Using an Improved Differential Evolution Algorithm for Parameter Estimation to Simulate Glycolysis Pathway

Chuii Khim Chong, Mohd Saberi Mohamad, Safaai Deris, Shahir Shamsir, Afnizanfaizal Abdullah, Yee Wen Choon, Lian En Chai, and Sigeru Omatu

Abstract. This paper presents an improved Differential Evolution algorithm (IDE). It is aimed at improving its performance in estimating the relevant parameters for metabolic pathway data to simulate glycolysis pathway for yeast. Metabolic pathway data are expected to be of significant help in the development of efficient tools in kinetic modeling and parameter estimation platforms. Nonetheless, due to the noisy data and difficulty of the system in estimating myriad of parameters, many computation algorithms face obstacles and require longer computational time to estimate the relevant parameters. The IDE proposed in this paper is a hybrid of a Differential Evolution algorithm (DE) and a Kalman Filter (KF). The outcome of IDE is proven to be superior than a Genetic Algorithm (GA) and DE. The results of IDE from this experiment show estimated optimal kinetic parameters values, shorter computation time and better accuracy of simulated results compared to the other estimation algorithms.

Keywords: Parameter Estimation, Differential Evolution Algorithm, Kalman Filter, Simulation.

Chuii Khim Chong · Mohd Saberi Mohamad · Safaai Deris · Afnizanfaizal Abdullah · Yee Wen Choon · Lian En Chai
Artificial Intelligence and Bioinformatics Research Group, Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
e-mail: ckchong2@live.utm.my, saberi@utm.my, safaai@utm.my, afnizanfaizal@utm.my, ywchoon2@live.utm.my, lechai2@live.utm.my

Shahir Shamsir
Department of Biological Sciences, Faculty of Biosciences and Bioengineering, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia
e-mail: shahir@fbb.utm.my

Sigeru Omatu
Department of Electronics, Information and Communication Engineering, Osaka Institute of Technology, Osaka 535-8585, Japan
e-mail: omatu@rsh.oit.ac.jp

S. Omatu et al. (Eds.): Distributed Computing and Artificial Intelligence, AISC 151, pp. 709–716.
springerlink.com

1 Introduction

Metabolic Engineering is an approach which alters host cells to enable them to generate a novel or enhance the production of compounds for industrial and medical use. Recent studies have focused on the mean of analysis by modifying the computer readable data from the biological process. Hence, scientists are able to simulate the process inside the cell by using mathematical modeling and studying the metabolic pathway. Glycolysis is the metabolic pathway which produces pyruvate from glucose and is studied in this paper.

Parameter estimation is one of the critical steps in constructing a mathematical model. Unfortunately, it possesses several problems; on one hand the existence of noisy data leads to low accuracy [1], and on the other hand the increasing number of unidentified parameters and equations in the model makes it a complex model [2]. Thus, we proposed IDE which is a hybrid of DE and KF, to solve the problems regarding the increasing number of unidentified parameters which consequently adds to the difficulty of the model in estimating the kinetic parameters, and the existence of noisy data that leads to low accuracy for estimated result.

The benefits of using DE are efficiency, high speed, straightforwardness, and ease of use as it contains only few control parameters [3]. The use of KF can improve DE's performance as it updates the population with Kalman gain value which handles noisy data [1]. DE has been implemented as a parameter estimation approach by Christophe Chassagnole *et al.* [4] and Moonchai Sompop *et al.* [5] to enhance the production of bacteriocin, aspartate, beer, and the simulation of the real process in cell by estimating the kinetic parameters and control parameters. Parameter estimation with DE is done without noisy data handling process. Noisy data occurs when the obtained results differ from each other and this is caused by the apparatus limitation or human error. IDE takes advantage of KF which includes getting feedback from the noisy measurement to improve the performance of each result that was generated by DE.

2 An Improved Differential Evolution Algorithm

This paper proposes an improved differential evolution algorithm (IDE), which is a hybrid combination of DE [6] and KF [7]. In parameter estimation, existing algorithms [4, 5] solely use DE whereas IDE uses a hybrid of DE and KF. Fig. 2.1 shows the details of the IDE. Kinetic parameters existed in the glycolysis pathway model for yeast [8] go through IDE to estimate their optimal values. Fixed control parameter values used in this study are population size, NP=10, mutation factor, F=0.5, and crossover constant, CR=0.9.

