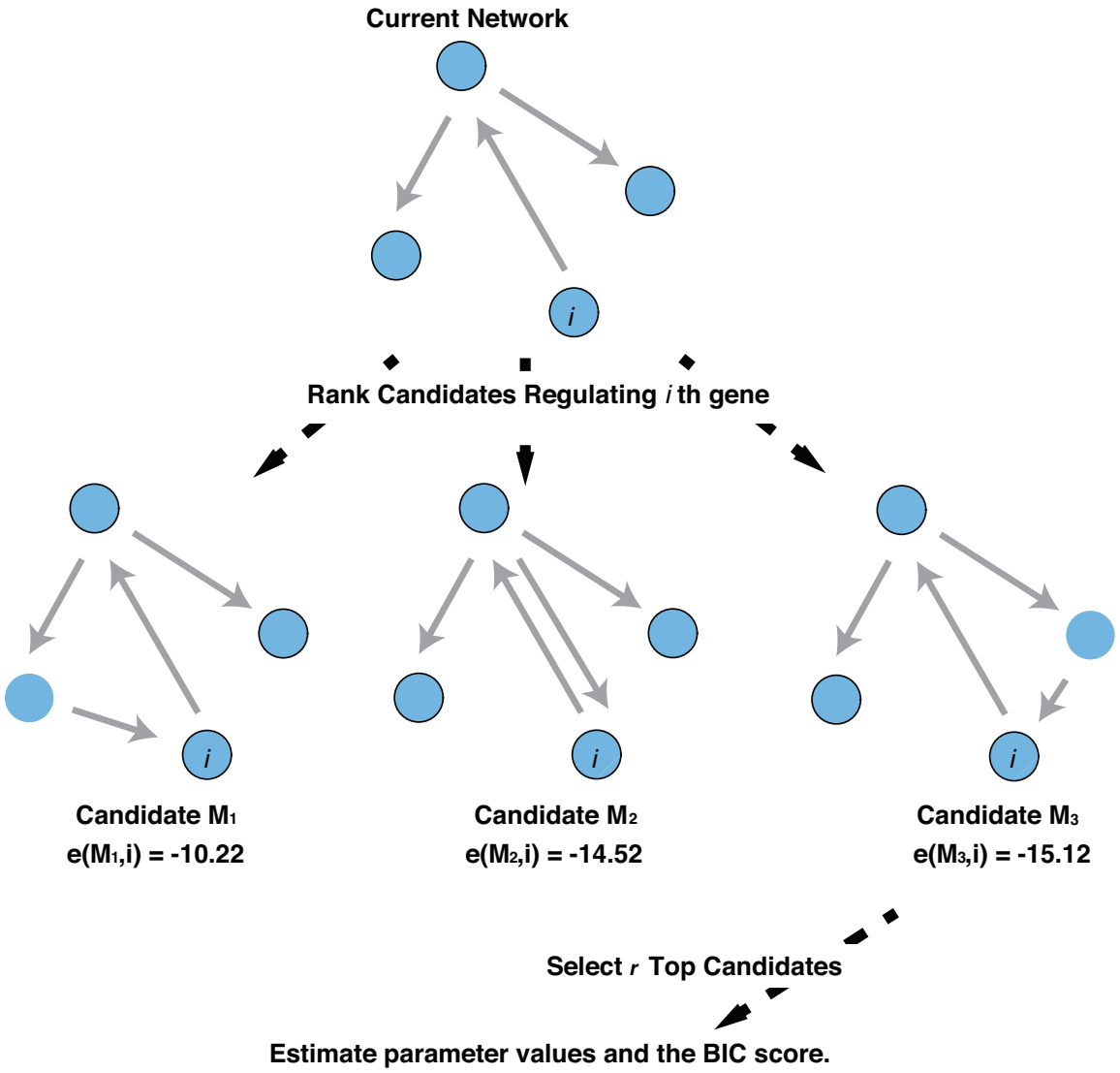

FIG. 3. A cartoon figure of the proposed algorithm. For the current network, the proposed algorithm constructs candidate networks by adding, deleting, and replacing edges, and ranks them using $e(\mathcal{M}, i)$. Then, only r top networks with respect to $-e(\mathcal{M}, i)$ are evaluated with the BIC score by estimating the parameter values.

