The Third International Symposium on Optimization and Systems Biology (OSB'09)
Zhangjiajie, China, September 20–22, 2009


# Parameter Optimization in Network Dynamics Including Unmeasured Variables by the Symbolic-numeric Approach*

Masahiko Nakatsui[1] Katsuhisa Horimoto[1,2]

[1]Computational Biology Research Center, National Institute of Advanced Industrial Science and Technology, Tokyo 135-0064, Japan
[2]Institute of Systems Biology, Shanghai University, Shanghai 200444, China

**Abstract** In this report, we propose a new symbolic-numeric method of differential algebra and a numerical parameter optimization algorithm. First, we utilize differential elimination, which is an algebraic approach for rewriting a system of differential equations into another equivalent system, to derive the constraints between the kinetic parameters from the original system. Second, we introduce these constraints to effectively optimize the parameters into a genetic algorithm, Real-Coded Genetic Algorithms (RCGAs), which is a numerical parameter optimizing method. To evaluate the ability of our method, we performed a simulation study for an artificial biological network including one measured and three unmeasured molecules. As a result, our method, the symbolic-numeric method of differential elimination and RCGAs, precisely estimated the kinetic parameters in the simulated network, while RCGAs failed. Thus, our method is useful for analyzing the dynamics of a biological network including unmeasured molecules.

**Keywords** symbolic-numeric method, differential elimination, real-coded genetic algorithms (RCGAs)

## 1 Introduction

The investigation of network dynamics is a major issue in systems biology [1]. A network model for describing the kinetics of constituent molecules is usually first constructed with reference to the biological knowledge, and then the model is mathematically expressed by differential equations, based on the chemical reactions underlying the kinetics. Finally, the kinetic parameters in the model are estimated by various parameter optimization techniques [2], from the time series data measured for the constituent molecules. In the last stage, we cannot always obtain the data measured for all of the constituent molecules, due to limitations of measurement techniques and ethical constraints. Thus, one of the issues we should resolve is that the parameters are estimated from the data for only some of the constituent molecules. Unfortunately, it is frequently difficult to estimate the parameters in such a network model including unmeasured variables.

*This work was partly supported by a project grant, 'Development of Analysis Technology for Gene Functions with Cell Arrays', from The New Energy and Industrial Technology Development Organization (NEDO).

Recently, Boulier and his colleagues developed differential elimination [3], derived from the Rosenfeld-Gröbner base [4], and applied it to solve the issue of network dynamics including unmeasured variables [3, 5]. In their application, the equations rewritten by differential elimination from the system of differential equations were utilized to estimate the initial values for the parameter optimization, by Newton-type numerical optimization.

Here, we propose a new method for optimizing the parameters in a network including unmeasured variables, in combination with a genetic algorithm [6, 7] and differential elimination [3]. Our method partially utilizes a technique from a previous study [3] about the introduction of differential elimination into the parameter optimization in a network including unmeasured variables. Instead of using differential elimination for estimating the initial values for the following parameter optimization, the equations reduced by differential elimination are directly introduced as the constraints into the genetic algorithm for the parameter optimization. To validate the effectiveness of the constraint introduction, we performed a simulation where we assumed that a network was composed of four molecules, and the data for only one molecule among them were measured. The parameter values estimated by our method were compared with those generated by the genetic algorithm without the constraints.

# 2 Results and Discussion

## 2.1 Analyzed Model

We prepared a network model containing four molecules, as shown in Fig. 1 (A). The differential equation representation of Fig. 1 is shown in eq. 1.

$$\begin{aligned}
\frac{dx_1(t)}{dt} &= k_{21}x_2 + k_{31}x_3 + k_{41}x_4 - k_{e1}x_1(t) \qquad (1)\\
\frac{dx_2(t)}{dt} &= -k_{e2}x_2 - k_{21}x_2\\
\frac{dx_3(t)}{dt} &= -k_{e2}x_3 - k_{31}x_3\\
\frac{dx_4(t)}{dt} &= -k_{e2}x_4 - k_{41}x_4
\end{aligned}$$

In the model, we assume that the molecules, $x_2$, $x_3$, and $x_4$, activate $x_1$, with linear relationships. Notably, only the amount of molecule $x_1$ among the four molecules is measured.

