BIOINFORMATICS

Vol. 29 ISMB/ECCB 2013, pages i98–i107
doi:10.1093/bioinformatics/btt232

A framework for scalable parameter estimation of gene circuit models using structural information

Hiroyuki Kuwahara[1,†], Ming Fan[1,†], Suojin Wang[2] and Xin Gao[1,*]

[1]Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia and [2]Department of Statistics, Texas A&M University, College Station, TX 77843, USA

ABSTRACT

Motivation: Systematic and scalable parameter estimation is a key to construct complex gene regulatory models and to ultimately facilitate an integrative systems biology approach to quantitatively understand the molecular mechanisms underpinning gene regulation.

Results: Here, we report a novel framework for efficient and scalable parameter estimation that focuses specifically on modeling of gene circuits. Exploiting the structure commonly found in gene circuit models, this framework decomposes a system of coupled rate equations into individual ones and efficiently integrates them separately to reconstruct the mean time evolution of the gene products. The accuracy of the parameter estimates is refined by iteratively increasing the accuracy of numerical integration using the model structure. As a case study, we applied our framework to four gene circuit models with complex dynamics based on three synthetic datasets and one time series microarray data set. We compared our framework to three state-of-the-art parameter estimation methods and found that our approach consistently generated higher quality parameter solutions efficiently. Although many general-purpose parameter estimation methods have been applied for modeling of gene circuits, our results suggest that the use of more tailored approaches to use domain-specific information may be a key to reverse engineering of complex biological systems.

Availability: http://sfb.kaust.edu.sa/Pages/Software.aspx

Contact: xin.gao@kaust.edu.sa

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

A quantitative understanding of how expression of genes is controlled in time and space through the integration of computational and experimental methods is a main goal of molecular systems biology (Church, 2005; Ideker *et al.*, 2001; Kitano, 2002). Among the major obstacles in such an integrative systems biology approach is the construction of kinetic models that quantitatively support the current knowledge of a given gene circuit. What makes the construction of gene circuit models especially difficult is the quantification of all reaction parameters, as direct measurements of gene regulation kinetics are seldom available. Thus, model parameters are often estimated indirectly using more readily available experimental data [e.g. Schoeberl *et al.* (2002); Zwolak *et al.* (2005)]. Even in modeling of relatively well-known gene circuits, such as the phage-λ lysis–lysogeny developmental pathway (Arkin *et al.*, 1998), there are a number of unknown parameters, which are phenomenologically determined by fitting the model's outputs to some experimental observations.

The quality of time series gene expression data is crucial to the construction of phenomenological models that accurately capture the observed dynamical characteristics of a given gene circuit. With advances in the gene expression detection technologies, single-molecule level measurements of gene expression can now be obtained in a wide range of organisms (Baugh *et al.*, 2011; Cai *et al.*, 2006; Golding *et al.*, 2005; Materna *et al.*, 2010; Newman *et al.*, 2006; Suter *et al.*, 2011; Taniguchi *et al.*, 2010; Zenklusen *et al.*, 2008). In particular, recent advances in fluorescence imaging techniques (Joo *et al.*, 2008; Raj and van Oudenaarden, 2009) facilitate real-time measurements of gene expression at the single-molecule level, making more accurate parameter estimation for quantitative modeling of gene circuits possible. Such single-cell gene expression data are, however, noisy because of intrinsic and extrinsic fluctuations (Cai *et al.*, 2006; Elowitz *et al.*, 2002; Newman *et al.*, 2006; Golding *et al.*, 2005; Raj *et al.*, 2006; Raser and O'Shea, 2005; Suter *et al.*, 2011; Taniguchi *et al.*, 2010) and often limited to lower concentration molecular species such as mRNAs (Kulkarni, 2011; van Oijen, 2011). Because of such noisy gene expression and highly nonlinear dynamics involved in transcriptional regulations, manual parameter estimation in nontrivial gene circuit models is generally infeasible.

To systematically estimate the parameters of a biochemical kinetic model, the parameter estimation problem is often treated as an optimization problem in which parameter values are selected to minimize a certain objective function (Schwartz, 2008). Although several stochastic optimization and Bayesian-based methods were successfully applied to estimate parameters of biochemical models (Baker *et al.*, 2010; Moles *et al.*, 2003), they often suffer from scalability problems when there are a large number of unknown parameters. To make the estimation of parameters more efficient, several methods have been proposed to reduce the parameter search space by decomposing rate equations (Jia *et al.*, 2011; Koh *et al.*, 2006; Zhan and Yeung, 2011). However, the quality of these methods strongly depends on interpolation and smoothing functions, which are often independent of the underlying model structure and can add strong artifacts. Recently, Kalman filter-based approaches, which can alleviate the scalability problem, were applied to efficiently estimate kinetic parameters (Lillacci and Khammash, 2010; Quach *et al.*, 2007; Sun *et al.*, 2008). Although these approaches support parameter estimation of models with unobserved variables, a

[*]To whom all correspondence should be addressed.
[†]The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

recent comparative study showed that their performance could be sensitive to the initial condition, and estimated parameters might be far from the true ones if the initial guess was not close to the solution (Liu and Niranjan, 2012). Most of these existing methods are applicable to parameter estimation problems of generic dynamical models, as they do not demand any domain-specific knowledge. Although these general-purpose methods can easily be applied to modeling of any biological systems, it is clear that each of these methods has its advantages and disadvantages, and that no single method is versatile enough to efficiently give optimal parameter sets for all biological models. This observation has led us to develop a more tailored parameter estimation method that focuses on a specific yet important subclass of biological models, namely, gene circuit models.

To facilitate the mechanistic construction of thermodynamics-based models (Shea and Ackers, 1985; Sherman and Cohen, 2012) that describe the quantitative behavior of gene regulation from time series mRNA data, we developed a novel parameter estimation framework called *Parameter Estimation by Decomposition and Integration* (PEDI) that specifically focuses on modeling of gene circuits. The main paradigm of PEDI is 'divide' and 'conquer'; by using the given mRNA data and exploiting the structure of gene circuit models, our framework divides a high-dimensional parameter estimation problem into subproblems with a much smaller parameter space, each of which is, in turn, conquered (i.e. solved) by using any constrained optimization method. At the initial step, this problem reduction process leads to a crude linearization for numerical integrations, which often results in poor estimates especially for highly nonlinear systems. To improve the quality of the estimate with a basically negligible increase in computing time, PEDI places intermediate integration points using the underlying structural information of a given gene circuit model and iteratively increases the accuracy of these intermediate points to increase the accuracy of the numerical integration, which in turn improves the reconstructed dynamics. This article introduces PEDI and, through the use of simulated annealing (SA) as the optimization method, applies the framework to three-gene circuit models with complex dynamics based on synthetic time series mRNA datasets and one yeast gene circuit model based on time series microarray data. We compared PEDI with three state-of-the-art parameter estimation methods, namely, the *evolutionary strategy with stochastic ranking* (SRES) (Runarsson and Yao, 2000), the *moment matching method coupled with hybrid extended Kalman filter* (HEKF + MM) (Lillacci and Khammash, 2010) and the *two-phase dynamic decoupling method* (TDDM) (Jia *et al*., 2011). Our results show that PEDI consistently produced the most accurate estimates efficiently in all the four parameter estimation experiments. This study, thus, demonstrated that PEDI could provide an effective approach to efficiently estimating kinetic parameters of gene circuit models.