true network. We evaluated the average performance of true positive (TP), false positive (FP), true negative (TN), false negative (FN), precision rate ($PR = \frac{TP}{TP+FP}$), recall rate ($RR = \frac{TP}{TP+FN}$), and F-measure ($= \frac{2PR \cdot RR}{PR+RR}$) over 10 trials for each set of data. In contrast to the usual way, we counted TP when an altered edge was successfully improved as the true model, FN when an altered edge was not improved, and FP when an edge in the true model was changed in the improved model. The results of using the four time-courses are summarized in Tables 1–4 (the proposed method is noted as ''UKF-Completion''), respectively. These results clearly show that the proposed algorithm has the highest performance as compared to the other methods for all data sets. In particular, for non-equally spaced time-point data, the proposed method could better infer true regulations than the previous methods since our approach utilizes the hidden state and can handle nonobservational time points.

3.2. Real data analysis using the yeast cell cycle network

As an application example of improving and extending literature-based networks, we dealt with a yeast cell-cycle network from KEGG (Kanehisa et al., 2012) and used the corresponding observational data (Spellman et al., 1998). By using time-course data including 25 genes of which regulatory relationships are

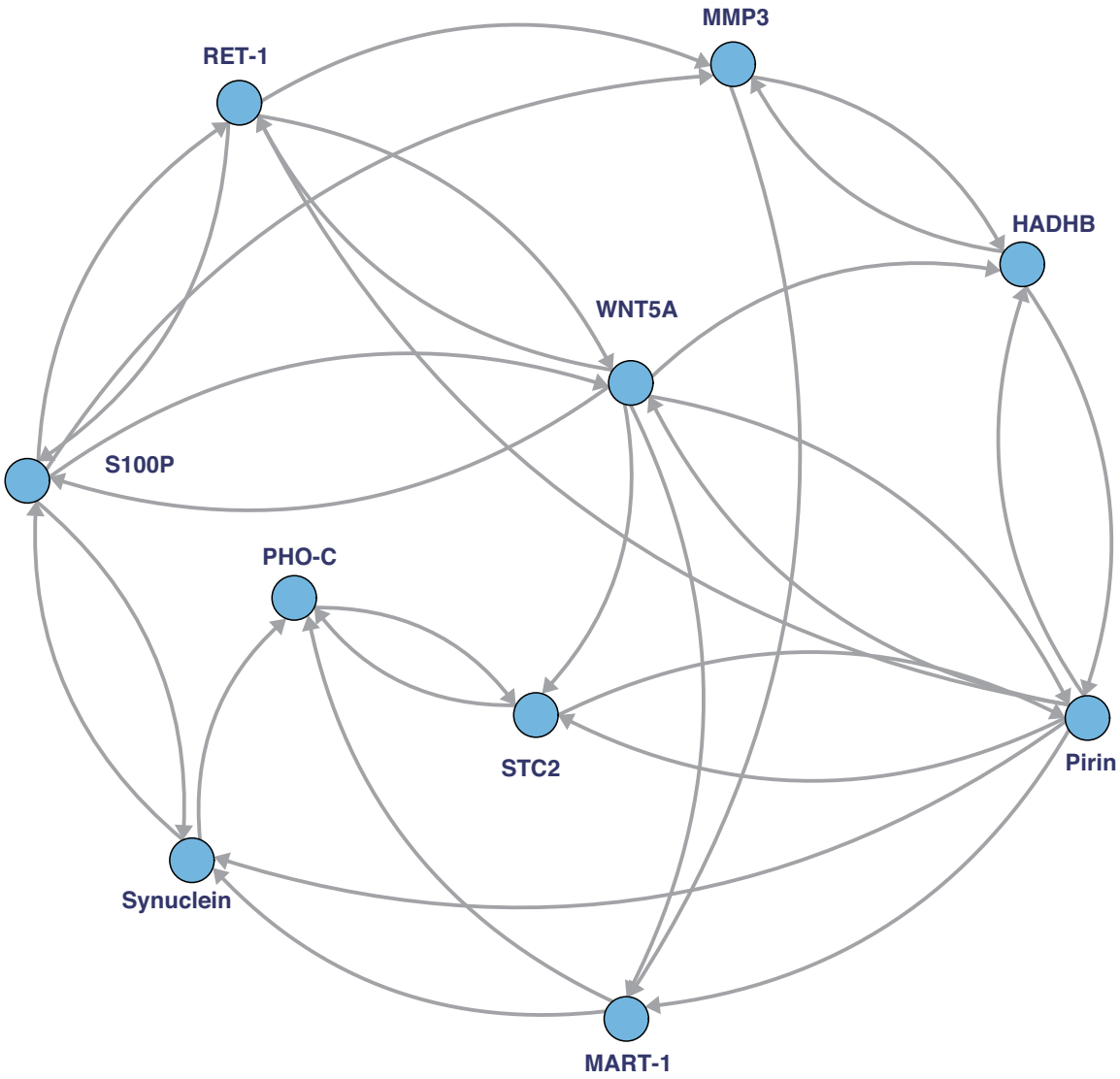

FIG. 4. A real biological network, termed WNT5A network (Kim et al., 2002), used for the comparison analysis. Based on the network, the original networks are generated by randomly adding and deleting five edges.

