厚生労働科学研究費補助金（肝炎等克服緊急対策研究事業）
分担研究報告書

分子標的創薬に向けた細胞内活性化ネットワークの推定研究

堀本　勝久　産業技術総合研究所　研究チーム長

研究要旨：昨年度開発した、予後データや検体サンプル背景などの表現型データから分子機能を推定する新規方法のプロトタイプを改良し、新規診断および治療薬開発に直接的に関連する分子signatureの特定を試みた。新規改良法は、プロトタイプでの表現型の相異に関与するパスウェイの同定に加え、それらから分子signatureを抽出することができる。本年度は、プロテオミクスデータの解析により、解析における表現型指向の性能についてテストを行った。具体的には、肺がん細胞株41種について、細胞表面、細胞溶液、細胞質の3か所のタンパク質量分布データについて表現型指向解析法を適用し、EGFR及びK-RASにおける突然変異の有無に関するサンプルの相異を表現する、それぞれ75、78のタンパク質セットを同定した。さらに、これらタンパク質セットも基づき予後データに関する生存解析を行った結果、共に有意確率0.001以下を得た。この結果は、これらタンパク質セットがEGFR及びK-RASにおける突然変異に関するサンプル間の相異を代表するだけでなく、突然変異の有無が関与する予後予測のマーカーセットとして有効であることを示している。

A. 研究目的

昨年開発した、表現型（予後データ等）の情報を積極的に活用することによる、分子標的創薬過程における薬効メカニズムの推定する方法を改良し、分子 signature を合理的に同定し、疾患に関与するターゲット分子候補の推定を試みる。

B. 研究方法

昨年度開発した解析法では、分子データを予め分子機能によって分類し（現在、887 の機能グループに分類）、それぞれの分子群についてクラスタリングを行い、さらに分子特性に基づいて分類されたサンプルについて生存解析等の予後データの解析を行い、有意な差の検出されたパスウェイを特定する。本年度は、検出されたパスウェイからさらに分子 signature を抽出することを可能にした。

（倫理面への配慮）

当機関においては、数理研究のみを実施するため、動物実験等の該当は無い。

C. 研究結果

肺がん細胞株 41 種について、細胞表面、細胞溶液、細胞質の 3 か所のタンパク質量分布データについて表現型指向解析法を適用し、EGFR 及び K-RAS における突然変異

の有無に関するサンプルの相異を表現する、それぞれ 75、78 のタンパク質セットを同定した。さらにこれら分子群について、予後データに関する生存解析を行った結果、共に有意確率 0.001 以下を得た。

D. 考察

改良方法によって探索された分子 signature を構成するタンパク質セットが EGFR 及び K-RAS における突然変異に関するサンプル間の相異を代表するだけでなく、突然変異の有無が関与する予後予測のマーカーセットとして有効であることを示している。

E. 結論

昨年度より開発を行った表現型相異指向分子機能推定法が、分子 signature を推定する有用なアプローチである可能性を示すことができた。さらに、予後データに関する良好な検証結果から、実験計測データからのターゲット候補の絞り込み方法であるのみならず、予後予測に関する有用な新規マーカーの探索も可能な方法であることが示唆された。

F. 研究発表

1. 論文発表

1) Piao, G., Saito, S., Sun, Y., Liu, Z., Wang, Y., Han, X., Wu, J., Zhou, H., Chen, L. and Horimoto, K.: A computational procedure for identifying master regulator candidates for diabetes progression in Goto-Kakizaki rat, BMC Sys. Biol., 6(Suppl1), ,S2, 2012

2) Nagamatsu, G., Kosaka, T., Saito, S., Takubo, K., Akiyama, H., Sudo, T., Horimoto, K., Oya, M. and Suda, T.: Tracing the conversion process for primordial germ cells to pluripotent stem cells. Biology of Reproduction, 86, 1–11, 2012

3) Nagamatsu, G., Saito, S., Kosaka, T., Takubo, K., Oya, M., Horimoto, K., Suda, T.: Optimal ratio of transcription factors for somatic cell reprogramming. J. Biol. Chem., 287: 36273-82, 2012.

4) Nakajima, N., Tamura, T., Yamanishi, Y., Horimoto, K., Akutsu, T.: Network Completion Using Dynamic Programming and Least-Squares Fitting. The Scientific World Journal, 2012: 957620, 2012

2. 学会発表

1) 堀本勝久：「創薬のための計算システム生物学」シンポジウム「第3回新たな創薬パラダイムの創出」、2012年1月12日、東京大学医科学研究所 1 号館講堂

2) Toshimori, A., Shimizu, K., Saito, S., Horimoto, K., Yu, T., Fan, Z., Jinrong, X., Zhou, X., Zhou, H.: Exploration of cellular relationships from characteristically expressed genes by partial canonical correlation analysis. The fifth International Workshop on Intelligent Informatics in Biology and Medicine (IIBM 2012), Palermo, Italy, July 4 - 6, 2012

3) Saito, S., Nagamatsu, G., Zhou, H., Horimoto, K.: Phenotype-Difference Oriented Analysis for Detecting Molecular Functions from Omics Data. The Fourth

Annual International Conference on Computational and Systems Biology (ICCSB2012), SHANGHAI – CHINA October 12-14, 2012

4) Saito, S., Nagamatsu, G., Zhou, H., Horimoto, K.: Outcome-Difference Oriented Analysis for Detecting Molecular Functions from Omics Data. The 1st International Conference on Translational Biomedical Informatics (ICTBI 2012), Taicang City, Jiangsu Province, China, Dec. 10, 2012

G. 知的所有権の出願・取得状況

1. 特許取得

なし

2. 実用新案登録

なし

3. その他

特になし

III. 研究成果の刊行に関する一覧表

別紙５

研究成果の刊行に関する一覧表

書籍

著者氏名	論文タイトル名	書籍全体の編集者名	書　籍　名	出版社名	出版地	出版年	ページ
Nakatsui M,（堀本）	A General Procedure for the Accurate Parameter Estimation in Dynamic Systems Using New Estimation Errors.	Horimoto, K. Nakatsui, M., Popov, N.	Algebraic and Numeric Biology	Springer	Heidelberg	2012	149-166