In IDE, we added the process of updating the population as a new step that improved the conventional DE. This is a self-adapting approach. In conventional DE, the original population which is an m x n population matrix, is generated from the first generation (*Gen_1*) and continues until it reaches the maximum generation (*Gen_i*) in the initialization process. m represents the number of

generations and *n* represents the number of identifiable parameters. In evaluation process, the fitness function, *J* represented as

$$J = \sum_{i=1}^{N} |f(X, X0, \varnothing 0) - f(Y, X0, \varnothing)|^2 \quad (1)$$

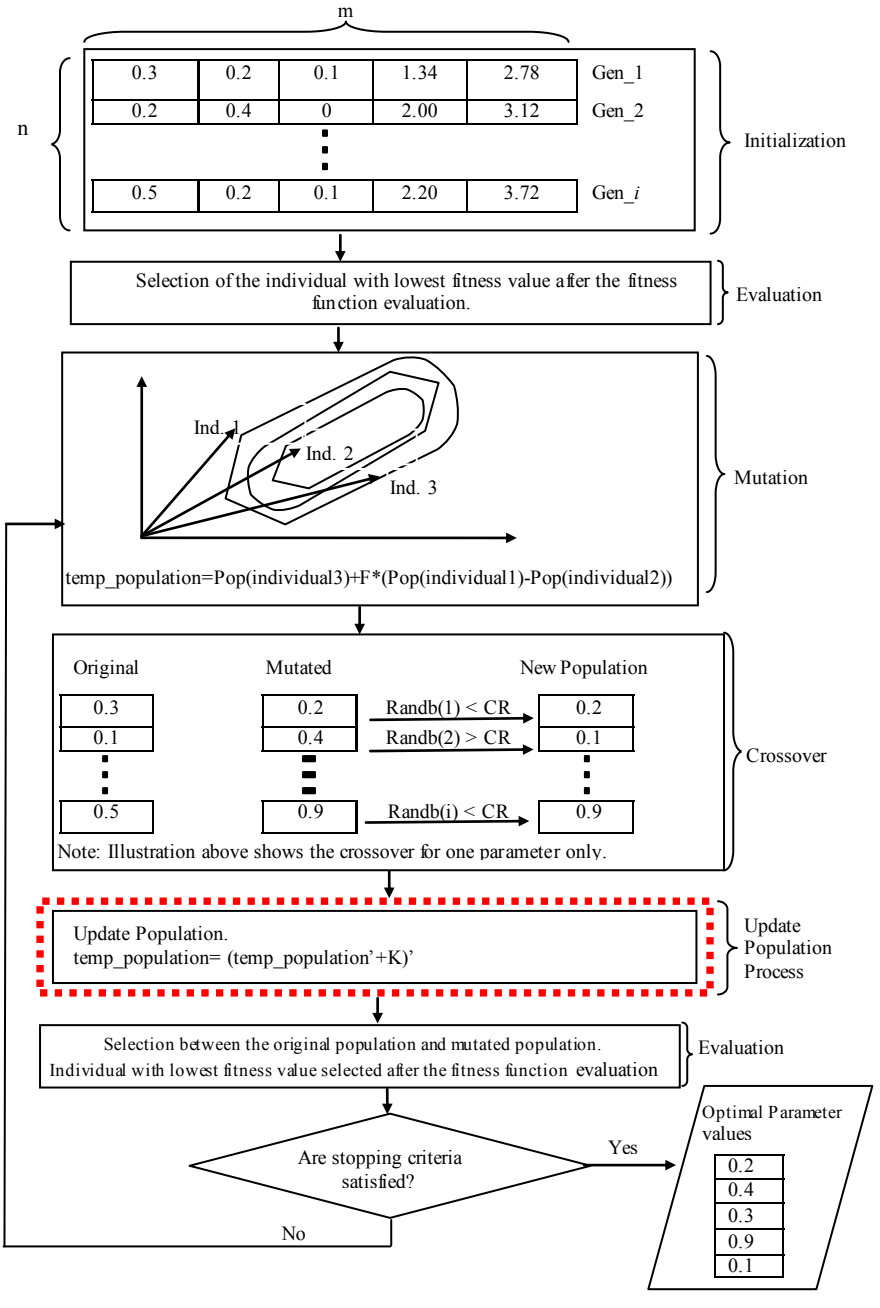
is applied to evaluate the fitness of each individual. *X* represents the state vector for measurement system, *Y* represents the state vector for simulated system, $\varnothing 0$ represents a set of original parameters, $\varnothing$ represents a set of estimated parameters, *X0* represents the initial state, *N*=the ending index, and *i*=the index variable.