represented as red arrows in Figure 6, and considering this as an original network, we attempted to improve the network. However, since the network is classical and highly reliable in KEGG database, we focused on the extension of the network using additional genes. Thus, we considered the network consisting of these 25 genes and 38 additional candidate genes, which can have functions related to a yeast cell cycle pathway, from the Saccharomyces Genome Database (SGD) (Cherry et al., 2012). We did not set prior regulatory structure to these 38 genes and extended the KEGG-based network consisting of 25 genes by adding regulations to these 38 genes ($del_{max} = 0$).

Consequently, 38 candidate genes were integrated in the KEGG-based yeast cell cycle network as illustrated in Figure 6. In this figure, the KEGG-based regulatory network consisting of 25 genes was drawn as rectangles (gene) and red arrows (regulation), and newly estimated relationships were drawn as circles (gene) and black chained arrows (regulation). Interestingly, there exist many combinatorial regulations of which regulated genes have more than two regulations. Since these regulations can have nonzero values of the combinatorial effect $b_{i,(j,k)}$, the results may not be obtained by linear models. Furthermore, some genes, such as *YOX*1 and *Cdc*6, become hub genes regulating many other genes, and they are known as upper stream genes regulating downstream genes on the KEGG database. These results show the possibility of the causal relationships between them.