2 METHODS

2.1 Problem setting

We concern ourselves with time series gene expression data generated from an N-gene network at equally spaced $M+1$ time points, $t_0 < t_1 \cdots < t_M$. Gene g_i is transcribed into mRNA m_i, which is then translated into protein p_i, which can then be used to regulate the transcription of genes in the network. We denote by m_{ij} and p_{ij} random variables representing the levels of the mRNA copy and the protein copy of gene g_i at time t_j, respectively. We further assume that these random variables be expressed as follows:

$$\begin{aligned} m_{ij} &= \mu_{m_{ij}} + v_{ij}, \\ p_{ij} &= \mu_{p_{ij}} + u_{ij}, \quad \text{for } i = 1, \ldots, N \text{ and } j = 0, \ldots, M, \end{aligned}$$

where $\mu_{m_{ij}}$ and $\mu_{p_{ij}}$ are the true mean of m_{ij} and p_{ij}, respectively, whereas each of v_{ij} and u_{ij} is a statistically independent random variable with mean 0. We consider that only the levels of mRNAs are observable from the experiments, but we assume that the true mean of each protein p_i be known at time t_0.

Here, we are interested in constructing a kinetic model that estimates the average trajectory of mRNAs given by $\mu_{m_{ij}}$, and we do not focus on the time evolution of higher moments, as experimental time-series data often contain only few datasets. Our model describes the average time evolution of a gene circuit as a continuous-time deterministic process, which is governed by a system of ordinary differential equations (ODEs) as follows:

$$\begin{aligned} \frac{d\hat{m}_i}{dt} &= h_i(\hat{p}; \theta_i) - \theta_{i1}\hat{m}_i, \\ \frac{d\hat{p}_i}{dt} &= \alpha_i \hat{m}_i - \beta_i \hat{p}_i, \end{aligned} \tag{1}$$

with the initial conditions:

$$\begin{aligned} \hat{m}_i(t_0) &= \overline{m}_{i0}, \\ \hat{p}_i(t_0) &= \mu_{p_{i0}}, \quad \text{for } i = 1, \ldots, N. \end{aligned}$$

Here, $\hat{m}_i$ and $\hat{p}_i$ are time-dependent variables that estimate the dynamics of $\mu_{m_{ij}}$ and $\mu_{p_{ij}}$, respectively; $\theta_i \equiv (\theta_{i1}, \ldots, \theta_{iK_i})$ is a K_i dimensional vector that represents the parameters used in the rate equation representing the regulation of mRNA m_i; h_i is the transcription rate function based on the equilibrium thermodynamics model of the *cis* regulation of gene g_i; $\hat{p}$ is an N dimensional vector whose i-th element is $\hat{p}_i$; α_i and β_i are the parameters used in the regulation of protein p_i, which we assume to be known; and $\overline{m}_{ij}$ is the sample mean of m_i at time t_j. The regulation of each gene is modeled using four reaction processes, transcription, mRNA degradation, translation and protein degradation, whereby transcription is considered to be the main regulatory step. Using this model, our objective is to search for values of the unknown parameters, which minimize a weighted sum of squared residuals of the sample means of mRNAs. As a system of coupled nonlinear rate equations can seldom be solved analytically, the numerical integration via simulation is usually used to estimate the levels of mRNAs for a given parameter value. When the dimension of the unknown parameters is high, however, finding practical solution for $\theta \equiv (\theta_1, \ldots, \theta_N)$ based on simulation for each parameter change becomes computationally intensive, and such an approach eventually proves to be infeasible.

2.2 Overview of PEDI

Figure 1 illustrates a high-level workflow of PEDI. The main idea of our framework is to optimize parameters separately by using what is available rather than dealing with exponentially larger parameter space involved in the optimization of θ. The framework takes advantage of the fact that the rate functions in our gene circuit models have the following structure:

$$\frac{dy}{dt} = f(t) - ky, \tag{2}$$

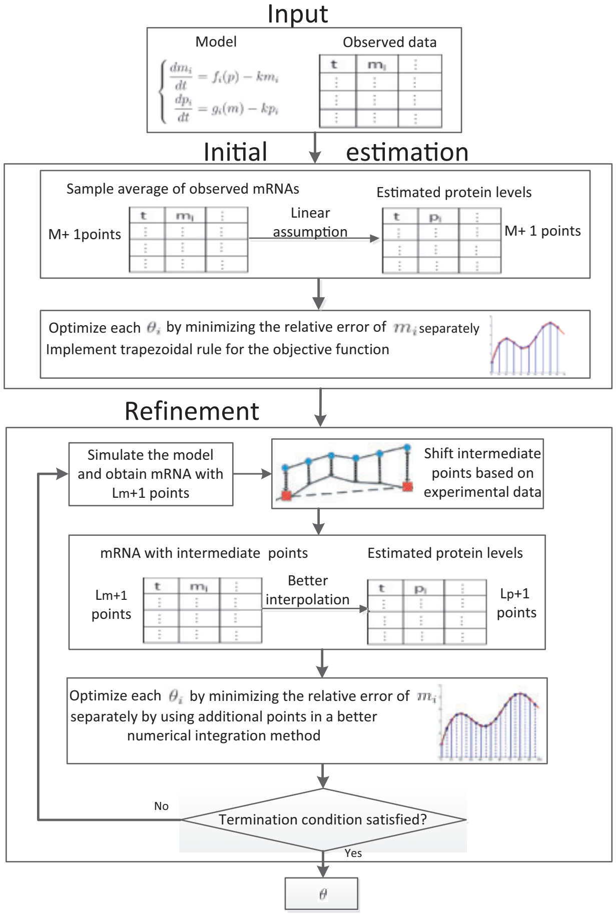
Input
Model
$\begin{cases} \frac{dm_i}{dt} = f_i(p) - km_i \\ \frac{dp_i}{dt} = g_i(m) - kp_i \end{cases}$
Observed data
t
m_i
Initial estimation
Sample average of observed mRNAs
Estimated protein levels
M+ 1points
Linear assumption
M+ 1 points
p_i
Optimize each θ_i by minimizing the relative error of m_i separately
Implement trapezoidal rule for the objective function
Refinement
Simulate the model and obtain mRNA with Lm+1 points
Shift intermediate points based on experimental data
mRNA with intermediate points
Estimated protein levels
Lm+1 points
Better interpolation
Lp+1 points
Optimize each θ_i by minimizing the relative error of m_i separately by using additional points in a better numerical integration method
No
Termination condition satisfied?
Yes
θ

Fig. 1. An illustration of the workflow of PEDI. Briefly, given a model structure and time series mRNA data at $M+1$ time points, it first makes a linear assumption and estimates the proteins data points, which are then used to estimate parameters for the mRNA regulations. These initial estimates are then iteratively refined by placing L_m+1 mRNA integration points and L_p+1 protein integration points and by increasing the accuracy of these integration points

whose definite time integral from time t_0 to time t_j has the following form:

$$\begin{aligned} y(t_j) &= e^{-k(t_j-t_0)}\left[y(t_0) + \int_{t_0}^{t_j} f(t)e^{k(t-t_0)}dt\right] \\ &= e^{-k(t_j-t_{j-1})}\left[y(t_{j-1}) + \int_{t_{j-1}}^{t_j} f(t)e^{k(t-t_{j-1})}dt\right], \end{aligned} \quad (3)$$

where $t_{j-1} < t_j$. As the transcription rate functions depend on regulatory proteins, our parameter estimation framework based on the decomposition of a gene circuit model requires the estimate of the protein levels first. To this end, PEDI uses the time series sample average of the mRNA and makes a linear assumption to estimate the mean time evolution of each protein level. However, as gene circuits often involve highly nonlinear reactions, such a crude linear interpolation may result in an inadequate parameter estimation. To refine the quality of the parameter estimation, the framework enriches the number of the data points by estimating intermediate points of the observed data points using the output from a computational simulation. These intermediate data points are then used to make the interpolation of the observed data points and the numerical integration of the rate functions more accurate. The introduction of these intermediate data points does not increase the complexity of the parameter search space, as they are only used for numerical integrations. By repeating this process, PEDI attempts to increase the accuracy of the interpolation and the fitness of the parameter estimation. Thus, PEDI can efficiently perform parameter estimation by avoiding computationally intensive search in a high-dimensional parameter space while keeping the quality of the parameter estimation high.