雑誌

発表者氏名	論文タイトル名	発表誌名	巻号	ページ	出版年
Yamashita T,（金子）	Discrete nature of EpCAM(+) and CD90(+) cancer stem cells in human hepatocellular carcinoma.	Hepatology	-	-	(in press)
Mizukoshi E,（金子）	Enhancement of tumor-associated antigen-specific T cell responses by radiofrequency ablation of hepatocellular carcinoma.	Hepatology	-	-	(in press)
Mizuno H,（金子）	Heterogeneous nuclear ribonucleoprotein A2/B1 in association with hTERT is a potential biomarker for hepatocellular carcinoma.	Liver Int	32(7)	1146-1155	2012
Mizukoshi E,（金子）	Expression of chondroitin-glucuronate C5-epimerase and cellular immune responses in patients with hepatocellular carcinoma.	Liver Int	32(10)	1516-1526	2012
Okada H,（金子）	Acyclic retinoid targets platelet-derived growth factor signaling in the prevention of hepatic fibrosis and hepatocellular carcinoma development.	Cancer Res	72(17)	4459-4471	2012
Kitao A,（金子）	Hypervascular hepatocellular carcinoma: correlation between biologic features and signal intensity on gadoxetic acid-enhanced MR images.	Radiology	265(3)	780-789	2012
Oishi N,（金子）	Transcriptomic profiling reveals hepatic stem-like gene signatures and interplay of miR-200c and epithelial-mesenchymal transition in intrahepatic cholangiocarcinoma.	Hepatology	56(5)	1792-1803	2012
Shugo H,（金子）	Nucleostemin in injury-induced liver regeneration.	Stem Cells Dev	21(16)	3044-3054	2012

Fumihiko Y, (小原)	Synthetic lipophilic antioxidant BO-653 suppresses HCV replication.	J Med Virol	-	-	(in press)
Makoto S, (小原)	Hepatitis C virus induces overexpression of 3β-hydroxysterol Δ24-reductase through Sp1.	J Med Virol	84	733-746	2012
Yuichi H, (小原)	Self-enhancement of Hepatitis C Virus Replication by Promotion of Specific Sphingolipid Biosynthesis.	PLoS Pathog	8(8)	e1002860	2012
Kazuaki I, (小原)	Impairment of interferon regulatory factor-1 3 activation by hepatitis C virus core protein basic region 1.	Biochem Biophys Res Commun	428(4)	494-499	2012
Satoshi S, (小原)	Immunization with a recombinant vaccinia virus that encodes nonstructural proteins of the hepatitis C virus suppresses viral protein levels in mouse liver.	PLoS One	7(12)	e51656	2012
C. J. Hipolito, (菅)	Ribosomal production and in vitro selection of natural product-like peptidomimetics: The FIT and RaPID systems.	Current Opinion in Chemical Biology	16	196-203	2012
Y. Hayashi, (菅)	In vitro selection of anti-Akt2 thioether-macrocyclic peptides leading to isoform-selective inhibitors.	ACS Chemical Biology	7	607-613	2012
K. Iwasaki, (菅)	Selective thioether macrocyclization of peptides having the N-terminal 2-chloroacetyl group and competing two or three cysteine residues in translation.	Organic & Biomolecular Chemistry	10	5783-5786	2012
Murakami Y, (深澤)	Selective estrogen receptor modulators inhibit hepatitis C virus infection at multiple steps of the virus lifecycle.	Microbes and Infection	15	45-55	2013
Shimizu S, (竹原)	Inhibition of autophagy potentiates the antitumor effect of the multikinase inhibitor sorafenib in hepatocellular carcinoma.	Int J Cancer	131	548-557	2012
Hikita H, (竹原)	Bak deficiency inhibits liver carcinogenesis: a causal link between apoptosis and carcinogenesis.	J Hepatol	57	92-100	2012
Imakiire K, (宇都)	Difference in serum complement component C4a levels between hepatitis C virus carriers with persistently normal alanine aminotransferase levels or chronic hepatitis C.	Mol Med Report	6 (2)	259-264	2012

Uto H, （宇都）	Clinical features of hepatitis C virus carriers with persistently normal alanine aminotransferase levels.	Hepat Mon	12 (2)	77-84	2012
Komase K, （前川）	Serum RANTES level influences the response to pegylated interferon and ribavirin therapy in chronic hepatitis C.	Hepatol Res	-	-	(in press)
Sueki R, （前川）	Correlation between pretreatment viral sequences and the emergence of lamivudine resistance in hepatitis B virus infection.	J Med Virol	84(9)	1360-1368	2012
Shindo H, （前川）	IL-28B (IFN-λ3) and IFN-α synergistically inhibit HCV replication.	J Viral Hepat	AUG 7	AUG 7	2012
Maekawa S, （前川）	Comprehensive analysis for viral elements and interleukin-28B polymorphisms in response to pegylated interferon plus ribavirin therapy in hepatitis C virus 1B infection.	Hepatology	56(5)	1611-1621	2012
Sakamoto T, （村上）	Mechanism of the Dependence of Hepatitis B Virus Genotype G on Co-infection with Other Genotypes for Viral Replication.	J Viral Hepat	-	-	(in press)
Watanabe T, （村上）	Hepatitis C virus kinetics by administration of pegylated interferon-α in human and chimeric mice carrying human hepatocytes with variants of the IL28B gene.	Gut	-	-	(in press)
Ragheb M, （村上）	Multiple intra-familial transmission patterns of hepatitis B virus genotype D in north-eastern Egypt.	J Med Virol	84(4)	587-595	2012
Piao G, （堀本）	A computational procedure for identifying master regulator candidates for diabetes progression in Goto-Kakizaki rat.	BMC Sys Biol	6(Suppl1)	S2	2012
Nagamatsu G, （堀本）	Tracing the conversion process for primordial germ cells to pluripotent stem cells.	Biology of Reproduction	86	1-11	2012
Nagamatsu G, （堀本）	Optimal ratio of transcription factors for somatic cell reprogramming.	J Biol Chem	287	36273-36282	2012
Nakajima N, （堀本）	Network Completion Using Dynamic Programming and Least-Squares Fitting.	The Scientific World Journal	2012	957620	2012