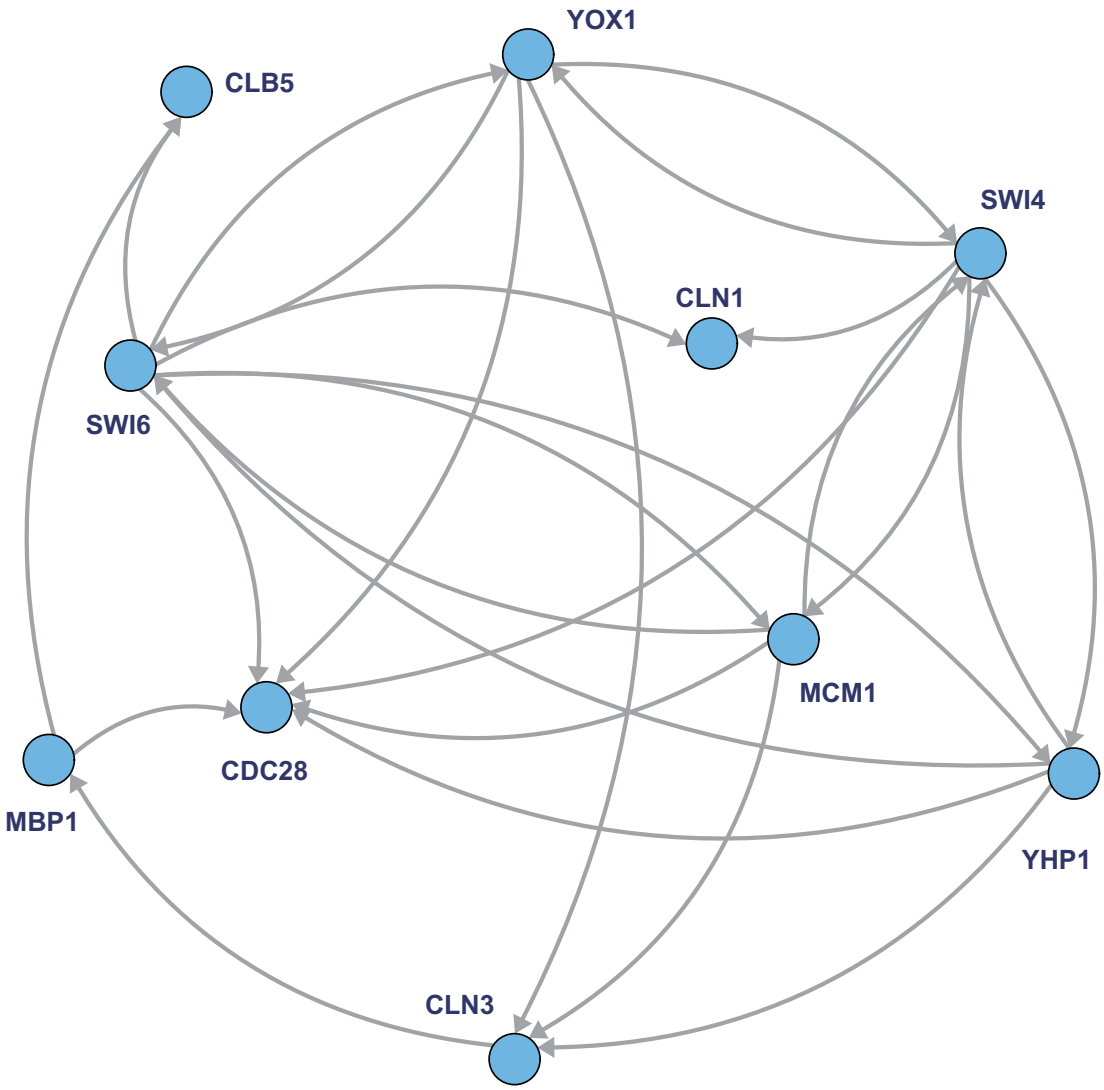

FIG. 5. A real biological network of yeast cell cycle from the KEGG database (Nakajima et al., 2012; Kanehisa et al., 2012) used for comparison analysis. Based on the network, the original networks are generated by randomly adding and deleting five edges.

4. CONCLUSION

We proposed a genomic data assimilation schema using a nonlinear simulation model for improving and extending literature-based networks. The method can efficiently estimate parameter values of a simulation model by using the EM-algorithm with UKF. Furthermore, the proposed algorithm avoids evaluating all possible

Table 1. Comparison of the Proposed Method and DPLSQ Using Equally Spaced Artificial Data from WNT5A Network

	PR	*RR*	*F-measure*	*TP*	*FP*	*TN*	*FN*
DPLSQ	0.580	0.290	0.386	2.9	2.1	87.9	7.1
DPLSQ (BIC)	0.677	0.670	0.673	6.7	3.2	86.8	3.3
DPLSQ (AIC)	0.700	0.650	0.673	6.5	2.8	87.2	3.5
UKF-Completion	0.760	0.760	0.760	7.6	2.4	87.6	2.4

PR, precision rate; RR, recall rate; TP, true positive; FP, false positive; TN, true negative; FN, false negative; BIC, Bayesian information criterion; AIC, Akaike information criterion; UKF, unscented Kalman filter.

TABLE 2. COMPARISON OF THE PROPOSED METHOD AND DPLSQ USING NON-EQUALLY SPACED ARTIFICIAL DATA FROM WNT5A NETWORK

	PR	*RR*	*F-measure*	*TP*	*FP*	*TN*	*FN*
DPLSQ	0.540	0.270	0.360	2.7	2.3	87.7	7.3
DPLSQ (BIC)	0.520	0.520	0.520	5.2	4.8	85.2	4.8
DPLSQ (AIC)	0.523	0.49	0.506	4.9	4.5	85.5	5.1
UKF-Completion	0.720	0.720	0.720	7.2	2.8	87.2	2.8

TABLE 3. COMPARISON OF THE PROPOSED METHOD AND DPLSQ USING EQUALLY SPACED ARTIFICIAL DATA FROM A YEAST CELL CYCLE NETWORK