As PEDI decomposes a system of ODEs into individual ODEs, it has an objective function for each mRNA m_i. The form of J_i, the objective function of mRNA m_i is a weighted sum of squared residuals. More detailed information on the objective functions in PEDI is described in Supplementary Section S1.

2.3 Initial optimization process

PEDI decomposes a gene circuit model into individual rate equations. This process involves uncoupling of coupled rate equations. To estimate the time evolution of the mRNAs from the decomposed rate equations, we first need to estimate the time evolution of the transcription factors of each gene in the model. Thus, the first step of our framework is to generate the initial estimate of each protein copy at the M time points (i.e. t_1 to t_M). To this end, we estimate $\hat{p}_i(t_j)$ by applying the time-integral form in Equation (3), using the time series sample average of m_i and using the trapezoidal rule to approximate the numerical integration. This estimates the mean levels of the transcription factors of each gene g_i at the M time points, making the evaluation of the transcriptional kinetic function of each mRNA m_i at the M time points possible.

Using the initial estimates of the protein levels at the M time points, PEDI sets out to estimate the mean time-course of m_i by optimizing the value of θ_i. To estimate the mean time evolution of m_i, we once again use the time-integral form in Equation (3) and apply the trapezoidal rule to approximate the integration of the rate equation of $\hat{m}_i$. This approximate integration is used to compute $\hat{m}_i$ for each i with a given parameter combination, which is then used in a metaheuristic optimization—such as SA and genetic algorithms—to test the fitness of each parameter combination and to find the initial estimate of the optimal θ_i. This parameter optimization process is largely independent of the values of the parameters in a model and remains efficient even when a combination of the parameter values makes the timescale of some rate equations widely different and the ODEs stiff.

While facilitating an efficient and scalable parameter estimation, a model decomposition involving the linear approximation of the time integral of each rate equation may not result in a high-quality estimate, especially when the model of interest is highly nonlinear or the given time series data are sparse. In addition, such a linearization inevitably introduces integration errors, making the assessment of the prediction error for each parameter combination difficult. More detailed information on the initial optimization process is described in Supplementary Section S1.

2.4 Parameter estimation refinement

To improve the accuracy of the numerical integration and the parameter estimation, PEDI next performs a simulation of the ODE model, given the current estimate of θ. This simulation-based numerical integration not only gives a much more accurate picture in terms of the performance of the current estimate but also generates an arbitrary number of data points for each $\hat{m}_i$ and $\hat{p}_i$. From this simulation, we generate L_m+1 equally spaced $\hat{m}_i$ data points between t_0 and t_M for each mRNA, where we set $L_m = c_m M$ for some integer $c_m > 1$. The value of L_m may come with additional constraints depending on the choice of a numerical integration method.

By using the simulated mRNA data points, PEDI attempts to better estimate the time evolution of proteins than the simple linear interpolation that is used in the initial estimate. To this end, we first adjust the simulated data of $\hat{m}_i$ so that they can better reflect the time evolution of the sample mean of each m_i. Let $d_m + 1$ be the number of simulated mRNA data points in each time interval between time points t_j and t_{j+1} (i.e. $d_m = c_m$). Then, to estimate the $d_m + 1$ mRNA points in this time interval, we adjust every simulated data point between t_j and t_{j+1} by considering the difference between the sample mean and the simulated data point of mRNA m_i. Specifically, by letting Δt be the time interval between t_j and t_{j+1} and $\check{m}_i$ be a time-dependent variable that represents the $L_m + 1$ adjusted $\hat{m}_i$ data points, we express $\check{m}_i(t_j + q_m\Delta t)$ for all $q_m \in \{0, 1/d_m, 2/d_m, \ldots, 1\}$ as follows:

$$\check{m}_i(t_j + q_m\Delta t) = \hat{m}_i(t_j + q_m\Delta t) + (1 - q_m)r_j + q_m r_{j+1}, \qquad (4)$$

where r_j is $\overline{m}_{ij} - \hat{m}_i(t_j)$. This definition makes sure that, at each time point t_j, we have $\check{m}(t_j) = \overline{m}_{ij}$. This allows us to use the additional data points from $\check{m}_i$ to make the interpolation of $\overline{m}_{ij}$ and, in turn, the estimation of the time evolution of p_i more accurate than the ones based on a simple linearization.

By using the $L_m + 1$ data points of $\check{m}_i$ between t_0 and t_M, we generate equally spaced $L_p + 1$ data points for each p_i. Here, we require that $L_p \equiv d_p M$ be smaller than L_m where d_p is a positive integer. We denote by $\check{p}_i$ a time-dependent variable that represents the $L_p + 1$ data points of p_i. To compute the values of $\check{p}_i$ at the $L_p + 1$ time points, we first set $\check{p}_i(t_0)$ to be $\mu_{p_{i0}}$. Next, we iteratively compute the next data point of $\check{p_i}$ for the other L_p time points. To this end, we integrate the rate equation of p_i between each time interval $\Delta t/d_p$ using the L_m/L_p data points of $\check{m}_i$ within this time interval (see Supplementary Section S1 for details).

Using the $L_p + 1$ time points of the newly generated protein variables, $\check{p}(t) \equiv (\check{p}_1(t), \ldots, \check{p}_N(t))$, we can better interpolate the dynamics of the transcription factors of gene g_i and estimate the dynamics of $\hat{m}_i(t_j)$ from the decomposed rate equation of m_i for a given parameter combination of θ_i. Thus, applying this approach for the calculation of J_i within an optimization method, we can search for a parameter combination $\hat{\theta}_i$ of θ_i that increases the quality of the estimate (see Supplementary Section S1 for details).

By using $\hat{\theta} \equiv (\hat{\theta}_1, \cdots, \hat{\theta}_N)$ generated from this optimization, we simulate the model and calculate the sum of J_i. We repeat the parameter refinement process until a given termination condition is satisfied (e.g. until the value of the sum of J_i stabilizes). For the next iteration of the refinement process, if the current error is smaller than the previous one, we use the current $\hat{\theta}$ as the seed parameter values for the next iteration. Otherwise, we select the current $\hat{\theta}$ over the previous one at probability of $p_{max}\exp(1 - \epsilon_c/\epsilon_o)$ where p_{max} is the maximum probability of choosing the current estimate, ϵ_c is the current sum of J_i and ϵ_o is the previous one. That is, if the error from the current $\hat{\theta}$ is worse than the previous one, the probability of accepting the current $\hat{\theta}$ for the next round becomes smaller. The detailed information of specific configurations of PEDI used in the Section 3 of this article is described in Supplementary Sections S1 and S2.