Ⅳ．研究成果の刊行物・別刷

A General Procedure for Accurate Parameter Estimation in Dynamic Systems Using New Estimation Errors

Masahiko Nakatsui
AIST
Tokyo, Japan
m.nakatsui@aist.go.jp

Alexandre Sedoglavic
University Lille I. LIFL
Lille, France
alexandre.sedoglavic@lifl.fr

François Lemaire
University Lille I. LIFL
Lille, France
Francois.Lemaire@lifl.fr

François Boulier
University Lille I. LIFL
Lille, France
Francois.Boulier@lifl.fr

Asli Ürgüplü
University Lille I. LIFL
Lille, France
Asli.Urguplu@lifl.fr

Katsuhisa Horimoto
AIST
Tokyo, Japan
k.horimoto@aist.go.jp

Abstract

The investigation of network dynamics is a major issue in systems and synthetic biology. One of the essential steps in a dynamics investigation is the parameter estimation in the model that expresses biological phenomena. Indeed, various techniques for parameter optimization have been devised and implemented in both free and commercial software. While the computational time for parameter estimation has been greatly reduced, due to improvements in calculation algorithms and the advent of high performance computers, the accuracy of parameter estimation has not been addressed.

We previously proposed an approach for accurate parameter optimization by using Differential Elimination, which is an algebraic approach for rewriting a system of differential equations into another equivalent system. The equivalent system has the same solution as the original system, and it includes high-order derivatives, which contain information about the form of the observed time-series data. The introduction of an equivalent system into the numerical parameter optimizing procedure resulted in the drastic improvement of the estimation accuracy, since our approach evaluates the difference of not only the values but also the forms between the measured and estimated data, while the classical numerical approach evaluates only the value difference. In this report, we describe the detailed procedure of our approach for accurate parameter estimation in dynamic systems. The ability of our approach is illustrated in terms of the parameter estimation accuracy, in comparison with classical methods.

1 Introduction

The investigation of network dynamics is a major issue in systems and synthetic biology[1]. In general, a network model for describing the kinetics of constituent molecules is 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. While the computational time for parameter estimation has been greatly reduced, due to the improvements in calculation algorithms and the advent of high performance computers, the accurate numerical estimation of parameter values for a given model remains a limiting step. Indeed, the parameter values estimated by various optimization techniques are frequently quite variable, due to the conditions for parameter estimation, such as the initial values. In particular, we cannot always obtain the data measured for all of the constituent molecules, due to limitations of measurement techniques and ethical constraints. In this case, 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 quite difficult to estimate the parameters in such a network model including unmeasured variables.

Differential elimination was applied[3] to improve the parameter estimation methods, especially in the model dynamics including unmonitored variables. The idea consisted of computing differential equations from the input system, from which the unmonitored variables were eliminated. These differential equations could then be used to guess the initial values for the Newton-type numerical parameter optimization scheme. The overall method was implemented over the BLAD libraries[4]. Differential elimination theory is a branch of the differential algebra of Ritt and Kolchin[5, 6]. Its basis was developed by Ritt, who founded the theory of characteristic sets. Ritt's ideas were subsequently developed by Seidenberg [7], Wu[8], Boulier et al.[9, 10] and many other researchers. The Rosenfeld-Gröbner algorithm[9, 10] is the first complete algorithm for differential elimination ever implemented. It relies on Ritt and Seidenberg's ideas, on the Rosenfeld Lemma, which reduces differential problems to non-differential polynomial ones, and on the Gröbner bases theory for solving non-differential polynomial systems (although recent implementations completely avoid Gröbner bases computations). The Rosenfeld-Gröbner algorithm was implemented in 1996 in the diffalg package of the MAPLE computer algebra software. Starting from MAPLE 14, it should be replaced by the MAPLE DifferentialAlgebra package, which relies on the BLAD libraries[11].

Recently, we proposed a new procedure for optimizing the parameters, by using differential elimination. Our procedure partially utilizes a technique from a previous study[12, 13], regarding the introduction of differential elimination into parameter optimization in a network. Instead of using differential elimination for estimating the initial values for the following parameter optimization, the equations derived by differential elimination are directly introduced as the constraints into the objective function for the parameter optimization[14, 15, 16, 17]. Here, we will describe the detailed procedure of our approach, by using a simple model represented as non-linear differential equations. We also discuss the merits and pitfalls of our procedure, in terms of its extension to more realistic and complex models.

2 Procedure

2.1 Overview of Present Procedure

The key point of this study is the introduction of new constraints obtained by differential elimination into the objective function, to improve the parameter accuracy. This section outlines our new procedure for estimating the parameters, using constraints built from differential elimination, and compared it with the classical constraints based on the total relative error. For clarity, the method is described using an academic example.

We first present the example. We then show how to build our new constraints using differential elimination, and how to optimize the evaluation of those new constraints over numeric values. Subsequently, we present our genetic algorithm for estimating the parameter values, and finish with the results. All Maple commands used for computing the expressions described in the following subsections are provided in appendix A.

2.2 Example

Differential algebra aims at studying differential equations from a purely algebraic point of view[5, 6]. Differential elimination theory is a sub theory of differential algebra, based on Rosenfeld-Gröbner[9]. Differential elimination rewrites the inputted system of differential equations to another equivalent system, according to (order of terms). Here, we provide an example of differential elimination, as shown below, according to Boulier[12].

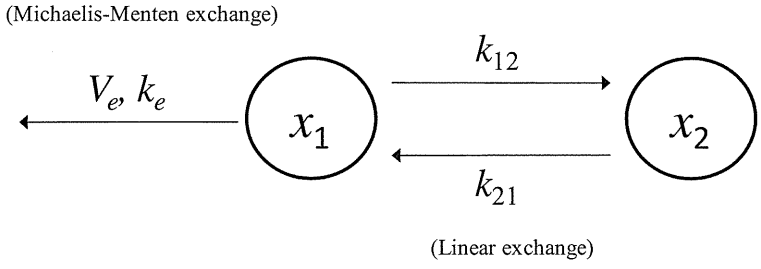


Figure 1: Schematic representation of the model

The model is composed of two state variables, x_1 and x_2. We assumed that the time-series data for one of the variable, x_1, are obtained.