	PR	*RR*	*F-measure*	*TP*	*FP*	*TN*	*FN*
DPLSQ	0.600	0.300	0.400	3.0	2.0	88.0	7.0
DPLSQ (BIC)	0.597	0.590	0.593	5.9	4.0	86.0	4.1
DPLSQ (AIC)	0.600	0.600	0.600	6.0	4.0	86.0	4.0
UKF-completion	0.650	0.650	0.650	6.5	3.5	86.5	3.5

TABLE 4. COMPARISON OF THE PROPOSED METHOD AND DPLSQ USING NON-EQUALLY SPACED ARTIFICIAL DATA FROM A YEAST CELL CYCLE NETWORK

	PR	*RR*	*F-measure*	*TP*	*FP*	*TN*	*FN*
DPLSQ	0.475	0.238	0.317	2.4	2.6	87.4	7.6
DPLSQ (BIC)	0.413	0.413	0.413	4.1	5.9	84.1	5.9
DPLSQ (AIC)	0.413	0.413	0.413	4.1	5.9	84.1	5.9
UKF-Completion	0.588	0.588	0.588	5.9	4.1	85.9	4.1

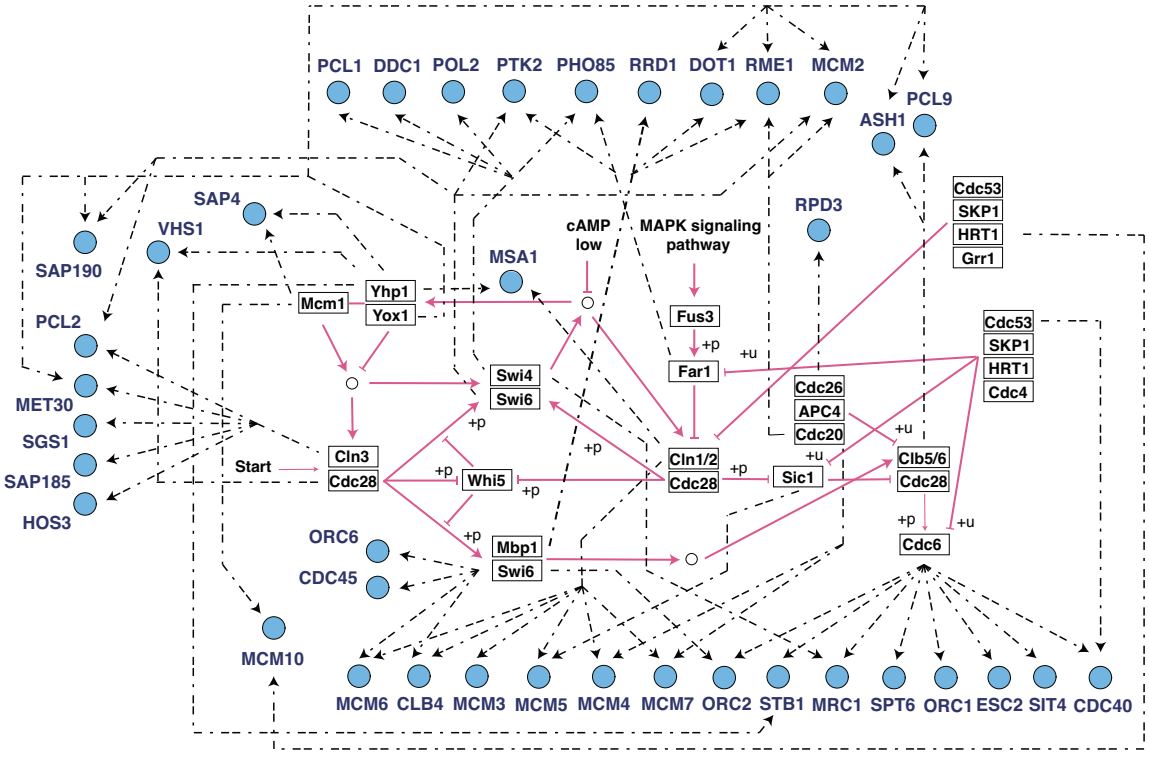

FIG. 6. A part of a yeast cell cycle network, and candidate genes for extending the network. The KEGG-based regulatory network consisting of 25 genes is drawn as rectangles (gene) and red arrows (regulation), and newly estimated relationships are drawn as circles (gene) and black chained arrows (regulation).

candidates that are constructed by modifying the original network and selects only plausible ones through measuring the effectiveness when modifying the regulation of the current network for the data. Therefore, this schema makes it possible to deal with many candidate networks and finds better networks for the data.