We generated the reference curve of $x_1$ for estimating the kinetic parameter set under the following initial conditions for each molecule and kinetic constant: $x_1(0) = 10.0, x_2(0) = 130.0, x_3(0) = 80.0, x_4(0) = 170.0, k_{21} = 0.01, k_{31} = 0.1, k_{41} = 10.0, k_{e1} = 5.0, k_{e2} = 3.0$. The generated reference curve ($0 \leq t \leq 1$ at 0.01 intervals) is shown in Fig. 1 (B).

## 2.2 Constraints from Differential Elimination

We derived the constraint equations from differential equations of the analyzed network model (eq. 1), by using differential elimination (see details in 3.2).

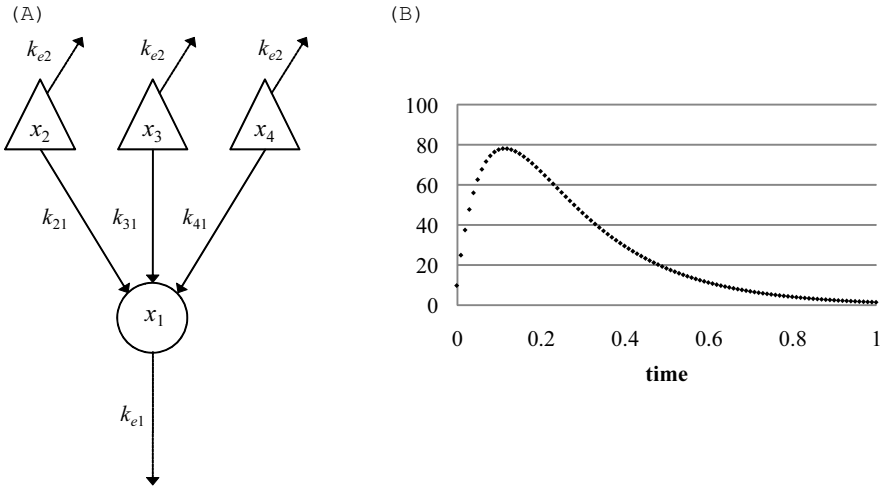


Figure 1: Analyzed model and reference curve
(A) Analyzed model. (B) Reference curve of $x_1$ generated by given parameter values.

$$
\begin{aligned}
C_{1,t} \quad = \quad & \frac{1.0}{k_{21}(k_{21}-k_{31})(k_{21}-k_{41})}\Big(\frac{d^3}{dt^3}x_1(t)+(k_{31}+k_{41}+k_{e1}+2k_{e2})\frac{d^2}{dt^2}x_1(t) \\
& +(k_{31}k_{41}+k_{31}k_{e1}+k_{41}k_{e1}+k_{31}k_{e2}+k_{41}k_{e2}+2k_{e1}k_{e2}+k_{e2}^2)\frac{d}{dt}x_1(t) \\
& +k_{e1}(k_{31}+k_{e2})(k_{41}+k_{e2})x_1(t)-k_{21}(k_{21}-k_{31})(k_{21}-k_{41})x_2(t)\Big)=0
\end{aligned} \tag{2}
$$

$$
\begin{aligned}
C_{2,t} \quad = \quad & \frac{1.0}{(k_{21}-k_{31})k_{31}(k_{31}-k_{41})}\Big(\frac{d^3}{dt^3}x_1(t)+(k_{21}+k_{41}+k_{e1}+2k_{e2})\frac{d^2}{dt^2}x_1(t) \\
& +(k_{41}(k_{e1}+k_{e2})+k_{21}(k_{41}+k_{e1}+k_{e2})+k_{e2}(2k_{e1}+k_{e2}))\frac{d}{dt}x_1(t) \\
& +k_{e1}(k_{21}+k_{e2})(k_{41}+k_{e2})x_1(t))+x_3(t)=0
\end{aligned} \tag{3}
$$

