Aspartate Biosynthesis Pathway Simulation Using an Improved Differential Evolution Algorithm through Parameter Estimation

Chuii Khim Chong[1], Mohd Saberi Mohamad[1], Safaai Deris[1], Shahir Shamsir[2], Yee Wen Choon[1] and Lian En Chai[1]

[1] Artificial Intelligence and Bioinformatics Research Group, Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia.

[2] Department of Biological Sciences, Faculty of Biosciences and Bioengineering, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia

Corresponding author's e-mail: saberi@utm.my

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

Introduction.

Metabolic Engineering is a method which allows modifications of the pathways in suitable host cells. It aims at producing a novel or achieving an expected amount of desire compound for medical and industrial use. Recent studies mainly have concentrated on the aim of analysis by altering the computer readable data from the biological process. Thus, with the study of metabolic pathway, scientists can simulate the process inside the cell by mathematical modeling. The main goal of system biology is to develop an accurate pathway model that functions as a biological function simulator. Aspartate biosynthesis pathway is a sequence of events that occur in a cell causing production of amino acid called aspartate. Aspartate is very crucial in the urea cycle for the proper elimination of waste products from dietary protein.

Parameter estimation is one of the crucial steps in constructing mathematical model. Regrettably, it faces a number of difficulties, for example high complexity of the model which is caused by the increasing number of unidentified parameters and equations in the model [1], and the existence of noisy data which causes low accuracy [2]. Thus, we proposed IDE which is a hybrid of DE and KF, to solve the existence of noisy data that leads to low accuracy for estimated result and the increasing unidentified parameters which lead to the complexity of the model. Noisy data can occur when the retrieved results differ from each other and this is due to apparatus limitation or human error.

The advantages of DE are speed, straightforwardness, efficiency, and ease of use as it contains only few control parameters [3]. Moreover, KF can improve DE's performance as it uses Kalman gain value which handles noisy data to update the population [2]. Moonchai Sompop et al. [4] and Christophe Chassagnole et al. [5], implemented DE as a parameter estimation approach to enhance the production of aspartate, bacteriocin, beer, and the simulation of the actual process in cell by estimating the control parameters and kinetic parameters.

Methods.

The IDE proposed in this paper is a hybrid of DE [6] and KF [7]. Kinetic parameters contained in the aspartate biosynthesis pathway model for plant model *Arabidopsis* [8] undergo IDE to estimate its optimal value. Fixed control parameter values used in this study are as follow:

I. Population size, *NP*: 10
II. Mutation factor, *F*: 0.5
III. Crossover constant, *CR*: 0.9

A new step, the process of updating population is added to the conventional DE. This is a self-adapting approach. In conventional DE, the original population first execute initialization and an *m* x *n* population matrix is generated from first generation (*Gen_1*) till maximum generation (*Gen_i*). *m* indicates number of identifiable parameters and *n* shows number of generations. In evaluation process, the fitness function, *J* showed as

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

is implemented to analyze the fitness of each individual. *X* indicates state vector for measurement system, *Y* represents state vector for simulated system, ∅ shows set of unknown parameters that used for parameter estimation, *u* indicates the external force e.g. noisy data, *N*=the ending index, and *i*=the index variable.

Three individuals (*Ind1*, *Ind2* and *Ind3*) first being selected then applied in the formula showed in Eq. 2. In mutation section, *temp_population* shows the mutated population matrix, *F* indicates mutation factor, and *Pop* represents the original population. The following crossover process is mainly carried out according to *CR*, which represents crossover constant value, and *Randb(i)* which represents *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 turns into the individual for the final population of the crossover process and vice versa. Updating process is the subsequence process and it is done based on the Eq. 3. This step updates the population, which is generated by the crossover process and it is carried out according the Kalman gain value *K,* gained from the Eq. 4. The Kalman gain value from the Eq. 4 involved measurement noise covariance and process noise covariance. These noisy data values were gained from the experiment and the noisy data values implemented in this study are 0.1. The updated population once again undergoes the evaluation process after handling the noisy data, and the whole process is repeated till the stopping criterion is met. Stopping criteria are the fitness functions have converged or predefined maximum loop values. The updating population process is emphasized with the dotted box in Fig. 1. It is performed based on the following formula.