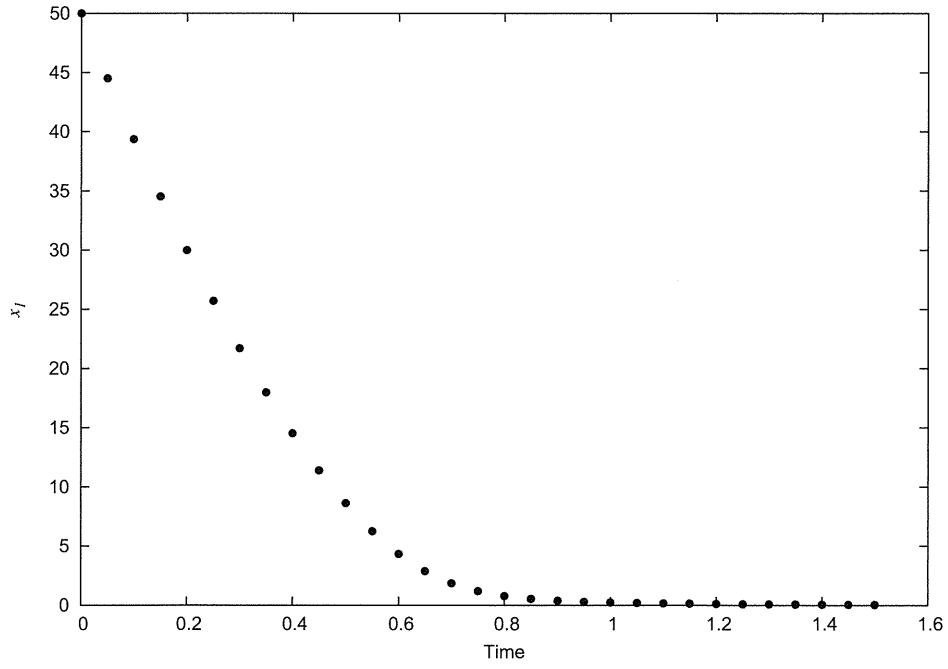


Figure 2: Reference curve

According to the kinetics of the model (Eqn. (1)), a reference curve of one variable, x_1, was generated for $0 \leq t \leq 1.5$ with intervals of 0.05, under the following conditions: $x_1(0) = 50.0, x_2(0) = 0.0, Ve = 101.0, k_{12} = 0.5, k_{21} = 3.0$ and $k_e = 7.0$.

Assume a model of two variables, x_1 and x_2, as schematically depicted in Fig. 1, with the corresponding system of differential equations expressed as follows:

$$\begin{cases} \dot{x_1} = -k_{12}x_1 + k_{21}x_2 - \frac{V_e x_1}{k_e + x_1} \\ \dot{x_2} = k_{12}x_1 - k_{21}x_2 \end{cases} \tag{1}$$

where k_{12}, k_{21}, k_e and V_e are some constants. Two molecules are assumed to bind according to Michaelis-Menten kinetics.

Here we assume that the time-series of only one variable, x_1, can be observed. x_2 is assumed to be non-observed; however, we assumed that $x_2(0) = 0$ was known. According to the model in Fig. 1, a reference curve of one variable, x_1, was generated in Fig. 2. Among the parameters in the model, the values of three parameters, k_{12}, k_{21}, and V_e, were estimated, and the values of the remaining parameters were set to the same values as those used in the generation of the reference curve of Fig. 2.

2.3 Differential Elimination

The differential elimination then produces the following two equations equivalent to the above system.

$$\begin{cases} \dot{x_1}(k_{21}+x_1)+k_{21}x_1^2+(k_{12}+V_e)x_1-k_{21}(k_e+x_1)x_2=0 \\ \ddot{x_1}(x_1+k_e)^2+(k_{12}+k_{21})\dot{x_1}(x_1+k_e)^2+V_e\dot{x_1}k_e+k_{21}V_ex_1(x_1+k_e)=0 \end{cases} \tag{2}$$

As a consequence, the latter two equations should be zero for any solution of (1). The latter to equations, respectfully, called $C_{1,t}$ and $C_{2,t}$ in the following, will be used to define our error estimation, based on the evaluation of $|C_{1,t}|+|C_{2,t}|$.

System (2) can be computed in Maple 14, using the following commands:

```
> with(DifferentialAlgebra):
> sys := [
>     x1[t] - ( -k12*x1 + k21*x2 - Ve*x1/(ke+x1)),
>     x2[t] - ( k12*x1 - k21*x2)
>        ];
                                          Ve x1
      sys := [x1[t] + k12 x1 - k21 x2 + -------, x2[t] - k12 x1 + k21 x2]
                                         ke + x1

> R := DifferentialRing(blocks=[x2,x1,k12(),k21(),Ve(),ke()], derivations=[t]);
                              R := differential_ring

> Ids := RosenfeldGroebner( numer(sys), denom(sys), R,
>                           basefield=field(generators=[k12,k21,Ve,ke]));
                      Ids := [regular_differential_chain]

> eqs := Equations(Ids[1]);
eqs := [
                                                          2
    k21 x2 x1 + k21 x2 ke - x1[t] x1 - x1[t] ke - k12 x1  - k12 x1 ke - Ve x1,

              2                                    2           2
    x1[t, t] x1  + 2 x1[t, t] x1 ke + x1[t, t] ke  + x1[t] x1  k12

              2                                                          2
     + x1[t] x1  k21 + 2 x1[t] x1 k12 ke + 2 x1[t] x1 k21 ke + x1[t] k12 ke

                   2                    2
     + x1[t] k21 ke  + x1[t] Ve ke + x1  k21 Ve + x1 k21 Ve ke]
```

2.4 Simplification

In general, the problem of reducing the evaluation complexity (additions, multiplications) is difficult and requires a large number of computer operations (a.k.a. a high algorithmic complexity). Moreover, the evaluation complexity of the Rosenfeld-Gröbner output tends to be exponential in the evaluation complexity of the input, especially when using elimination rankings, as in this case. Consequently, before directly applying techniques such as factorization, Horner schemes, common sub expression detection, etc. for reducing the evaluation complexity, we try to use the knowledge we already have on the initial ODE system.