$$
\begin{aligned}
C_{3,t} \quad = \quad & \frac{1.0}{(k_{21}-k_{41})(k_{31}-k_{41})k_{41}}\Big(\frac{d^3}{dt^3}x_1(t)+(k_{21}+k_{31}+k_{e1}+2k_{e2})\frac{d^2}{dt^2}x_1(t) \\
& +(k_{21}k_{31}+k_{21}k_{e1}+k_{31}k_{e1}+k_{21}k_{e2}+k_{31}k_{e2}+2k_{e1}k_{e2}+k_{e2}^2)\frac{d}{dt}x_1(t) \\
& +k_{e1}(k_{21}+k_{e2})(k_{31}+k_{e2})x_1(t)+(k_{21}-k_{41})k_{41}(-k_{31}+k_{41})x_4(t)\Big)=0
\end{aligned} \tag{4}
$$

$$
\begin{aligned}
C_{4,t} \quad = \quad & \frac{d^4}{dt^4}x_1(t) + (k_{21}+k_{31}+k_{41}+k_{e1}+3k_{e2})\frac{d^3}{dt^3}x_1(t) \\
& +(k_{21}k_{31}+k_{21}k_{41}+k_{31}k_{41}+k_{21}k_{e1}+k_{31}k_{e1}+k_{41}k_{e1} \\
& +2k_{21}k_{e2}+2k_{31}k_{e2}+2k_{41}k_{e2}+3k_{e1}k_{e2}+3k_{e2}^2)\frac{d^2}{dt^2}x_1(t) \\
& +(k_{31}(k_{41}(k_{e1}+k_{e2})+k_{e2}(2k_{e1}+k_{e2}))+k_{21}(k_{41}(k_{e1}+k_{e2})+k_{31}(k_{41}+k_{e1}+k_{e2}) \\
& +k_{e2}(2k_{e1}+k_{e2}))+k_{e2}(k_{41}(2k_{e1}+k_{e2})+k_{e2}(3k_{e1}+k_{e2})))\frac{d}{dt}x_1(t) \\
& +k_{e1}(k_{21}+k_{e2})(k_{31}+k_{e2})(k_{41}+k_{e2})x_1(t) = 0
\end{aligned} \tag{5}
$$

In the above four equations, $C_{4,t}$ is an equation of $x_1$, its derivatives, and the parameters, and the three remaining constraints are equations of $x_1$, its derivatives, the parameters, and either $x_2$, $x_3$, or $x_4$. Since the solutions of $x_2$, $x_3$, and $x_4$ can be analytically obtained from eq. 1, we can numerically calculate the four constraint values by using the parameter values. The above constraints (eq. 2 - eq. 5) are introduced into the genetic algorithm as the constraints for parameter estimation, as described below.

## 2.3 Objective Function in Symbolic-Numeric Method

In our study, the objective function is composed of two terms: one is the standard error function between the estimated and sample data, and the other is the constraints (eq. 2 - 5) obtained by differential elimination (DE constraints). The error function is defined as follows: Suppose that $x_{1,t}^{\mathrm{c}}$ is the time-course data at time $t$ of $x_1$ calculated by using the estimated parameter values, and $x_{1,t}^{\mathrm{s}}$ represents the sampling data (reference curve) at time $t$ of $x_1$. The sum of the absolute value of the relative error between $x_{1,t}^{\mathrm{c}}$ and $x_{1,t}^{\mathrm{s}}$ gives the total relative error, $E$.

$$
E = \sum_{t=1}^{T} \left| \frac{x_{1,t}^{\mathrm{c}} - x_{1,t}^{\mathrm{s}}}{x_{1,t}^{\mathrm{s}}} \right| \tag{6}
$$

As usual, the above error function is an objective function for RCGAs ($\mathrm{OF}_{\mathrm{RCGAs}}$), i.e.,

$$
\mathrm{OF}_{\mathrm{RCGAs}} = E. \tag{7}
$$

Next we obtain the DE constraints as the linear combination of eqs. 2 - 5, as follows:

$$
C_{\mathrm{DE}} = \sum_{l=1}^{L}\sum_{t=1}^{T} |C_{l,t}| \tag{8}
$$

where $L = 4$ and $T = 100$.

Finally, we introduce the DE constraints into the objective function of the RCGAs: we defined the objective functions for our symbolic-numeric method ($\mathrm{OF}_{\mathrm{SN}}$) as:

$$
\mathrm{OF}_{\mathrm{SN}} = \alpha \mathrm{OF}_{\mathrm{RCGAs}} + (1-\alpha) C_{\mathrm{DE}} \tag{9}
$$

where $\alpha = 0.99975$. The computational task is to determine a set of parameter values that minimize to $\mathrm{OF}_{\mathrm{SN}}$ (see details in 3.3).