$$temp\_population = Pop(individual3) + F * (Pop(individual1) - Pop(individual2))' \qquad (2)$$

$$temp\_population = (temp\_population' + K)' \qquad (3)$$

$$K = P * H' * inv(H * P * H' + R) \qquad (4)$$

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*.

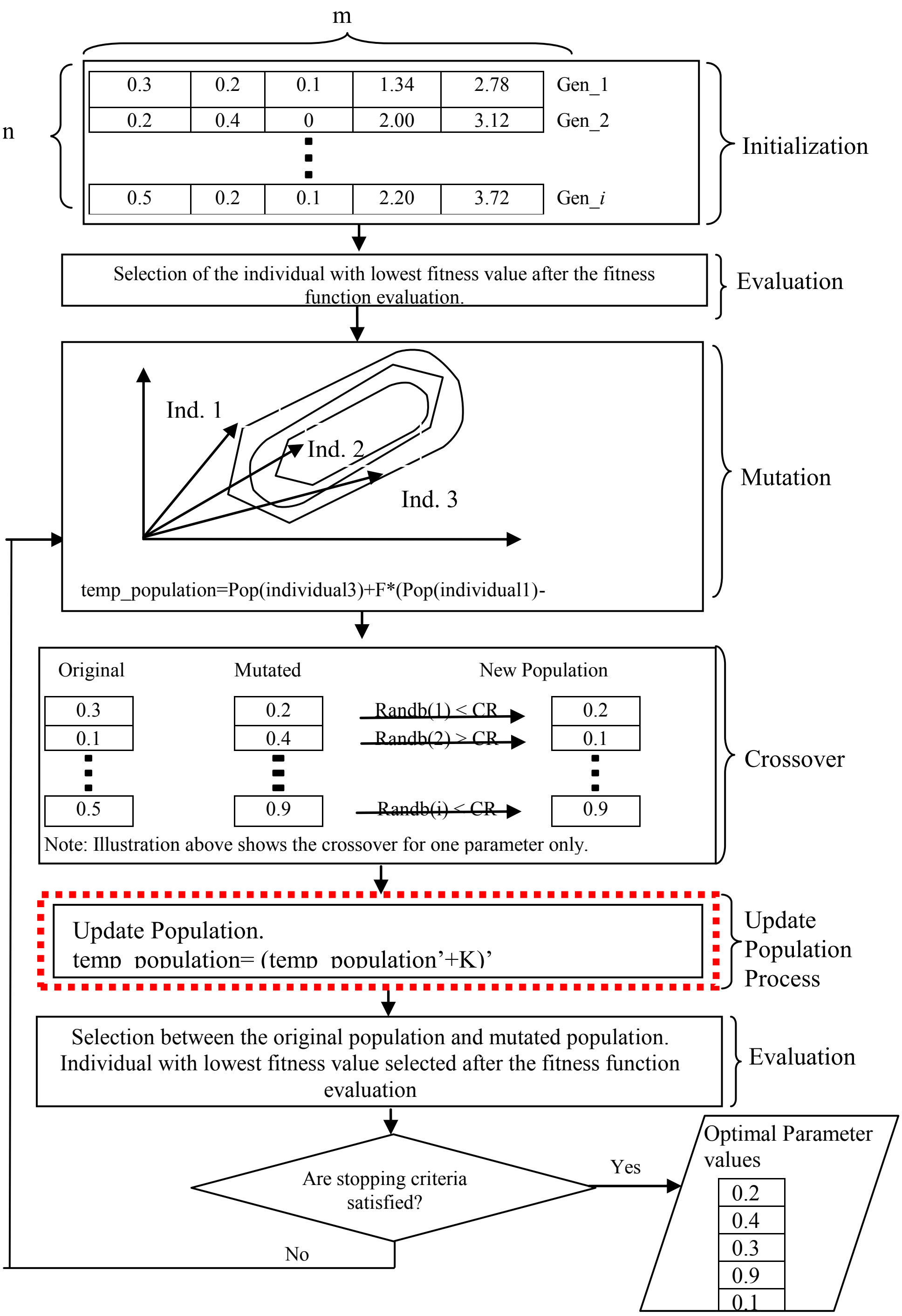
Results and Discussions.