In mutation process, three individuals (*Ind1*, *Ind2* and *Ind3*) first being selected then treated with the formula showed in Fig 2.1. In the mutation section, *temp_population* represents the mutated population matrix, *F* represents the mutation factor, and *Pop* represents the original population matrix. The subsequent crossover process is mainly performed based on *CR*, which indicates crossover constant value, and *Randb(i)* which indicates *i*-th random evaluation of a uniform random number generator [0,1]. If the *randb(i)* value of the individual in mutated population is lower than the *CR* value then that individual becomes the individual for the resultant population of the crossover process and vice versa. This is followed by the updating process that is performed according to the Equation 2.2. This step updates the population, which is generated by the crossover process and it is based on the Kalman gain value *K,* retrieved from the Equation 2.3. The Kalman gain value from the Equation 2.3 takes the process noise covariance and measurement noise covariance into account. These noisy data values were obtained from the experiment and in this study the noisy data values used are 0.1. After handling the noisy data, the updated population once again undergoes the evaluation process and the whole process is repeated till the stopping criterion is met. The stopping criteria are set via predefined maximum loop values or when the fitness functions have converged. The updating population process is highlighted with the dotted box in Fig. 2.1 and is carried out according to the following formula.

$$temp\_population = (temp\_population' + K)' \quad (2.2)$$

$$K = P * H' * inv(H * P * H' + R) \quad (2.3)$$

Where *K*=Kalman gain value, *A*=state transition matrix, *B*=input matrix, *H*=observation matrix, *Q*=process noise covariance, *R*=measurement noise covariance, *P*=covariance of the state vector estimate, and *H'*=inverse of matrix *H*.



Note: Updating population process is added after the crossover process to improve DE performance and it is highlighted with the dotted box.

Fig. 2.1 Schematic Overview of IDE.

3 Experimental Results

In this study, three estimation algorithms (GA, DE, and IDE) are compared. Kinetic parameter values in Table 3.1 are retrieved from literature review [8] and generated by the estimation algorithms. To evaluate the accuracy of each estimation algorithm, time series data for concentration of adenosine monophosphate (AMP) and pyruvic acid (PYR) were generated. AMP and PYR are significant metabolites. AMP acts as an energy regulator and sensor while PYR acts as an energy supplier in presence of oxygen through citric acid cycle. From the time series data, we calculate the average error rate. The details of the accuracy measurement are discussed in this section.

Table 3.1 Kinetic parameter values of IDE compared with GA and DE.

Kinetic parameters	Measurement kinetic parameter values[8]	Simulated kinetic parameter values		
		GA	DE	IDE
V1	0.5	1.1492	0.1934	0.1524
K1GLC	0.1	0.0695	0.3064	0.4450
K1ATP	0.063	0.0568	0.0451	0.0432
V2	1.5	7.8890	2.0267	1.4561
K2	0.0016	0.0010	0.0024	0.0037
k2	0.0017	0.0492	0.0019	0.0493
K2ATP	0.01	0.0185	0.0229	0.0099
K3f	1	0.3510	0.1984	5.5314
K3b	50	48.4765	72.5643	12.1150
V4	20	9.8508	9.1572	4.0841
K4GAP	1	0.5554	0.7145	0.4708
K4NAD	1	0.9907	1.2682	7.6088
K5f	1	0.4973	1.7580	0.7912
K5b	0.5	0.1361	0.6089	0.1241
V6	10	23.9385	16.5141	47.7182
K6PEP	0.2	0.1357	0.8816	0.8593
K6ADP	0.3	0.0618	0.4417	0.1116
V7	2	10.5984	0.4764	0.6771
K7PYR	0.3	0.7196	0.2546	0.2508
k8f	1	0.3443	0.3972	0.2114
k8b	0.000143	0.0002	0.0005	0.0003
k9f	10	26.5712	1.1240	2.2127
k9b	10	6.1839	54.3684	10.1453
k10	0.05	0.1264	0.0047	0.2720

Note: Shaded rows are the kinetic parameter values contributed to the calculation of average error rate for metabolite AMP in Table 3.2 and PYR in Table 3.3.