We now describe a preprocessing step that facilitates the evaluation of $\bar{C}_{DE} = |C_{1,t}| + |C_{2,t}|$.

The expressions of $C_{1,t}$ and $C_{2,t}$ given in (2) are not the expressions originally computed by the Rosenfeld-Gröbner algorithm. Indeed, the Rosenfeld-Gröbner algorithm outputs expanded expressions.

Thus, using the Rosenfeld-Gröbner outputs, one has to evaluate the following expression, $\bar{C}_{DE}$:

$$\begin{aligned}\bar{C}_{DE} &= |-k_{21}x_2k_e - k_{21}x_2x_1 + \dot{x}_1k_e + \dot{x}_1x_1 + k_{12}x_1k_e + k_{12}x_1^2 + V_ex_1| \\ &\quad + |k_{21}k_eV_ex_1 + 2k_ek_{12}x_1\dot{x}_1 + 2k_{21}k_e\dot{x}_1x_1 + 2k_e\ddot{x}_1x_1 + k_{12}\dot{x}_1k_e^2 \\ &\quad + k_eV_e\dot{x}_1 + k_{12}x_1^2\dot{x}_1 + k_{21}k_e^2\dot{x}_1 + k_{21}x_1^2\dot{x}_1 + k_{21}x_1^2V_e + \ddot{x}_1k_e^2 + \ddot{x}_1x_1^2|\end{aligned} \tag{3}$$

requiring 18 additions + 46 multiplications (+2 function evaluations for the absolute value). These operations represent the evaluation complexity of the expression $\bar{C}_{DE}$.

Since the expressions of $C_{1,t}$ and $C_{2,t}$ were computed from an ODE system involving the denominator $k_e + x_1$, from a Michaelis-Menten factor, the expression $k_e + x_1$ can be likely be factorized. By introducing a new variable, $d_e = k_e + x_1$, and applying the substitution $k_e \rightarrow d_e - x_1$ in the previous expression of $\bar{C}_{DE}$, one gets

$$\begin{aligned}\bar{C}_{DE} &= |-k_{21}x_2d_e + \dot{x}_1d_e + k_{12}x_1d_e + V_ex_1| \\ &\quad + |k_{21}V_ex_1d_e + k_{12}\dot{x}_1d_e^2 + V_e\dot{x}_1d_e - V_e\dot{x}_1x_1 + k_{21}\dot{x}_1d_e^2 + \ddot{x}_1d_e^2|\end{aligned} \tag{4}$$

requiring 9 additions + 21 multiplications.

Please note that the last expression of $\bar{C}_{DE}$ does not involve k_e anymore, which shows that the variable k_e only appears in $\bar{C}_{DE}$ in the term $k_e + x_1$.

This trick with the denominators has divided the number of operations by 2. On more complex systems, the benefit can be much greater. It is worth noting that the trick works quite similarly if several denominators are involved and if each denominator linearly involves a parameter that is not involved in the other denominators. More precisely, if one has n denominators of the form $k_i + f_i$, and if k_i is not involved in any f_i, then one performs n substitutions $k_i \rightarrow f_i - d_i$.

Further computations using a Horner scheme can now be accomplished. For example, applying a recursive Horner scheme with decreasing priority on the variables $d_e, x_1, x_2, \dot{x}_1, \ddot{x}_1$ yields:

$$\begin{aligned}\bar{C}_{DE} &= |V_ex_1 - (k_{21}x_2 - \dot{x}_1 - k_{12}x_1)d_e| \\ &\quad + |-V_e\dot{x}_1x_1 + (k_{21}V_ex_1 + V_e\dot{x}_1 + (\ddot{x}_1 + (k_{12} + k_{21})\dot{x}_1)d_e)d_e|\end{aligned} \tag{5}$$

requiring 9 additions + 12 multiplications.

To finish, further simplification can be achieved using the optimize command of the optimize package in the Computer Algebra software Maple. This last command tries to recognize common expressions in

order to compute common subexpressions only once. This command is not very costly, since it is based on easy heuristics. In our case, it yields the sequence of commands:

$$\begin{aligned} t7 &= |V_e x_1 - (k_{21}x_2 - \dot{x_1} - k_{12}x_1)d_e|, \\ t8 &= V_e \dot{x_1}, \\ t19 &= |-t8x_1 + (k_{21}V_e x_1 + t8 + (\ddot{x_1} + (k_{12}+k_{21})\dot{x_1})d_e)d_e|, \\ C_{DE} &= t7 + t19 \end{aligned} \tag{6}$$

requring 9 additions + 11 multiplications + 4 assignments. Note that the last gain here is only 1 multiplication, but can be higher on larger systems.

All previous operations can be automated in Maple (see appendix A for the complete set of Maple commands); the *C* command of the optimize package yields the C code as

```
t7 = fabs(Ve*x1-(k21*x2-x1t-k12*x1)*de);
t8 = Ve*x1t;
t19 = fabs(-t8*x1+(k21*Ve*x1+t8+(x1tt+(k12+k21)*x1t)*de)*de);
E = t7+t19;.
```

2.5 Introduction of Constraints

The objective function for parameter optimization in this study is composed of two terms: one is the standard error function between the estimated and monitored data, and the other is the constraints obtained by differential elimination. The error function is defined as follows: Suppose $x^c_{i,t}$ is the time-series data at time t of x_i, simulated by using the estimated parameter values and the model equations by integration, and $x^m_{i,t}$ represents the monitored data at time t. The sum of the absolute values of the relative error between $x^c_{i,t}$ and $x^m_{i,t}$ gives the averaged relative error over the numbers of monitored variables and time points, E, as a standard error function, i.e.,

$$E = \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}\left|\frac{x^c_{i,t} - x^m_{i,t}}{x^m_{i,t}}\right| \tag{7}$$

where N and T are the numbers of monitored variables and time points, respectively.