## 2.4 Estimation of Kinetic Parameters

We performed RCGAs and our symbolic-numeric method to estimate the kinetic parameter set ($k_{21}$, $k_{31}$, and $k_{41}$) 200 times from the data in Fig. 1 (B). We successfully obtained 132 parameter sets by RCGAs and 90 sets by the symbolic-numeric method: we regarded the outcome as a success, when $E/T$ in 2.3 satisfies less than 0.01 until 20,000 generations are reached. The number of successful estimations by RCGAs was larger than that by the symbolic-numeric method. This is because the constraints in the symbolic-numeric method might strongly affect the parameter estimation.

The histograms of the estimated values for the three parameters are shown in Fig. 2. As seen in the figures, the three parameters were correctly estimated by the symbolic-numeric method, while all of the estimations by RCGAs failed. Figure 2 (A) shows the histogram of the $k_{21}$ values. The most frequent values estimated by the symbolic-numeric method were found in the bin corresponding to the range from 0.005 <estimated $k_{21}$ $\leq$ 0.015, which included the correct value, 0.01. In addition, the estimated values were concentrated around the correct value. In contrast, although the most frequent values generated by RCGAs were also found near the correct value, the estimated values were distributed uniformly in the range from 0.01 to 0.05. A similar situation is shown in Fig. 2 (C). The most frequent values estimated by the symbolic-numeric method were found in the bin including the correct value, 10.0, while the values obtained by RCGAs were widely distributed. In Fig. 2 (B), our symbolic-numeric method correctly estimated the parameter values ($k_{31} = 0.1$), but RCGAs definitely failed. In summary, the parameter values estimated by our method were narrowly distributed in the bin including the correct values in all cases, but those generated by RCGAs were widely distributed.

## 2.5 Further Remarks

Our symbolic-numeric method correctly estimated the parameter values from the measured data of one molecule, while by RCGAs failed. This clearly indicates that the ability of our method for estimating the parameter values was far superior to that of RCGAs. Although the present study focused on a simple model, our method is a feasible approach for parameter estimation in network dynamics including unmeasured variables.

Finally, we discuss further application possibilities of our method. As indicated in 3.2, the differential elimination is not dependent on linearity, but shows its power in the nonlinear system. By the differential elimination, $x_2$ in the first nonlinear equation of eq. 10 in 3.2 can be eliminated in the first equation of eq. 11. Thus, our method has different types of differential equations in its application. Another possibility of our method is application to network inference without known structure. Since the present method is designed with the assumption of known network structure, the application range of our method to network inference is naturally restricted. However, our method can select the most possible network structure among the networks with similar structures. Indeed, we designed a similar procedure for evaluating the network structures with the measured data [8]. In our previous approach, we adopted the transformation of a system of differential equations into the equivalent system of algebraic equations by Laplace transformation. In this case, the system must be linear, due to the Laplace transformation. Furthermore, the numeric optimization in the previous approach frequently faces difficulties, due to the existence of the pole in the Laplace domain. In contrast, the above pitfalls are overcome in