The measurement kinetic parameter values and simulated kinetic parameter values were substituted into the ordinary differential equations (ODEs) (Equation 3.1 and Equation 3.2) of AMP and PYR respectively.

$$\frac{dAMP}{dt} = -AMPflow - reaction_9 \tag{3.1}$$

$$\frac{dPYR}{dt} = -PYRflow + reaction\_6 - reaction\_7 \tag{3.2}$$

Where *reaction_6*=*compartment* * (V6 * *adp* * *pep* / ((K6PEP + *pep*) * (K6ADP + *adp*), *reaction_7*=*compartment* * (V7 * *pyr* / (K7PYR + *pyr*)), *reaction_9*= *compartment* * (k9f * *amp* * *atp* - k9b * power(*adp*,2)), *AMPflow*=*compartment* * *amp* * *flow*, *PYRflow*= *compartment* * *pyr* * *flow*, *compartment*=constant value of 1, *flow*=value fixed to 0.011, *amp*=concentration of AMP, *pyr*=concentration for PYR, *adp*=concentration for adenosine diphosphate, *atp*=concentration of adenosine triphosphate, and *pep*=concentration for phosphoenolpyruvic acid.

Time series data for concentration of AMP and PYR were subsequently generated from Equation 3.1 and Equation 3.2. The time series data contain measurement results, *y,* and simulated results *yi* for IDE, DE, and GA respectively. Error rate (*e*) and Average error rate (*A*) are calculated based on Equation 3.3, and Equation 3.4 respectively.

$$e = \sum_{i=1}^{N} (y - yi)^2 \tag{3.3}$$

$$A = \frac{e}{N} \tag{3.4}$$

Table 3.2 and Table 3.3 show the average error rate for AMP and PYR respectively.

Table 3.2 Average error rate for AMP.

Evaluation criteria	GA	DE	IDE
Average error rate, *A*	0.000247729	0.059147889	0.000099818

Note: Shaded column represents the best results.

Table 3.3 Average error rate for PYR.

Evaluation criteria	GA	DE	IDE
Average error rate, *A*	0.000038704	0.000109841	0.000004786

Note: Shaded column represents the best results.

For AMP (Table 3.2), IDE presented the lowest average error rate with 0.0000998178. DE showed the worst performance with the average error rate of 0.059147889. GA showed more moderate performance with average error rate of 0.000247729. On the other hand, for PYR (Table 3.3), IDE once again performed better than other estimation algorithms where its average error rate is 0.000004786. The average error rate for GA and DE are 0.000038704 and 0.0000109841 respectively

Table 3.4 shows execution time of each estimation algorithm on a Core i5 PC with 4GB main memory. The result shows that DE required the longest time (9 minutes and 30 seconds) to find the optimal value for all kinetic parameters compared to IDE which took the shortest time (6 minutes 55 seconds). It is shown that IDE tends to use less computation time than DE and GA.

Table 3.4 Execution time of IDE compared with GA and DE.

Computation usage	GA	DE	IDE
Execution time (hh:mm:ss)	00:07:12	00:09:30	00:06:55

Note: Shaded column represents the best results.

IDE exhibits lesser computation time and possesses a higher accuracy when compared to both GA and DE. The implementation of DE that aims to estimate the relevant kinetic parameters and the additional of Kalman gain value which targets to handle the noisy data has improved the computational time and accuracy. Hence, the IDE which is a hybrid of DE and KF minimizes the computational time and also increases the accuracy between the simulated results and measurement results.