Three estimation algorithms (GA, DE, and IDE) are studied in this paper. Kinetic parameter values in Table 1 were gained from literature review [8] and generated by the estimation algorithms. Time series data for concentration of Aspartyl_P (AspP) was produced to analyze the accuracy of each estimation algorithm. AspP is symbolic metabolites selected to be observed in this paper. AspP is a precursor for aspartate production. We calculated the average of error rate and standard deviation (STD) values from the time series data.

Table 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
Vak1_AK1_kforward_app_exp	5.65	30.1	12.6062	31.9836
Vak1_AK1_kreverse_app_exp	1.6	1	0.8541	2.5848
Vak1_AK1_Lys_Ki_app_exp	550	307.5	200.183	107.050
Vak1_AK1_AdoMet_Ka_app_exp	3.5	0.6	2.7837	4.3236
Vak1_AK1_nH_exp	2	7.9	3.5692	0.5969
Vak2_AK2_kforward_app_exp	3.15	2.5	4.8357	0.41
Vak2_AK2_kreverse_app_exp	0.86	1.2	0.3506	6.8017
Vak2_AK2_Lys_Ki_app_exp	22	52.8	35.8675	149.518
Vak2_AK2_nH_exp	1.1	1.2	1.0746	4.1122
VakI_AKI_kforward_app_exp	0.36	0.6	0.0988	4.3702
VakI_AKI_kreverse_app_exp	0.15	0.2	0.1548	0.0444
VakI_AKI_Thr_Ki_app_exp	124	14.7	246.685	23.2625
VakI_AKI_nH_exp	2	2.3	3.186	0.7184
VakII_AKII_kforward_app_exp	1.35	0.3	0.3799	0.1569
VakII_AKII_kreverse_app_exp	0.22	0.3	0.6536	0.1368
VakII_AKII_Thr_Ki_app_exp	109	1211.7	563.736	15.7876
VakII_AKII_nH_exp	2	2	2.9304	0.5377
Vasadh_ASADH_kforward_app_exp	0.9	0.4	0.38	1.3291
Vasadh_ASADH_kreverse_app_exp	0.23	0.9	0.6343	0.26

Note: Table shows the kinetic parameter values used in the calculation of average of error rate and STD values for metabolite AspP in Table 2.


m
n
0.3 0.2 0.1 1.34 2.78 Gen_1
0.2 0.4 0 2.00 3.12 Gen_2
0.5 0.2 0.1 2.20 3.72 Gen_i
Initialization
Selection of the individual with lowest fitness value after the fitness function evaluation.
Evaluation
Ind. 1
Ind. 2
Ind. 3
temp_population=Pop(individual3)+F*(Pop(individual1)-
Mutation
Original
Mutated
New Population
0.3
0.1
0.5
0.2
0.4
0.9
Randb(1) < CR
Randb(2) > CR
Randb(i) < CR
0.2
0.1
0.9
Note: Illustration above shows the crossover for one parameter only.
Crossover
Update Population.
temp_population= (temp_population'+K)'
Update Population Process
Selection between the original population and mutated population. Individual with lowest fitness value selected after the fitness function evaluation
Evaluation
Are stopping criteria satisfied?
Yes
No
Optimal Parameter values
0.2
0.4
0.3
0.9
0.1

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

Figure 1 Schematic Overview of IDE.