The performance of this approach was demonstrated by implementing artificial simulation data from real biological networks termed WNT5A and a yeast cell cycle network. Consequently, our proposed method can evaluate GRNs more accurately than could a previously developed method (DPLSQ). In particular, since our method is based on the state space representation using the hidden state for representing gene regulatory dynamics, the flexibility for the observational data, that is, which can handle observational data with non-equally spaced time points, can be ensured. These results indicated the high performance and adaptability of the proposed method to improve and extend the original network using time-course observational data. As an application example, using a part of a well-investigated yeast cell-cycle network from KEGG, we applied the proposed method to extend the network by integrating additional candidate genes from SGD (Cherry et al., 2012). Interestingly, we found hub genes regulating candidate genes that are indicated as upstream genes in KEGG database. Since these are biologically related candidates of the original networks, these extensions might be true regulations and thus should be confirmed by biological experiments.

ACKNOWLEDGMENTS

The super-computing resource was provided by the Human Genome Center, the Institute of Medical Science, the University of Tokyo.

This work was partly supported by Grant-in-Aid for JSPS Fellows Number 24-9639.

AUTHOR DISCLOSURE STATEMENT

No competing financial interests exist.

REFERENCES

Akaike, H. 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control* 19, 716–723.

Akutsu, T., Tamura, T., and Horimoto, K. 2009. Completing networks using observed data, 126–140. *In* Gavalda, R., Lugosi, G., Zeugmann, T., et al., eds. *Algorithmic Learning Theory*, Volume 5809, *Lecture Notes in Computer Science*. Springer, Berlin Heidelberg.

Asif, H.M.S., and Sanguinetti, G. 2011. Large-scale learning of combinatorial transcriptional dynamics from gene expression. *Bioinformatics* 27, 1277–1283.

Beal, M.J., Falciani, F., Ghahramani, Z., Rangel, C., and Wild, D.L. 2005. A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. *Bioinformatics* 21, 349–356.

Cherry, J.M., Hong, E.L., Amundsen, C., et al. 2012. Saccharomyces genome database: the genomics resource of budding yeast. *Nucleic Acids Res.* 40, 700–705.

Chow, S.-M., Ferrer, E., and Nesselroade, J.R. 2007. An unscented kalman filter approach to the estimation of nonlinear dynamical systems models. *Multivariate Behavioral Research* 42, 283–321.

Dempster, A.P., Laird, N.M., and Rubin, D.B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B* 39, 1–38.

Friedman, J., Hastie, T., and Tibshirani, R. 2007. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9, 432–441.

Hasegawa, T., Nagasaki, M., Yamaguchi, R., et al. 2014. An efficient method of exploring simulation models by assimilating literature and biological observational data. *Biosystems* 121, 54–66.

Hasegawa, T., Yamaguchi, R., Nagasaki, M., et al. 2011. Comprehensive pharmacogenomic pathway screening by data assimilation. *Proceedings of the 7th International Conference on Bioinformatics Research and Applications*, ISBRA 11, 160–171. Springer-Verlag, Berlin, Heidelberg.

Hirose, O., Yoshida, R., Imoto, S., 2008. Statistical inference of transcriptional module-based gene networks from time course gene expression profiles by using state space models. *Bioinformatics* 24, 932–942.

Julier, S. 2002. The scaled unscented transformation. *American Control Conference, 2002*, 6, 4555–4559.

Julier, S., and Uhlmann, J. 2004. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE* 92, 401–422.

Julier, S., Uhlmann, J., and Durrant-Whyte, H. 2000. A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Transactions on Automatic Control* 45, 477–482.

Julier, S.J., and Uhlmann, J.K. 1997. A new extension of the kalman filter to nonlinear systems. *Proc. of AeroSense: The 11th Int. Symp. on Aerospace/Defense Sensing, Simulations and Controls*, 182–193.

Kalman, R.E. 1960. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering* 82 (Series D), 35–45.

Kanehisa, M., Goto, S., Sato, Y., et al. 2012. KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res.* 40, D109–D114.