4 Conclusion and Future Work

In this paper, the experiment to compare the performances of three different estimation algorithms using glycolysis pathway data in yeast [8] showed that an improved algorithm, IDE, which is a hybrid algorithm of DE and KF with the shortest execution time and the lowest average error rate performed better than the rest of algorithms. It successfully reduces the high difficulty of the system in estimating the relevant kinetic parameters resulting in shorter computation time. The ability to handle noisy data has contributed to an improved accuracy of the estimated results. In conclusion, IDE is shown to be superior compared to both GA and DE in terms of computational time and accuracy. IDE can be generalized where it can be implemented in the areas which its data consists of noisy for example electrical and electronic engineering field [9].

DE shows to be very delicate to control parameters: population size (*NP*), crossover constant (*CR*), and mutation factor (*F*) [10]. Thus, for future work, self-adapting approach to these control parameters can be implemented to enhance the

performance of the IDE. Moreover, additional steps can be added to the process of generating new populations with the aim of improving the performance of IDE.

Acknowledgments. This work is financed by Institutional Scholarship MyPhd provided by the Ministry of Higher Education of Malaysia. We also would like to thank Universiti Teknologi Malaysia for supporting this research by UTM GUP research grants (vot number: QJ130000.7123.00H67 and QJ130000.7107.01H29).

References

[1] Lillacci, G., Khammash, M.: Parameter Estimation and Model Selection in Computational Biology. PLoS Computational Biology 6(3), 1–17 (2010), doi:10.1371/journal.pcbi .1000696

[2] Chou, I.C., Voit, E.O.: Recent developments in parameter estimation and structure identification of biochemical and genomic systems. Mathematical Biosciences 219, 57–83 (2009), doi:10.1016/j.mbs.2009.03.002

[3] Wang, F.S., Chiou, J.P.: Differential evolution for dynamic optimization of differential-algebraic systems. In: IEEE International Conference on Evolutionary Computation, April 13-16, pp. 531–536 (1997), doi:10.1109/ICEC.1997.592367

[4] Chassagnole, C., Doncescu, A., Manyri, L., Yang, L.T.: Parameters Estimation by Differential Evolutionary Algorithms for Simulation of metabolic pathways in Escherichia coli. In: International Conference on AINA 2006, April 18-20, pp. 593–598. IEEE Computer Science (2006), doi:10.1109/AINA.2006.258

[5] Moonchai, S., Madlhoo, W., Jariyachavalit, K., Shimizu, H., Shioya, S., Chauvatcharin, S.: Application of a mathematical model and Differential Evolution algorithm approach to optimization of bacteriocin production by Lactococcus lactis C7. Bioprocess Biosyst. Eng. 28, 1–17 (2005), doi:10.1007/s00449-005-0004-5

[6] Storn, R., Price, K.: Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization 11(4), 341–359 (1997), doi:10.1023/A:1008202821328

[7] Cetto, J.A.: The Kalman Filter. Institut de Robotica i Informatica Industrial, UPC-CSIC. Llorens i Artigas 4-6, Edifici U, 2a pl. Barcelona 08028, Spain (2002), `http://digital.csic.es/bitstream/10261/30069/1/doc1.pdf` (accessed January 22, 2011)

[8] Nielson, K., Sorensen, P.G., Hynne, F., Busse, H.G.: Sustained oscillations in glycolysis: an experimental and theoretical study of chaotic and complex periodic behavior and of quenching of simple oscillations. Biophysical Chemistry 72, 49–62 (1998), doi:10.1016/S0301-4622(98)00122-7

[9] Ijaz, U.Z., Khambampati, A.K., Lee, J.S., Kim, S., Kim, K.Y.: Nonstationary phase boundary estimation in electrical impedance tomography using unscented Kalman filter. Journal of Computational Physics 227(15), 7089–7112 (2008), doi:10.1016/j.jcp.2007.12.025

[10] Feng, L., Yang, Y.F., Wang, Y.X.: A New Approach to Adapting Control Parameters in Differential Evolution Algorithm. In: Li, X., Kirley, M., Zhang, M., Green, D., Ciesielski, V., Abbass, H.A., Michalewicz, Z., Hendtlass, T., Deb, K., Tan, K.C., Branke, J., Shi, Y. (eds.) SEAL 2008. LNCS, vol. 5361, pp. 21–30. Springer, Heidelberg (2008), doi:10.1007/978-3-540-89694-4_3