Next we define the DE constraints obtained by the differential elimination and simplification procedure. The simplified equivalent system (Eqn. (6)) is composed of x_1, its derivatives ($\dot{x_1}$ and $\ddot{x_1}$), x_2, and the parameters (k_{12}, k_{21}, V_e and k_e). Note that x_2 in Eqn. (6) can be estimated by x_1, the parameters, and $x_2(0)$. The derivatives of variable x_1 can be estimated numerically by the following procedure. First, we obtain two equations by a Taylor expansion of $x_1(t)$,

$$x_1(t+h) = x_1(t) + hx_1'(t) + \frac{h^2}{2}x_1''(t) + \frac{h^3}{6}x_1'''(t) + \cdots, \tag{8}$$

$$x_1(t-h) = x_1(t) - hx_1'(t) + \frac{h^2}{2}x_1''(t) - \frac{h^3}{6}x_1'''(t) + \cdots. \tag{9}$$

Second, we subtract Eqn. (9) from (8),

$$\begin{aligned} x_1(t+h) - x_1(t-h) &= 2hx_1'(t) + \frac{1}{3}h^3x_1'''(t) + \cdots, \\ 2hx_1'(t) &= x_1(t+h) - x_1(t-h) - \frac{1}{3}h^3x_1'''(t) + \cdots, \\ x_1'(t) &= \frac{x_1(t+h) - x_1(t-h)}{2h} - \frac{h^2}{6}x_1'''(t) + \cdots. \end{aligned} \tag{10}$$

Finally, we obtain following approximation, under the assumption of $0 < h < 1$,

$$x_1'(t) = \frac{x_1(t+h) - x_1(t-h)}{2h} + O(h^2). \tag{11}$$

We are able to obtain higher-order derivatives from lower-order derivatives in same way, as mentioned above. For instance, we can estimate second order derivatives of x_1 by using following equation,

$$x_1''(t) = \frac{x_1'(t+h) - x_1'(t-h)}{2h} + O(h^2). \tag{12}$$

The value of the simplified equivalent system (Eqn. (6)) can be calculated by the substitution of the observed x_1, its numerically the estimated derivatives, estimated x_2, and the parameter values estimated by the numerical parameter optimizing procedure. In general, Differential Elimination rewrites the original system of differential equations into an equivalent system, which means both systems have the same solutions. This clearly shows that the evaluated values of the equivalent system will be zero with exactly estimated parameter sets, time-series data without noise, and derivatives. Thus, the equivalent system can be regarded as a kind of objective function that expresses the difference between the monitored and estimated data. In this study, we express DE Constraint (C_{DE}), as the average of the linear combination of the equation in the equivalent system over the number of equations and time points, as follows:

$$C_{DE} = \frac{1}{LT} \sum_{l=1}^{L} \sum_{t=1}^{T} |C_{l,t}| \tag{13}$$

where L and T are the numbers of equivalent equations and time points, respectively. Finally, we introduce C_{DE} $\bar{C}_{DE}$, which is simplified as C_{DE}, into the objective function, F, in combination with E, as:

$$F = \alpha E + (1-\alpha) C_{DE} \tag{14}$$

where $\alpha (0 \leq \alpha \leq 1)$ is the weight of the two functions. As a result, our computational task is to find a set of parameter values that minimize F. When we apply the simplification procedure (see 2.4), then $\frac{1}{LT}\bar{C}_{DE}$ is used instead of C_{DE}.

The weighting factor α in the objective function F is estimated from the slope of the Pareto-optimal solutions. First, we obtained some parameter sets (in the case study, we obtained 200 kinds of parameter sets) by the compute_parameter_set function, under the conditions of $\delta = 1.0$ and the tentative value of α $ta = 1$ (this means we used the classical objective function, i.e. $F = E$). Second, we selected the Pareto-optimal solutions from the list of estimated parameter sets, by the select_pareto_optimal_solutions function. By fitting the linear function $C = aE + b$ to the selected the Pareto-optimal solutions, we obtained the slope of Pareto-optimal solutions, a. Finally, we estimated the value of α from the slope a. The detailed algorithms for estimating the value of α are shown in Algorithm 1 and 2. Fig. 3 represents a part of the estimated parameter sets in the case study (the detailed algorithms for the parameter optimization we used for the case study are shown in 2.6), the Pareto-optimal solutions, and the fitted line for the Pareto-optimal solutions. We obtained the slope $a = 20.7653$ for the case study, and the value of α was estimated as $\alpha = 0.95406$.

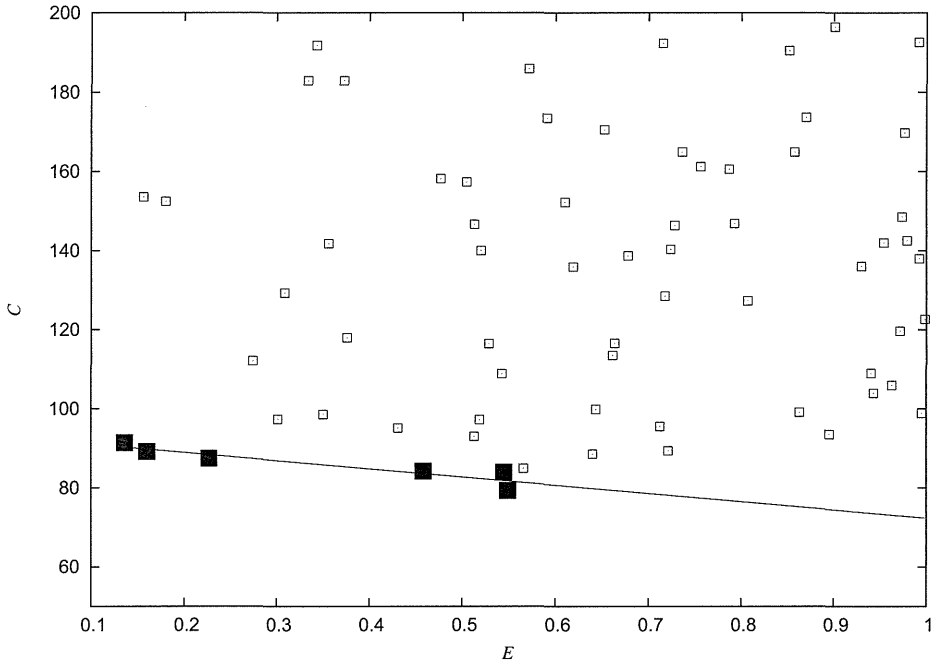


Figure 3: Slope of Pareto-optimal solutions

The empty squares (□) indicate the set of evaluated values, E and C_{DE}. The filled squares (■) show the Pareto-optimal solutions, and the line represents the fitted line for Pareto-optimal solutions.