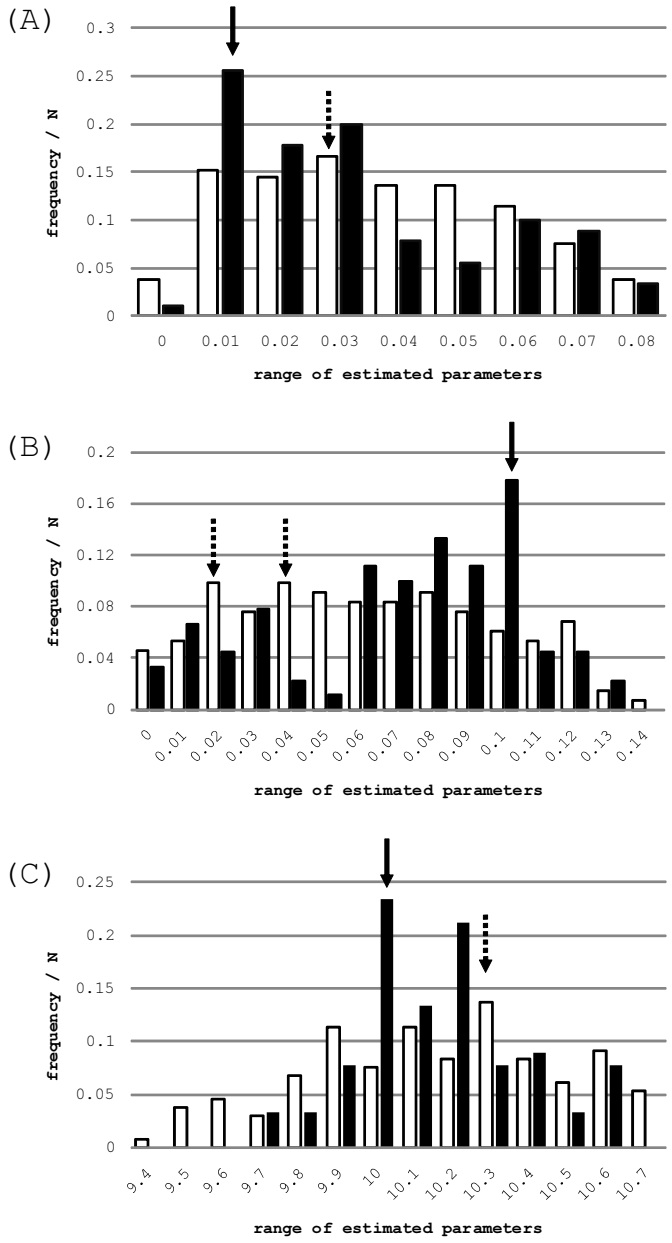


Figure 2: Histograms of estimated parameters. (A) Histogram of estimated $k_{21}$. (B) Histogram of estimated $k_{31}$. (C) Histogram of estimated $k_{41}$. In these histograms, the abscissa represents the bin. The bin range for (A) and (B) is 0.01, and that for (C) is 0.1. The ordinate shows the ratio of frequency (frequency divided by the number of estimated parameter sets ($N$). $N$ for RCGAs was 132, and that for the symbolic-numeric method was 90). Black bars represent the ratio of frequency for the symbolic-numeric method. White bars represent that for the RCGAs.

the present method, by the combination of genetic algorithm and differential elimination. In the near future, the application of the present method to various forms of networks will be presented to further investigate the merits and pitfalls of our method, in terms of parameter optimization and model selection.

# 3 Methods

## 3.1 Real-Coded Genetic Algorithms

The Genetic Algorithm is a well-known parameter optimization framework, which is inspired by the evolutionary process of biology [6, 7]. We have applied an efficient computational technique based on RCGAs as a nonlinear numerical optimization method [9], by the combination of *unimodal normal distribution crossover* (UNDX) [10] and *minimal generation gap* (MGG) [11]. The generation-alternations are repeated until either the value of the objective function $E$ becomes less than a given threshold (we called this threshold the error allowance on RCGAs) or the number of generation-alternation iterations reaches a given threshold of maximum generation counts.

The general procedure for a typical or conventional GA is as follows:

1. *Generation of Initial Population* Generate initial population randomly.
2. *Selection for Reproduction* Select individuals in population for the *Reproduction* step according to the generation-alternation model.
3. *Reproduction* Generate offspring by the crossover operator from the parent individuals selected in step 2.
4. *Selection* Evaluate the fitness of each individual and select individuals for survival (see also 2.3).
5. *Repeat the procedure between steps 2 to 4 until a certain condition for termination is satisfied.*

## 3.2 Differential Elimination

Differential algebra aims at studying differential equations from a purely algebraic point of view [12, 13]. The differential elimination theory is a sub theory of differential algebra [3], based on Rosenfeld-Gröbner [4]. The differential elimination process rewrites the inputted system of differential equations to another equivalent system according to ranking (order of terms). Here, we provide an example of the differential elimination process, as shown below, according to Boulier [3, 5].