2.5 Prediction error

Optimization-based parameter estimation methods may have different objective functions. To compare the accuracy of estimated parameters of various parameter estimation methods objectively and without depending on any specific objective functions, we define the prediction error of the i-th mRNA as follows:

$$PE_i = \sum_{j=1}^{M} \frac{|\hat{m}_i(t_j) - \overline{m}_{ij}|}{|\overline{m}_{ij} - \mu_{m_{ij}}| + \epsilon}, \qquad (5)$$

where ε is a small fixed value and the prediction error of the model as follows:

$$PE = \sum_{i=1}^{N} PE_i. \qquad (6)$$

In other words, we defined the prediction error to be the sum of the difference between the sample mean and the estimate with respect to the difference between the sample mean and the true mean at each time point and for each mRNA. As this definition of prediction error depends on the true mean of mRNAs—whose values are hidden from objective functions—this prediction error can be more objective to compare parameter estimation methods than using a specific objective function (e.g. sum of mean squared error). However, this definition can only be used when mRNA data are synthesized from a model, as the true mean values are not available in real biological systems. In this study, we set ε to be 0.1.

3 RESULTS

3.1 Models

To test the performance of our parameter estimation framework, we constructed three different gene circuit models, $\mathcal{M}1$, $\mathcal{M}2$ and $\mathcal{M}3$ (see Fig. 2). The first system is a three-gene circuit (Fig. 2A). In this system, the transcription of gene g_1 is upregulated by protein p_2 and downregulated by protein p_3. The transcription of gene g_2 is repressed by protein p_1, forming a negative feedback loop of gene g_2. Such a regulatory structure can be seen, for example, in the phage-λ lysis-lysogeny decision circuit in which *CII* upregulates synthesis of *CI*, *CI* in turn downregulates synthesis of *CII* and *Cro* downregulates synthesis of *CI* (Arkin *et al.*, 1998). Provided that the level of protein p_2 is high and the level of protein p_1 is low, this system exhibits a complex transient behavior. In this setting, protein p_1 initially increases rapidly because of the upregulation facilitated by protein p_2, and this increase in protein p_3 downregulates gene g_2, leading to a rapid decrease in protein p_2, which, in turn, downregulates gene g_1 and so on.

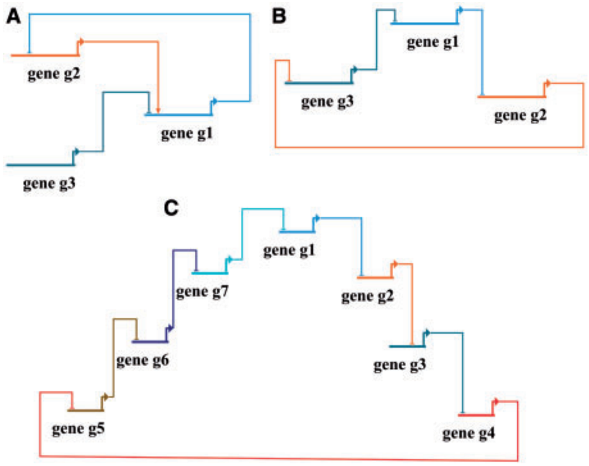

Fig. 2. The schematics of the three gene circuits used in this study. (**A**) The gene circuit structure of model $\mathcal{M}1$. (**B**) The three-gene repressilator structure represented in model $\mathcal{M}2$. (**C**) The seven-gene repressilator structure represented in model $\mathcal{M}3$

We modeled this gene circuit by the following system of ODEs, which we refer to as model $\mathcal{M}1$:

$$\begin{aligned}
\frac{dm_1}{dt} &= \frac{k_3(k_1p_2)^{n_1}}{1+(k_1p_2)^{n_1}+(k_2p_3)^{n_2}} - \gamma_1 m_1,\\
\frac{dm_2}{dt} &= \frac{k_5}{1+(k_4p_1)^{n_3}} - \gamma_2 m_2,\\
\frac{dm_3}{dt} &= k_6 - \gamma_3 m_3,\\
\frac{dp_i}{dt} &= \alpha_i m_i - \beta_i p_i, \quad \text{for } i = 1, 2, 3.
\end{aligned}$$

In this model, we treated the equilibrium rate constants and the maximum transcription rates: $k_1, \ldots, k_6$ as unknown and estimated them from synthetic time series mRNA data by adding a Gaussian noise to the simulated data. We simulated this model from time 0 to 1500 time units and sampled mRNA data at 31 time points. The initial condition for the simulation is $m_1(0) = 0$, $m_2(0) = 500$ and $m_3(0) = 0$, and all protein molecules are initially set to be absent in the system. We note that this parameter estimation problem has an infinite number of suboptimal solutions. This is because the transcriptional regulation kinetic function of p_1 can be simplified to $k_3(k_1p_2)^{n_1}/((k_1p_2)^{n_1} + (k_2p_3)^{n_2})$ if $k_1 \times p_2$ or $k_2 \times p_3$ is assumed to be always much greater than 1. In such a case, there is an infinite number of combinations of k_1 and k_2 with the same $k_2^{n_2}/k_1^{n_1}$ ratio that produce the same dynamics. As a result, we could obtain an infinite number of optimal solutions if this assumption were to be satisfied. However, as the initial amounts of proteins p_2 and p_3 are zero, those solutions may just be suboptimal and may not capture the initial transient behavior well. Thus, it is challenging to find an optimal solution that can capture the initial transient behavior without being stuck in one of those suboptimal solutions. Supplementary Table S1 shows the values of the parameters used in the simulation.

The second and third models are both based on a gene circuit structure, which has a potential to exhibit a sustained oscillation. This gene circuit is called *repressilator*, which was synthetically constructed to exhibit an oscillatory behavior based on transcription regulation with cyclic repression (Elowitz and Leibler, 2000). We model the mean time evolution of n-gene repressilator by the following system of ODEs:

$$\begin{aligned}
\frac{dm_i}{dt} &= \frac{k_{pi}}{1+(k_{ei}p_{(i-1)})^{n_i}} - k_{mi}m_i,\\
\frac{dp_i}{dt} &= \alpha m_i - \beta p_i, \quad \text{for } i = 1, \ldots, n,
\end{aligned}$$

where p_0 is equivalent to p_n. By having an odd number of interacting genes, the repressilator can exhibit an oscillation under specific parameter conditions. Here, we constructed two repressilator models with a different number of genes. One is a three-gene model, which we refer to as model $\mathcal{M}2$, and the other one is a seven-gene model, which we refer to as model $\mathcal{M}3$ (Fig. 2B and C). Given the parameter combinations that we selected (see Supplementary Tables S2 and S3), these two models exhibit oscillatory dynamics as the eigenvalues of the Jacobian matrix at the fixed point contain complex numbers. We set the initial condition of model $\mathcal{M}2$ to be $m_1(0) = 50$, $p_1(0) = 500$, and the other molecular species to be initially absent in the system. For the initial condition of model $\mathcal{M}3$, we set $m_1(0) = 20$ and the other molecular species to be initially absent in the system. To generate the dataset for the true mean trajectory of the mRNAs in the two-gene circuit models, we simulated each deterministic model and sampled the mRNA levels at 31 equally spaced time points. Using these true mean trajectory datasets, we later added the noise term for each data point for each sample and generated the sample mean of each mRNA.