Algorithm 1 Estimate value of weighting factor α

Function : estimate_alpha(δ, n, ta)
Input : error tolerance δ, number of trials n, and tentative value of α ta
Return : estimated value of weighting factor α

1: $RES \leftarrow$ compute_parameter_set(ta, δ)
2: $P \leftarrow$ select_pareto_optimal_solutions(RES)
3: $EV \leftarrow \phi$
4: $CV \leftarrow \phi$
5: n size of P
6: **for** $i = 0$ to n **do**
7: $EV \leftarrow EV$ union $E(R_i)$
8: $CV \leftarrow CV$ union $C(R_i)$
9: **end for**
10: fit $CV_i = -aEV_i + b$ from EV and CV by using least square method
11: **return** $a/(a+1)$

Algorithm 2 Select Pareto-optimal solutions

Function : select_pareto_optimal_solutions(R)
Input : R set of estimated parameters
Return : Pareto-optimal solutions (P)

```
1: P ← ϕ
2: EV ← ϕ
3: CV ← ϕ
4: n size of R
5: for i = 0 to n do
6:    EV ← EV union E(R_i)
7:    CV ← CV union C(R_i)
8: end for
9: for i = 0 to n do
10:   Flag lp = true
11:   for j = 0 to n do
12:      if !(EV_i ≤ EV_j and CV_i ≤ CV_j) then
13:         lp ← false
14:      end if
15:   end for
16:   if lp then
17:      P ← P union R_i
18:   end if
19: end for
20: return P
```

2.6 Optimization Algorithm

Our approach can be applied to many kinds of parameter optimizing procedures, such as the Gradient-based method and the evolutionary optimizing method, including the Modified Powell method[18, 19], Genetic Algorithms[20, 21], and Particle Swarm Optimization[22, 23], by modifying the objective function (cost function) only[16].

Here, we applied our approach to Real-coded Genetic Algorithms[24, 25, 26], to demonstrate its ability. The detailed algorithms used to analyze the case study (Fig. 1 and 2) are shown in Algorithm 3 to 5.

Let us explain the differences between our procedure and the classical constraint E. First of all, by using $\alpha = 1$, one obtains a classical genetic algorithm using the relative error E, since we have $F = E$ when $\alpha = 1$. Second, when using $\alpha < 1$, each parameter set k returned by the compute_parameter_sets satisfies $E(k) \leq \delta$, as in the classical procedure. However, the manner in which the population evolves (in the compute_next_generation) depends on the function F. To summarize, the objective function F is only used to direct the evolution of the population, by not using the objective function F to select the final candidates, and thus it makes sense to compare the parameter sets computed in the classical procedure and in our procedure.

2.7 Results

To evaluate the ability of our procedure, we performed a simulation study by using the objective function with and without the newly developed DE constraints, by estimating the kinetic parameters in Eqn. (1).

Algorithm 3 Modify the parameter set K by computing the next generation

Function compute_next_generation(α, K)
Input : the weighting factor α, a parameter set K

1: n size of K
2: denote $K = \{k_1, \ldots, k_n\}$
3: compute $1 \leq s \leq n$ such that k_s is the one best element according to the F function (i.e. $F(k_s)$ is the minimum of $F(k_1), \ldots, F(K_n)$)
4: pick a random number r such that $1 \leq r \leq n$, and r is different from s
5: mix k_s and k_r and compute a new set $k' = \{K'_1, \ldots, k'_n\}$
6: $K' \leftarrow K'$ union $\{k_s\}$
7: modify k by replacing k_s and k_r by the two best elements of K' according to the F function

Algorithm 4 Optimization process

Function : compute_one_parameter_set(α, δ, *pop*, *gen*)
Input : the weighting factor α, the error tolerance *delta* for function F, the population size of GA *pop*, the maximum generation counts *gen*
Return : a set containing zero or one parameter set

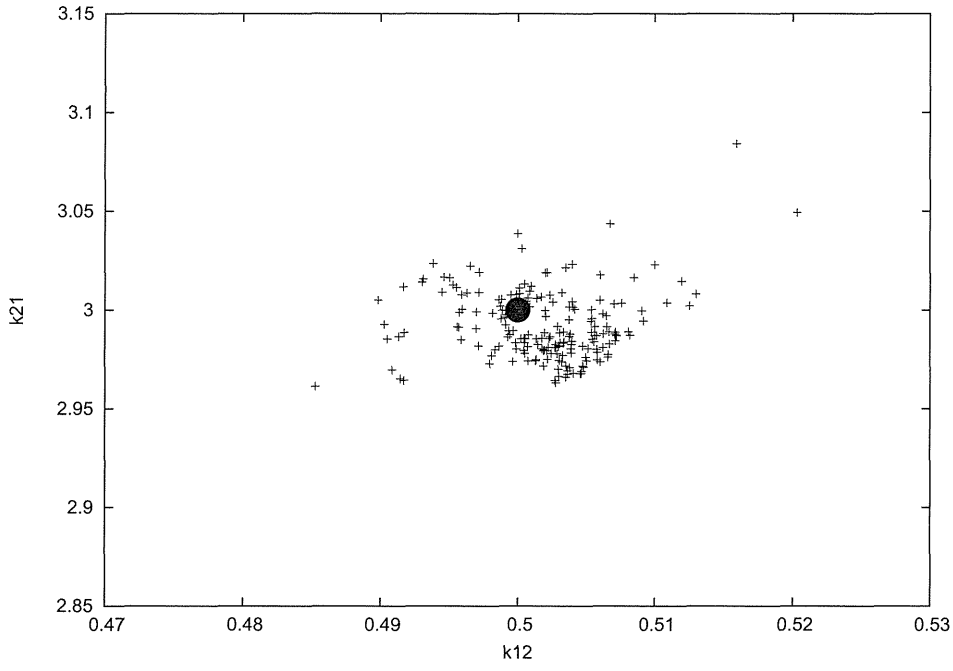
1: create a set K containing *pop* random parameter sets
2: **for** $i = 1$ to *gen* **do**
3: compute_next_generation(*alpha*, K)
4: **if** an element k in K satisfies $E(k) \leq \delta$ **then**
5: **return** k
6: **end if**
7: **end for**
8: **return** ϕ