The measurement kinetic parameter values and simulated kinetic parameter values were assigned into the ordinary differential equations (ODEs) (Eq. 5) of AspP.

$$\frac{dAspP}{dt} = +Vak1 + Vak2 + VakI + VakII - Vasadh \qquad (5)$$

Where *Vak1* = *c1* * *AK1* * ((*Vak1_AK1_kforward_app_exp* - *Vak1_AK1_kreverse_app_exp* * *AspP*) / (1 + power(*Lys* / (*Vak1_AK1_Lys_Ki_app_exp* / (1 + *AdoMet* / *Vak1_AK1_AdoMet_Ka_app_exp*)), *Vak1_AK1_nH_*exp))), *Vak2* = *c1* * *AK2* * ((*Vak2_AK2_kforward_app_exp* - *Vak2_AK2_kreverse_app_exp* * *AspP*) / (1 + power(*Lys* / *Vak2_AK2_Lys_Ki_app_exp*, *Vak2_AK2_nH_exp*))), *VakI* = *c1* * *AKHSDHI* * ((*VakI_AKI_kforward_app_exp* - *VakI_AKI_kreverse_app_exp* * *AspP*) / (1 + power(*Thr* / *VakI_AKI_Thr_Ki_app_exp*, *VakI_AKI_nH_exp*))), *VakII* = *c1* * *AKHSDHII* * ((*VakII_AKII_kforward_app_exp* - *VakII_AKII_kreverse_app_exp* * *AspP*) / (1 + power(*Thr* / *VakII_AKII_Thr_Ki_app_exp*, *VakII_AKII_nH_exp*))), Vasadh = *c1* * *ASADH* * (*Vasadh_ASADH_kforward_app_exp* * *AspP* - *Vasadh_ASADH_kreverse_app_exp* * *ASA*), *c1*=1, *AK1*= 0.25, *AspP*=concentrate of AspP, *Lys*=concentrate of *Lys*, Adomet=20, *AK2*=0.25, *AKHSDHI*=0.63, *Thr*=concentration of Thr, *AKHSDHII*= 0.63, *ASADH*=11.6, *ASA*=concentrate of ASA .

Time series data for concentration of AspP was then produced from Eq. 5. The time series data consist of measurement result, *y,* and simulated results *yi* for IDE, DE, and GA respectively. Error rate (*e*), Average of error rate (*A*), and STD value are calculated according Eq. 6, Eq. 7, and Eq. 8 respectively.

$$e = \sum_{i=1}^{N}(y - yi)^2 \qquad (6)$$

$$A = \frac{e}{N} \qquad (7)$$

$$STD = \sqrt{\frac{e}{N}} \qquad (8)$$

Table 2 shows the average of error rate and STD values for AspP.

Table 2. Average of error rate and STD values for AspP.

Evaluation criteria	GA	DE	IDE
Average of error rate, *A*	0.055672	0.274077	0.023673

Note: Shaded column represents the best results.

For AspP (Table 2), IDE presented the lowest average of error rate with 0.023673. DE showed the worst performance with 0.274077 for the average of error rate. GA showed more moderate performance with average error rate of 0.055673.

Table 3 shows execution time of each estimation algorithm on a Core i5 PC with 4GB main memory. From the table, IDE used the shortest time (7 minutes) whereas for DE used the longest time (7 minutes and 06 seconds) to find the optimal value for all kinetic parameters. It is showed that IDE tend to use less computation time than GA and DE.

Table 3. Execution time of IDE compared with GA and DE.

Computation usage	GA	DE	IDE
Execution time (hh:mm:ss)	00:07:04	00:07:06	00:07:00

Note: Shaded column represents the best results.