Kim, S., Li, H., Dougherty, E.R., et al. 2002. Can Markov chain models mimic biological regulation? *Journal of Biological Systems* 10, 337–357.

Kitagawa, G. 1998. A self-organizing state-space model. *J. Am. Stat. Assoc.* 93, 1203–1215.

Koh, C.H.H., Nagasaki, M., Saito, A., et al. 2010. DA 1.0: parameter estimation of biological pathways using data assimilation approach. *Bioinformatics* 26, 1794–1796.

Kojima, K., Yamaguchi, R., Imoto, S., et al. 2010. A state space representation of var models with sparse learning for dynamic gene networks. *International Conference on Genome Informatics* 22, 56–68.

Lillacci, G., and Khammash, M. 2010. Parameter estimation and model selection in computational biology. *PLoS Comput. Biol*. 6, e1000696.

Liu, X., and Niranjan, M. 2012. State and parameter estimation of the heat shock response system using kalman and particle filters. *Bioinformatics* 28, 1501–1507.

Mahdi, R., Madduri, A.S., Wang, G., et al. 2012. Empirical Bayes conditional independence graphs for regulatory network recovery. *Bioinformatics*, 28, 2029–2036.

Murtuza Baker, S., Poskar, C.H., Schreiber, F., and Junker, B.H. 2013. An improved constraint filtering technique for inferring hidden states and parameters of a biological model. *Bioinformatics* 29, 1052–1059.

Nagasaki, M., Yamaguchi, R., Yoshida, R., 2006. Genomic data assimilation for estimating hybrid functional petri net from time-course gene expression data. *Genome Informatics* 17, 46–61.

Nakajima, N., Tamura, T., Yamanishi, Y., Horimoto, K., and Akutsu, T. 2012. Network completion using dynamic programming and least-squares fitting. *Scientific World Journal* 2012.

Nakamura, K., Yoshida, R., Nagasaki, M., et al. 2009. Parameter estimation of *in silico* biological pathways with particle filtering toward a petascale computing. *Pacific Symposium on Biocomputing 2009* 14, 227–238.

Opper, M., and Sanguinetti, G. 2010. Learning combinatorial transcriptional dynamics from gene expression data. *Bioinformatics* 26, 1623–1629.

Quach, M., Brunel, N., and d'Alche Buc, F. 2007. Estimating parameters and hidden variables in non-linear state-space models based on odes for biological networks inference. *Bioinformatics* 23, 3209–3216.

Rangel, C., Angus, J., Ghahramani, Z., et al. 2004. Modeling t-cell activation using gene expression profiling and state-space models. *Bioinformatics* 20, 1361–1372.

Rogers, S., Khanin, R., and Girolami, M. 2007. Bayesian model-based inference of transcription factor activity. *BMC Bioinformatics* 8 (Suppl).

Sarkka, S. 2008. Unscented rauch–tung–striebel smoother. *IEEE Transactions on Automatic Control* 53, 845–849.

Savageau, M.A. 1969. Biochemical systems analysis: II. The steady-state solutions for an n-pool system using a power-law approximation. *J. Theor. Biol.* 25, 370–379.

Savageau, M.A., and Voit, E.O. 1987. Recasting nonlinear differential equations as s-systems: a canonical nonlinear form. *Math. Biosci.* 87, 83–115.

Schwarz, G. 1978. Estimating the dimension of a model. *The Annals of Statistics* 6, 461–464.

Spellman, P.T., Sherlock, G., Zhang, M.Q., 1998. Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Mol. Biol. Cell*. 9, 3273–3297.

Wang, W., Cherry, J.M., Nochomovitz, Y., 2005. Inference of combinatorial regulation in yeast transcriptional networks: A case study of sporulation. *Proceedings of the National Academy of Sciences of the United States of America* 102, 1998–2003.

Watanabe, Y., Seno, S., Takenaka, Y., and Matsuda, H. 2012. An estimation method for inference of gene regulatory network using Bayesian network with uniting of partial problems. *BMC Genomics*. 13, S12.

Address correspondence to:
Takanori Hasegawa
Bioinformatics Center
Institute for Chemical Research
Kyoto University
Gokasho, Uji,
Kyoto, 611-0011
Japan

E-mail: t-hasegw@kuicr.kyoto-u.ac.jp