Algorithm 5 generate a list of estimated parameter sets

Function : compute_parameter_sets(α, δ, *pop*, *gen*, *trials*)
Input : the weighting factor α, the error tolerance δ for function F, the population size of GA *pop*, the maximum generation counts *gen*, the trial number *trials*
Return : a list of parameter sets

1: $RES \leftarrow \phi$
2: **for** $i = 1$ to *trials* **do**
3: $RES \leftarrow RES$ union compute_one_parameter_set(*alpha*, *delta*, *pop*, *gen*)
4: **end for**
5: **return** RES

(A)



(B)

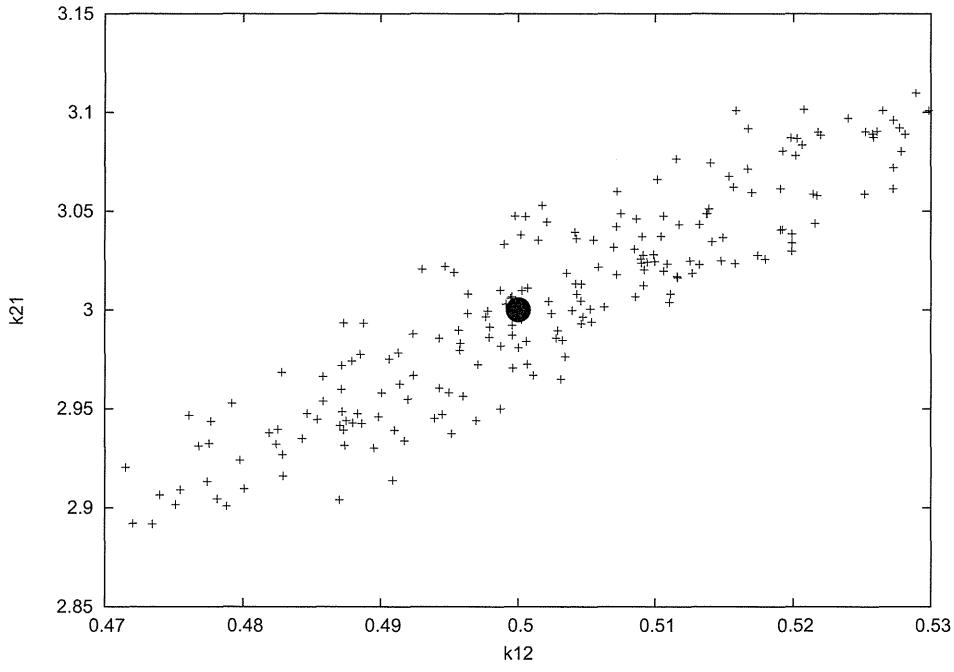


Figure 4: Comparison of parameter value clouds estimated by the classical or our proposed procedure, (A) with and (B) without DE constraints

The given values are as follows: $x_2(0) = 0.0$ and $k_e = 7.0$. The black circles indicate the correct parameter set.

Here, we assume that the time-series of only one variable, x_1, can be observed. According to the model, the reference curve of one variable, x_1, was generated in Fig. 2. Among the parameters in the model, the values of three parameters, k_{12}, k_{21}, and V_e, were estimated, and the values of the remaining parameters were set to the same values as those used in the generation of the reference curve.

The introduction of DE constraints into the objective function was quite effective, in the comparison with the distributions of the parameter values estimated with and without DE constraints (see Fig. 4). Indeed, the distribution of the estimated k_{12} and k_{21} values was highly concentrated around the correct values by the estimation with the introduction (Fig. 4 (A)), while the estimated parameters were widely distributed by the estimation without the introduction of DE constraints (Fig. 4 (B)).

3 Discussion

The accuracy of parameter estimation was clearly improved by the introduction of DE constraints into the objective function of the numerical parameter optimizing method. Indeed, the parameter value sets estimated with the introduction of DE constraints into the objective function were sharply distributed near the correct values, in contrast to the wide distribution without the introduction. In general, the derivatives included the information on the curve form of the observed time-series data, such as slope, extremal point and inflection point. This indicates that the new objective function we proposed estimates the difference of not only the values but also the forms between the measured and estimated data, while the classical objective function estimates only the value difference. Note that the DE constraint is rationally reduced from the original system of differential equations for a given model, in a mathematical sense. Thus, our approach is expected to become a general approach for parameter optimization to improve the parameter accuracy.

As expected, the new objective function requires more computational time, in comparison with an objective function with only a standard error function, due to the increase of the function in the DE constraints. In equivalent systems derived by Differential Elimination, the number of terms and operators frequently increases, and this may make the application of our procedure to a complex or large system difficult, without simplification of the equivalent system. To overcome the difficulty in the complex system, we applied simplification by symbolic computation (see 2.4). For instance, we could estimate the kinetic parameters in the negative feed-back oscillator model[27, 28, 29] by using the simplification procedure[17], while the estimation without the simplification failed, due to the immense computational time.

Another possible way to overcome the difficulty in complex models is to approximate the DE constraint. In the DE constraint, the terms with a higher order of derivatives in the differential equations generally appeared in the equivalent system. The magnitudes of the estimated values of the higher order derivatives were relatively smaller than those of the lower order derivatives. Although our procedure was useful, even for noisy data in a simple model[15], the estimated values of the higher order derivatives for noisy data may become large in this case. However, some techniques are frequently used for smoothing noisy data, and after smoothing, the values of the higher order derivatives may be smaller. If the terms with higher order derivatives can be neglected in the estimation, then the computational time may be reduced. Further studies to improve the computational time by approximation of the DE constraint will be reported in the near future.

A Implementation of Simplification

The following commands use the new DifferentialAlgebra package, and thus require Maple 14 to work.