3.2 Comparison using synthetic data

In this study, we used as the optimization component in PEDI an adaptive SA algorithm (Kirkpatrick *et al.*, 1983) in which the parameter to control the temperature schedule can be changed. To measure the improvement made by PEDI, we also used the SA algorithm—without the PEDI framework—for the parameter estimation of models $\mathcal{M}1$ and $\mathcal{M}2$ and compared the performance between PEDI and SA (see Supplementary Section S3). Our results demonstrated that PEDI improved the consistency and the accuracy of parameter estimation while increasing the runtime efficiency.

Next, we compared PEDI with three state-of-the-art parameter estimation methods, namely, the SRES (Runarsson and Yao, 2000), the HEKF + MM (Lillacci and Khammash, 2010) and the TDDM (Jia *et al.*, 2011). SRES is an evolutionary optimization algorithm, which was reported to be among the best candidates for parameter estimation of biological models in previous comparative studies (Moles *et al.*, 2003; Sun *et al.*, 2012). HEKF + MM is a hybrid parameter estimation method, which first applies the hybrid extended Kalman filter and then, if necessary, applies the moment-matching method as the refinement step. TDDM is another model decomposition-based method, which avoids costly ODE simulations by estimating the slopes of smooth piecewise polynomial functions that interpolate observed data. While PEDI, HEKF and TDDM were all implemented in Matlab, the moment-matching method was implemented in C. SRES was implemented in Matlab, but it calls a C library for stochastic ranking computations. Thus, we expected that the efficiency of HEKF + MM and SRES might be overestimated from the comparisons based on computational time.

To generate a sample data point of a given mRNA at a given time point, we sampled a value by adding to the true mean of mRNA a Gaussian random variable with mean 0 and variance being the time average of the true means of this mRNA. For the experiments with models $\mathcal{M}1$ and $\mathcal{M}2$, we generated four time series data samples and used the average of the four as the observed dataset. To analyze the performance of parameter estimation methods with a time series dataset at a higher noise level, we generated only a single time series data sample and used this as the observed dataset in the experiments with model $\mathcal{M}3$. For the parameter estimation of each of the three models, we ran each method 10 times. Detailed information about the specific settings used in the three parameter estimation methods in this comparison is described in Supplementary Section S4.

To evaluate the performance of each method, we used four basic criteria, the computational efficiency, the quality of reconstructed dynamics, the accuracy of estimated parameters and the quality of predictability. The computational efficiency was measured by computing the average runtime of the 10 runs from each method. The quality of reconstructed dynamics was analyzed by measuring the average, the smallest and the largest prediction errors of each method, whereas the accuracy of estimated parameters was analyzed by measuring the average relative error of the estimated parameter set with the smallest prediction error for each method. Finally, the quality of predictability was analyzed by extrapolating mRNA data at the next k observed time points using the parameter set with the smallest prediction error of each method; we measured the mean squared distance between the estimated data points and 100 samples that are generated for each of the k observed data points where we set k to be 1, 3 and 5.

The results from the comparison of the four methods using model $\mathcal{M}1$ are summarized in Table 1. Here, PEDI outperformed the other methods in three of the four criteria. Both SRES (with 100 generations of evolution) and TDDM performed poorly in terms of efficiency and accuracy. Although HEKF + MM was the most efficient method in this experiment, PEDI was also relatively efficient (0.5 min versus 2.2 min). By comparing the best parameter solutions of these two methods, we found that PEDI generated the most accurate estimate (Fig. 3A and Supplementary Fig. S9). HEKF + MM produced parameter sets with negative values in 7 of the 10 runs. This is due to the fact that the moment-matching algorithm used an unconstrained local optimization technique (Lillacci and Khammash, 2010). To analyze the typical behavior of each method, we measured the average prediction error and the average relative parameter error (Table 1 and Supplementary Fig. S10). These show that PEDI generated not only the best parameter solution but also the highest quality parameter solutions on average. PEDI was also able to extrapolate the mRNA levels at next few time points more accurately than the other three. Taken together, we found that PEDI was able to generate the highest quality parameter solutions efficiently among the four methods.

Table 1. Comparison of the results from model $\mathcal{M}1$

Method	PEDI	SRES[a]	HEKF + MM	TDDM
Runtime	2.2 min	8.1 min	**0.5 min**	11.3 min
Average PE	**225.9**	1584.0	N/A[b]	2145.0
Best PE	**127.5**	836.5	207.0	2144.9
Worst PE	**359.5**	2424.5	N/A[b]	2145.1
Best param[c]	**0.046**	8490.5	0.12	9382.6
Pred(1)[d]	**33.6**	378.8	40.0	3095.9
Pred(3)[d]	**31.8**	410.4	36.3	2831.9
Pred(5)[d]	**33.4**	392.8	38.6	2620.7

[a] With 100 generations. [b] Because seven runs resulted in negative parameter values. [c] The average relative error of the best parameter solution. [d] Pred(k) indicates the mean squared distance of the next k time points. The comparison criteria are as follows: the average runtime; the average, best and worst prediction errors; the average relative error of the best parameter set; and the quality of data extrapolation. Each bold face indicates the best among the four.

Next, we analyzed the performance of the four methods using model $\mathcal{M}2$. These results are summarized in Table 2. In this experiment, PEDI outperformed the other methods in three of the four criteria. Although HEKF + MM was once again the most efficient method in this experiment, both PEDI and TDDM had comparable speed with HEKF + MM. Again, PEDI substantially outperformed the other methods in terms of the quality of the estimates in this experiment. By comparing the best parameter solution of each method, we found that PEDI generated high-quality estimates with the smallest prediction error (Fig. 3B and Supplementary Fig. S11). Even the worst parameter solution of PEDI had a lower prediction error than the best solution of any of the other methods (Table 2). To analyze the typical behavior of each method, we measured the average prediction error and the average relative parameter error of the best parameter solution (Table 2 and Supplementary Fig. S12). These show that PEDI consistently outperformed the other methods and produced much higher quality parameter solutions in a computationally efficient fashion. Furthermore, PEDI was also able to extrapolate the mRNA levels at next few time points substantially more accurately than the other three.

We next applied the four methods to model $\mathcal{M}3$. The comparison of the four methods is summarized in Table 3. Among the four methods, the two model decomposition-based methods were much more efficient than the other two methods. TDDM was the fastest with its average runtime being 26 min, and PEDI was a close second with its average runtime being 33.5 min. Although HEKF + MM was the most efficient method for the 2 three-gene models (Tables 1 and 2), it was more than four times slower than PEDI in this experiment. These results, coupled with the results from models $\mathcal{M}1$ and $\mathcal{M}2$, show that model decomposition-based methods can scale better than typical parameter estimation methods. The least computational efficient method was SRES. We ran SRES with 2000 generations of evolution, which, on average, took more than eight times longer than PEDI did. However, the quality of the estimates from SRES was just on a par with that of TDDM.