Table 4 shows the mean and STD values of fitness value for aspartate biosynthesis pathway for 50 runs. Fitness function implemented in this study is minimizing the difference between measurement results and simulated results. Based on the result from the table, STD values and mean for metabolites AspP are 0.090716 and 0.0094954. It shows that IDE is a reliable parameter estimation algorithm as the mean and STD values are closer to 0 indicates the simulated results are closer to the measurement results.

Table 4. Mean and standard deviation (STD) values of fitness value for aspartate biosynthesis pathway for 50 runs

Evaluate Criteria	AspP
Mean	0.090716
STD	0.094954

IDE is proven to be more accurate when compared to both GA and DE and require less computation time. It is proven that the usage of Kalman gain value which handles noisy data for the process of updating population, consequently improves the results of this study in terms of accuracy. On the other hand, the usage of DE lessens the complexity in estimating unknown relevant kinetic parameters efficiently. Hence, the IDE which is a hybrid of DE and

KF increases the accuracy between the simulated results and measurement results and also lessens the computational time. Besides, IDE proved that it is a reliable algorithm.

Summary and Future Work

In this paper, experiments were carried out for three estimation algorithms using aspartate biosynthesis pathway data in plant model *Arabidopsis* [8]. IDE, an improved algorithm, which is a hybrid algorithm of DE and KF showed the shortest execution time and the lowest average of error rate. The ability of KF to handles the noisy data contributes to better accuracy of the estimated results and the usage of DE successfully minimizes high complexity of the system that leads to decreases the computation time. STD value of IDE is closer to 0 had also proved that it is a reliable estimation algorithm. In conclusion, IDE is shown to be superior compared to both DE and GA in terms of accuracy and computational time.

DE shows to be very delicate to control parameters: population size (*NP*), crossover constant (*CR*), and mutation factor (*F*) [9]. Thus, for future work, self- adapting approach to these control parameters can be applied to enhance the performance of the IDE as well as conventional DE. Beside that, 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 Student Affair Department (HEMA) and Universiti Teknologi Malaysia for supporting this research by UTM GUP research grants (vot number: QJ130000.7123.00H67 and QJ130000.7107.01H29) .

References

[1] I.C. Chou and E.O. Voit: *Recent developments in parameter estimation and structure identification of biochemical and genomic systems,* Mathematical Biosciences, Vol. 219 (2009), p. 57-83.

[2] G. Lillacci and M. Khammash: *Parameter Estimation and Model Selection in Computational Biology*, PLoS Computational Biology, Vol. 6, No. 3 (2010), p. 1-17.

[3] F.S. Wang and J.P. Chiou in: *Differential evolution for dynamic optimization of differential-algebraic systems*. Evolutionary Computation, IEEE International Conference (1997), in press.

[4] S. Moonchai, W. Madlhoo, K. Jariyachavalit, H. Shimizu, S. Shioya and S. Chauvatcharin: *Application of a mathematical model and Differential Evolution algorithm approach to optimization of bacteriocin production by Lactococcus lactis C7*, Bioprocess Biosyst Eng, Vol. 28 (2005), p. 1-17

[5] C. Chassagnole, A. Doncescu, L. Manyri and L.T. Yang in: *Parameters Estimation by Differential Evolutionary Algorithms for Simulation of metabolic pathways in Escherichia coli*. IEEE Computer Science, AINA '06 International Conference (2006), in press.

[6] R. Storn and K. Price: *Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces*, Journal of Global Optimization, Vol. 11, No.4, (1997), p. 341-359

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

[8] G. Curien, O. Bastien, M. Robert-Genthon, A. Cornish-Bowden, M.L. Cárdenas and R.Dumas: *Understanding the regulation of aspartate metabolism using a model based on measured kinetic parameters*, Mol. Syst. Biol. Vol. 5, No. 271, (2009), p. 1-32.

[9] L. Feng, Y.F. Yang and Y.X. Wang: *A New Approach to Adapting Control Parameters in Differential Evolution Algorithm*, Lecture Notes in Computer Science, Vol. 5361/2008 (2008), p. 21-30