Assume a system of parametric ordinary differential equations with two variables, $x_1$ and $x_2$, and then build the following equations,

$$\begin{aligned} \dot{x_1} &= -k_{12}x_1 + k_{21}x_2 - \frac{V_e x_1}{k_e + x_1} \\ \dot{x_2} &= k_{12}x_1 - k_{21}x_2 \end{aligned} \tag{10}$$

where $k_{12}$, $k_{21}$, $k_e$ and $V_e$ are some constants. The differential elimination, then produces the following two equations equivalent to the above system.

$$\begin{aligned} \ddot{x_1}(x_1+k_e)^2 + (k_{12}+k_{21})\dot{x_1}(x_1+k_e)^2 + V_e\dot{x_1}k_e + k_{21}V_e x_1(x_1+x_e) = 0 \\ \dot{x_1}(k_e+x_1) + k_{21}x_1^2 + (k_{12}+V_e)x_1 - k_{21}(k_e+x_1)x_2 = 0 \end{aligned} \tag{11}$$

The first equation in eq. 11 is composed of $x_1$, its derivatives, and the parameters obtained by eliminating $x_2$, and the second one is composed of $x_1$, its derivatives, the parameters, and $x_2$. Note that $x_2$ can be expressed by $x_1$, its derivatives, and the parameters in the second equation.

All of the symbolic computations for the differential elimination were performed using the *diffalg* package of MAPLE 10.

### 3.3 Combination of RCGAs and Differential Elimination

In general, the typical objective function for evaluating the reproducibility of an experimentally observed time-series for a parameter set is the total relative error, $E$, such as eq. 6. The parameter set is then estimated when the total relative error falls below a given threshold. However, the immense searching space of parameter values frequently prohibits the correct parameter estimation. To overcome this problem, we introduce the constraints derived by differential elimination into the objective function. In the present study, our strategy is to reduce the immense searching space, by introducing constraints between the estimated parameters derived from differential elimination, $C$, into the objective function for the Real-Coded Genetic Algorithm, which is a well-known heuristic numerical optimizer, i.e.,

$$\text{ObjectiveFunction} = \alpha E + (1-\alpha)C \tag{12}$$

where $\alpha$ is a weighting factor in the objective function. Here, $\alpha$ was defined such that $E$ and $C$ were equally weighted.

## References

[1] Kitano, H.: System Biology: A Brief Overview, *Science* Vol. 295, no. 5560, March (2002) 1662 - 1664.

[2] Nocedal, J., Wright, S. J.: Numerical Optimization, *Springer-Verlag, NY*, (1999).

[3] Boulier, F.: Differential Elimination and Biological Modeling, *Johann Radon Institute for Computational and Applied Mathematics (RICAM) Book Series*, Vol. 2, (2007) 111-139.

[4] Boulier, F., Lazard, D., Ollivier, F., Petitot, M.: Representation for the radical of a finitely generated differential ideal, *Proceedings of ISSAC 1995*, (1995), 158-166.

[5] Boulier, F.: Differential algebra and system modeling in cellular biology, *Proceedings of Algebraic Biology 2008*, (2008) 22-39.

[6] Holland, J. H.: Adaptation in Natural and Artificial Systems, *The University of Michigan Press, Ann Arbor, MI* (1975).

[7] Goldberg, D. D.: Genetic Algorithms in Search, Optimization and Machine Learning, *Addison-Wesley Longman Publishing Co., Inc., Boston, MA*, (1989).

[8] Yoshida, H., Nakagawa, K., Anai, H., and Horimoto, K.: An Algebraic-Numeric Algorithm for the Model Selection in Kinetic Networks, *Proceedings of the 10th CASC, LNCS* 4770, (2007) 433-447.

[9] Jonikow, C. Z. and Michalewicz, Z.: An Experimental Comparison of Binary and Floating Point Representations in Genetic Algorithms, *Proceedings of the Fourth International Conference on Genetic Algorithms*, (1991) 31-36.

[10] Ono, I. and Kobayashi, S.: A real-coded genetic algorithm for function optimization using unimodal distribution crossover, *Proc 7$^{th}$ ICGA*, (1997) 249–253.

[11] Satoh, H., Ono, I. and Kobayashi, S.: A new generation alternation model of genetic algorithm and its assessment, *J. of Japanese Society for Artificial Intelligence*, 15(2) (1997) 743-744.

[12] Ritt, J.F.: Differential Algebra, *Dover Publications Inc., NY*, (1950).

[13] Kolchin, E.R.: Differential Algebra and Algebraic Groups, *Academic Press, NY*, (1973).