In the other three criteria, PEDI outperformed the other three methods substantially. In terms of the quality of estimated parameters, PEDI performed substantially better than the other three methods. By comparing the best parameter solution of

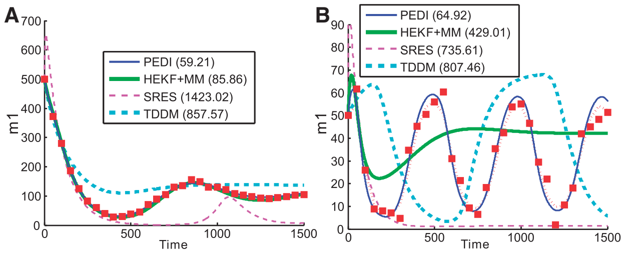

Fig. 3. Comparison of the four methods based on the reconstructed dynamics with the smallest prediction error from models $\mathcal{M}1$ and $\mathcal{M}2$. (**A**) The results of m_1 in $\mathcal{M}1$. (**B**) The results of m_1 in $\mathcal{M}2$. Here, each value within parentheses next to each method indicates the prediction error for a given mRNA, the red square points indicate the observed data points, and the dotted red lines indicate the true average trajectories

Table 2. Comparison of the results from model $\mathcal{M}2$

Method	PEDI	SRES[a]	HEKF + MM	TDDM
Runtime	15.0 min	268.7 min	**10.6 min**	10.9 min
Average PE	**781.4**	2119.9	N/A[b]	2197.1
Best PE	**227.0**	1854.0	2964.2	2196.8
Worst PE	**1590.9**	2502.7	N/A[b]	2197.2
Best param[c]	**0.091**	1592.7	0.64	0.45
Pred(1)[d]	**26.2**	1139.2	1164.0	3021.6
Pred(3)[d]	**26.7**	893.0	1274.9	2501.2
Pred(5)[d]	**22.9**	767.7	1099.4	1765.0

[a]With 400 generations. [b]Because nine runs resulted in negative parameter values. [c]The average relative error of the best parameter solution. [d]Pred(k) indicates the mean square distance of the next k time points. The comparison criteria are as follows: the average runtime; the average, best and worst prediction errors; the average relative error of the best parameter set; and the quality of data extrapolation. Each bold face indicates the best among the four.

Table 3. Comparison of the results from model $\mathcal{M}3$

Method	PEDI	SRES[a]	HEKF + MM	TDDM
Runtime	33.5 min	279.5 min	143.5 min	**26.0 min**
Average PE	**1123.5**	2399.6	N/A[b]	2250.1
Best PE	**647.1**	2174.6	1477.2	2195.9
Worst PE	**2215.5**	2735.5	N/A[b]	2587.5
Best param[c]	**0.16**	1.6	0.36	0.27
Pred(1)[d]	**112.4**	595.3	176.1	558.5
Pred(3)[d]	**91.1**	525.0	150.7	577.6
Pred(5)[d]	**103.5**	461.1	182.2	569.0

[a]With 2000 generations. [b]Because four runs resulted in negative parameter values. [c]The average relative error of the best parameter solution. [d]Pred(k) indicates the mean squared distance of the next k time points. The comparison criteria are as follows: the average runtime; the average, best and worst prediction errors; the average relative error of the best parameter set; and the quality of data extrapolation. Each bold face indicates the best among the four.

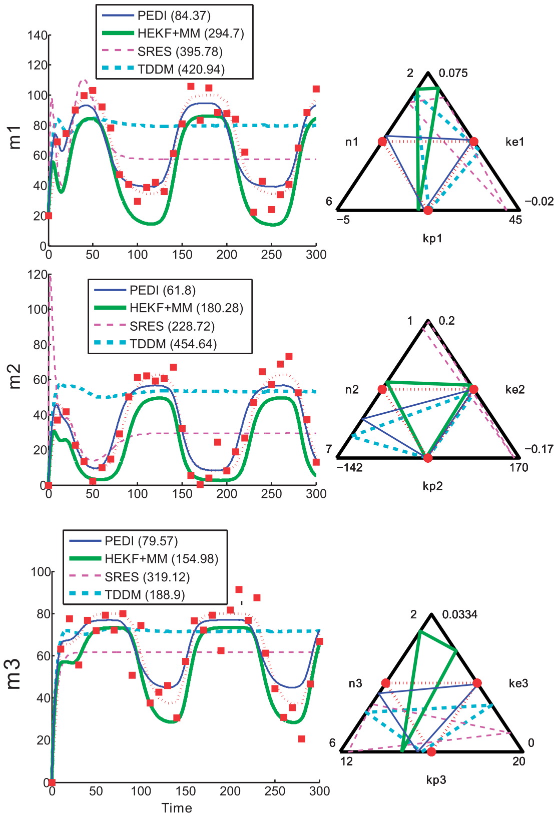

Fig. 4. Comparison of the four methods based on the estimated parameter set with the smallest prediction error from model $\mathcal{M}3$. This shows the results for mRNAs m_1, m_2 and m_3. The left-hand side panels show the comparison for the reconstructed trajectory. The numbers in the parentheses indicate the prediction error. The dotted red curve shows the true trajectory, whereas the red square points indicate the synthetic data points. The right-hand side panels show the comparison of the parameter combination generating the estimated trajectory for the four methods. Here, the red point in the middle of each side for the parameter comparison indicates the true value of each parameter

each method, PEDI came out to be the most accurate one with its prediction error being at least twice as good as the other methods. PEDI outperformed the other methods in terms of the accuracy of both the reconstructed dynamics and the parameter values (Fig. 4 and Supplementary Fig. S13). In addition, the average parameter solution of PEDI had at least 30% lower prediction error than the best parameter solution of any other (Table 3). Furthermore, the parameter sets from PEDI were the closest to the true parameter set on average, and the best parameter solution from PEDI had only 16% error to the true parameter set (Supplementary Fig. S14 and Table 3). PEDI was also able to extrapolate the data at the next time points much more accurately than the other three. As data extrapolation of biochemical dynamics—especially those with transient behaviors—is a challenging problem, this highlights the importance of high-throughput parameter estimation methods, which can generate high-quality parameter sets in a timely fashion to ultimately facilitate construction of a predictive model for given biological phenomena. To test whether our performance results remain intact in parameter estimation of a variant of seven-gene repressilator model, we modified $\mathcal{M}3$ by adding a repression connection from g_3 to g_6. Our results show that PEDI produced higher quality parameter solutions much more efficiently than the other methods (see Supplementary Section S5). Taken together, we found that PEDI was able to consistently produce high-quality parameter estimates under various conditions in a computationally efficient matter.

3.3 Parameter estimation with *yeast* microarray data

To compare the performance of the parameter estimation methods using experimental data, we used time series microarray experiments of the genomic expression patterns in the yeast *Saccharomyces cerevisiae* responding to several environmental changes (Gasch *et al.*, 2000). We modeled a gene circuit involving genes GCN4, LEU3 and ILV5. GCN4 is a master regulator of many genes including those for the amino acid biosynthesis pathway (Natarajan *et al.*, 2001). LEU3 is a gene encoding a transcription factor that regulates genes involved in amino acid biosynthesis, whereas ILV5 encodes an enzyme that catalyzes amino acid biosynthesis (Friden and Schimmel, 1988; Zelenaya-Troitskaya *et al.*, 1995).

The network structure of these three genes is reported to follow the network motif called the coherent type 1 feed-forward loop (Mangan and Alon, 2003). We described the model of this feed-forward gene circuit by the following system of ODEs:

$$\begin{aligned}
\frac{dm_1}{dt} &= f_1(t), \\
\frac{dm_2}{dt} &= \alpha\frac{(k_{XY}p_1)^{n_1}}{1+(k_{XY}p_1)^{n_1}} - k_1 m_2, \\
\frac{dm_3}{dt} &= \alpha\left(\frac{(k_{XZ}p_1)^{n_2}}{1+(k_{XZ}p_1)^{n_2}}\right)\left(\frac{(k_{YZ}p_2)^{n_3}}{1+(k_{YZ}p_2)^{n_3}}\right) - k_2 m_3, \\
\frac{dp_i}{dt} &= \beta(m_i - p_i), \quad \text{for } i = 1, 2, 3,
\end{aligned}$$

where f_1 is a piecewise polynomial function of t and m_1, m_2 and m_3 are the mRNA copies of GCN4, LEU3 and ILV5, respectively, whereas p_1, p_2 and p_3 are proteins Gcn4p, Leu3p and Ilv5p, respectively. As in Cao and Zhao (2008), we estimated the time evolution of m_1 with a smoothing method based on spline.

In this experiment, we set $\alpha = 1$ and $\beta = 0.5$ to have protein stability higher assuming that each transcription rate is close to the maximum value when the corresponding expression profile is at the highest. With this setting, we estimated eight unknown parameters in the regulation of genes LEU3 and ILV5. In this experiment, we did not use HEKF + MM, and we just compared PEDI with SRES (with 400 generations of evolution) and TDDM. This is because the dataset contains only one time course microarray data sample with eight time points, with which we could not satisfactorily estimate the covariance matrix that HEKF + MM demanded. To quantify the performance, we measured the mean squared error of each estimate with respect to the observed mRNA data points, and we compared the best parameter solution from each method that gave the lowest error. We found that the error from the best solution of PEDI was 0.02, whereas the errors from the best solutions of SRES and TDDM were more than 0.2 and 0.3, respectively (Supplementary Table S4). This shows that PEDI was able to approximate the dynamics of the RNA copy of LEU3 and ILV5 well. Indeed, the reconstructed dynamics from the best parameter solution of PEDI shows a close agreement between the results from PEDI and the time series microarray data (Fig. 5). The average computational time of PEDI, SRES and TDDM is 3.3 min, 143.3 min and 1.7 min, respectively. These results once again show that PEDI was able to generate a high-quality estimate efficiently.

By using the best parameter solution from PEDI, we next analyzed regulatory mechanisms of this gene circuit. By looking at the parameters controlling the binding affinity of Gcn4p to the *cis*-regulatory elements of gene LEU3 and gene ILV5, we found that Gcn4p had close to 20% higher binding affinity to the binding site for gene ILV5. As the protein–DNA binding cooperativity estimated by PEDI was high (Supplementary Table S4), we expect the change in transcription rates of ILV5 to be a switch-like, sigmoidal function of both Gcn4p and Leu3p. Thus, the differential binding affinity of Gcn4p allows for a delay in turning the ILV5 gene on after Gcn4p is turned on, as this AND-gate type transcription regulation of ILV5 in this model requires higher concentrations of both Gcn4p and Leu3p to turn ILV5 on. In the normal nonstarvation conditions, Gcn4p level is tightly controlled with a means of a rapid degradation through the ubiquitin pathway. On the other hand, Gcn4p level substantially increases in the amino acid starvation condition (Kornitzer *et al.*, 1994). Our results show that the delay caused by the differential binding affinity in the feed-forward loop may serve as an extra layer of protection to ensure that this amino acid biosynthesis pathway is only activated under the starvation condition. As our hypothesis is based on one type of time series expression dataset with only eight time points, however, it needs to be taken with caution and further analysis is required to contrast it with alternative explanations. Indeed, our hypothesis can be validated experimentally; by changing the Gcn4p binding site for gene ILV5 to have a lower binding affinity, it predicts that the regulation of the amino acid biosynthesis pathway will be disrupted more easily.

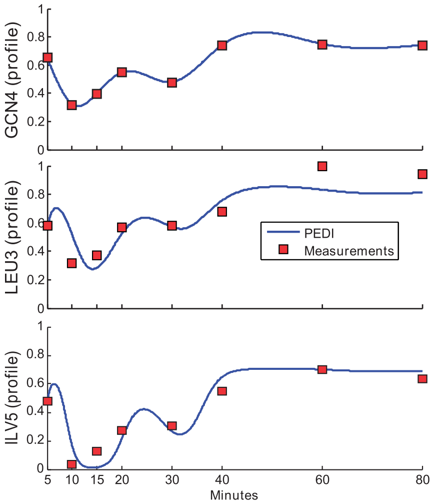

Fig. 5. The reconstructed dynamics from the best parameter solution of PEDI for the yeast feed-forward loop model. The dynamics of mRNA GCN4 was estimated by a smooth piecewise polynomial function. PEDI predicted the dynamics of mRNAs, LEU3 and ILV5

4 DISCUSSION

The parameter estimation problem in modeling of biological systems is challenging, as it usually involves many (often infinite) suboptimal solutions. Efficient and scalable parameter estimation is crucial to the systematic construction of quantitative models that support existing knowledge of complex biological systems and, more broadly, to the success of integrative systems biology going forward. Here, we have introduced a novel computational framework, which, instead of considering general applicability, is customized especially for parameter estimation of gene circuit models. To see how PEDI performs in comparison with state-of-the-art parameter estimation methods, we applied SRES, HEKF + MM and TDDM to the parameter estimation of the same gene circuit models with the same datasets. We found that PEDI consistently gave the most accurate estimates in a computationally efficient matter. To test how PEDI performs given experimental gene expression data, we applied it to modeling of a yeast gene circuit from time series microarray data. We found that the reconstructed dynamics from PEDI closely agreed with the experimental data, and by analyzing the estimated parameter set, we were also able to make a testable hypothesis for an underlying regulatory mechanism of this gene circuit.

Although PEDI can be applied to gene circuits with an arbitrary size and degree of transcriptional interaction connectivity, there are some limitations. For example, PEDI cannot directly support gene regulatory models including transcriptional elongation and posttranscriptional modifications. Although we can relax the conditions of PEDI so as to support more generic biological models by not requiring a model to have the form described by Equation (3), the efficiency and the accuracy of the model decomposition might decrease. While acknowledging the limited scope of the applicability, we believe that the value of a more tailored approach to the gene circuit domain far exceeds such potential drawbacks because of the importance of transcriptional regulation in quantitative understandings of cellular systems. By narrowing down our focus to gene circuit models, our customized approach was able to display two main advantages over the general parameter estimation methods: (i) it can make more appropriate assumptions about the kinds of gene expression data available for parameter estimation and (ii) it can exploit the structural information on gene circuit models—in particular, statistical thermodynamic-based gene circuit models. An additional practical benefit of PEDI is that it is relatively easy to implement in a lower level language such as C.

ACKNOWLEDGEMENTS

The authors thank Gabriele Lillacci for providing us with the source code for the hybrid extended Kalman filter method and the moment matching method and Jiguo Cao for providing us with the yeast time series microarray data.

Funding: This work was supported in part by an Academic Excellence Alliance grant awarded by the King Abdullah University of Science and Technology Office of Competitive Research Funds under the title ‘Association of genetic variation with phenotype at the network and function level’. S.W.’s research was partially supported by Award Number [KUS-CI-016-04], made by King Abdullah University of Science and Technology (KAUST).

Conflict of Interest: none declared.

REFERENCES

Arkin,A. *et al.* (1998) Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected *Escherichia coli* cells. *Genetics*, **149**, 1633–1648.

Baker,S.M. *et al.* (2010) Comparison of different algorithms for simultaneous estimation of multiple parameters in kinetic metabolic models. *J. Integr. Bioinform.*, **7**.

Baugh,L.R. *et al.* (2011) Sensitive and precise quantification of insulin-like mRNA expression in *Caenorhabditis elegans*. *PLoS One*, **6**, e18086.

Cai,L. *et al.* (2006) Stochastic protein expression in individual cells at the single molecule level. *Nature*, **440**, 358–362.

Cao,J. and Zhao,H. (2008) Estimating dynamic models for gene regulation networks. *Bioinformatics*, **24**, 1619–1624.

Church,G.M. (2005) From systems biology to synthetic biology. *Mol. Syst. Biol.*, **1**, 2005.0032.

Elowitz,M.B. and Leibler,S. (2000) A synthetic oscillatory network of transcriptional regulators. *Nature*, **403**, 335–338.

Elowitz,M.B. *et al.* (2002) Stochastic gene expression in a single cell. *Science*, **297**, 1183–1186.

Friden,P. and Schimmel,P. (1988) LEU3 of *Saccharomyces cerevisiae* activates multiple genes for branched-chain amino acid biosynthesis by binding to a common decanucleotide core sequence. *Mol. Cell. Biol.*, **8**, 2690–2697.

Gasch,A.P. *et al.* (2000) Genomic expression programs in the response of yeast cells to environmental changes. *Mol. Biol. Cell*, **11**, 4241–4257.

Golding,I. *et al.* (2005) Real-time kinetics of gene activity in individual bacteria. *Cell*, **123**, 1025–1036.

Ideker,T. *et al.* (2001) A new approach to decoding life: systems biology. *Annu. Rev. Genomics Hum. Genet.*, **2**, 343–372.

Jia,G. *et al.* (2011) Parameter estimation of kinetic models from metabolic profiles: two-phase dynamic decoupling method. *Bioinformatics*, **27**, 1964–1970.

Joo,C. *et al.* (2008) Advances in single-molecule fluorescence methods for molecular biology. *Annu. Rev. Biochem.*, **77**, 51–76.

Kirkpatrick,S. *et al.* (1983) Optimization by simulated annealing. *Science*, **220**, 671–680.

Kitano,H. (2002) Computational systems biology. *Nature*, **420**, 206–210.

Koh,G. *et al.* (2006) A decompositional approach to parameter estimation in pathway modeling: a case study of the Akt and MAPK pathways and their crosstalk. *Bioinformatics*, **22**, e271–e280.

Kornitzer,D. *et al.* (1994) Regulated degradation of the transcription factor gcn4. *EMBO J.*, **13**, 6021–6030.

Kulkarni,M.M. (2011) Digital multiplexed gene expression analysis using the nanostring ncounter system. *Curr. Protoc. Mol. Biol.*, **Chapter 25**, Unit25B.10.

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.

Mangan,S. and Alon,U. (2003) Structure and function of the feed-forward loop network motif. *Proc. Natl Acad. Sci. USA*, **100**, 11980–11985.

Materna,S.C. *et al.* (2010) High accuracy, high-resolution prevalence measurement for the majority of locally expressed regulatory genes in early sea urchin development. *Gene Expr. Patterns*, **10**, 177–184.

Moles,C.G. *et al.* (2003) Parameter estimation in biochemical pathways: a comparison of global optimization methods. *Genome Res.*, **13**, 2467–2474.

Natarajan,K. *et al.* (2001) Transcriptional profiling shows that Gcn4p is a master regulator of gene expression during amino acid starvation in yeast. *Mol. Cell. Biol.*, **21**, 4347–4368.

Newman,J.R.S. *et al.* (2006) Single-cell proteomic analysis of *S. cerevisiae* reveals the architecture of biological noise. *Nature*, **441**, 840–846.

Quach,M. *et al.* (2007) Estimating parameters and hidden variables in non-linear state-space models based on ODEs for biological networks inference. *Bioinformatics*, **23**, 3209–3216.

Raj,A. and van Oudenaarden,A. (2009) Single-molecule approaches to stochastic gene expression. *Annu. Rev. Biophys.*, **38**, 255–270.

Raj,A. *et al.* (2006) Stochastic mRNA synthesis in mammalian cells. *PLoS Biol.*, **4**, e309.

Raser,J.M. and O’Shea,E.K. (2005) Noise in gene expression: origins, consequences, and control. *Science*, **309**, 2010–2013.

Runarsson,T.P. and Yao,X. (2000) Stochastic ranking for constrained evolutionary optimization. *IEEE Trans. Evol. Comput.*, **4**, 284–294.

Schoeberl,B. *et al.* (2002) Computational modeling of the dynamics of the MAP kinase cascade activated by surface and internalized EGF receptors. *Nat. Biotechnol.*, **20**, 370–375.

Schwartz,R. (2008) *Biological Modeling and Simulation: a Survey of Practical Models, Algorithms, and Numerical Methods*. The MIT Press, Cambridge, MA, USA.

Shea,M.A. and Ackers,G.K. (1985) The OR control system of bacteriophage lambda: A physical-chemical model for gene regulation. *J. Mol. Biol.*, **181**, 211–230.

Sherman,M.S. and Cohen,B.A. (2012) Thermodynamic state ensemble models of *cis*-regulation. *PLoS Comput. Biol.*, **8**, e1002407.

Sun,J. *et al.* (2012) Parameter estimation using metaheuristics in systems biology: a comprehensive review. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, **9**, 185–202.

Sun,X. *et al.* (2008) Extended Kalman filter for estimation of parameters in nonlinear state-space models of biochemical networks. *PLoS One*, **3**, e3758.

Suter,D.M. *et al.* (2011) Mammalian genes are transcribed with widely different bursting kinetics. *Science*, **332**, 472–474.

Taniguchi,Y. *et al.* (2010) Quantifying *E. coli* proteome and transcriptome with single-molecule sensitivity in single cells. *Science*, **329**, 533–538.

van Oijen,A.M. (2011) Single-molecule approaches to characterizing kinetics of biomolecular interactions. *Curr. Opin. Biotechnol.*, **22**, 75–80.

Zelenaya-Troitskaya,O. *et al.* (1995) An enzyme in yeast mitochondria that catalyzes a step in branched-chain amino acid biosynthesis also functions in mitochondrial DNA stability. *EMBO J.*, **14**, 3268–3276.

Zenklusen,D. *et al.* (2008) Single-RNA counting reveals alternative modes of gene expression in yeast. *Nat. Struct. Mol. Biol.*, **15**, 1263–1271.

Zhan,C. and Yeung,L.F. (2011) Parameter estimation in systems biology models using spline approximation. *BMC Syst. Biol.*, **5**, 14.

Zwolak,J.W. *et al.* (2005) Parameter estimation for a mathematical model of the cell cycle in frog eggs. *J. Comput. Biol.*, **12**, 48–63.