Theronine Biosynthesis Pathway Simulation Using IBMDE With Parameter Estimation

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

[1]Artificial Intelligence and Bioinformatics Research Group, Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia, Skudai, 81310 Johor, Malaysia.
ckchong2@live.utm.my, {saberi, safaai}@utm.my, {lechai2, ywchoon2}@live.utm.my

[2]Department of Biological Sciences, Faculty of Biosciences and Bioengineering, Universiti Teknologi Malaysia, Skudai,81310 Johor, Malaysia
shahir@utm.my

*Corresponding author

Abstract. When analysing a metabolic pathway through mathematical model, it is important that the significant parameters are being correctly estimated. However, this process often comes across problems such as easily being trapped in local minima, repetitive exposure to worse results during the search process, and occurrence of noisy data. Thus, an improved Bee Memory Differential Evolution algorithm (IBMDE), which is a hybrid of the Differential Evolution algorithm (DE), the Kalman Filter (KF), Artificial Bee Colony algorithm (ABC), and a memory feature is presented this paper. IBMDE is an improved estimation algorithm as previous work only utilised DE. The theronine biosynthesis pathway is the metabolic pathways used in this paper. For metabolite O-Phosphohomoserine production simulation, the IBMDE able to produce the estimated optimal kinetic parameter values with significantly reduced error rate (63.67%) and shows a faster convergence time (71.46%) compared to the Nelder Mead (NM), the Simulated Annealing (SA), the Genetic Algorithm (GA), and DE respectively. In addition, IBMDE demostrates to be a reliable estimation algorithm.

Keywords: Parameter Estimation, Differential Evolution Algorithm, Kalman Filter, Artificial Bee Colony Algorithm, Memory feature.

1. Introduction

Systems biology manifests a biological system by a set of ordinary differential equations (ODEs) in mathematical models [1]. The essential interactions show quantitatively through ODEs with the aim in explaining the behaviour at the system level. Hand-tuning and *in-vitro* biochemical experiments are the main methods to retrieve the values of unknown parameters in the ODEs [2]. Under some conditions, these values are collected through estimation, and therefore, it is important that the estimation methods used in mathematical models are thoroughly studied [1]. Parameter estimation in system biology reduces the variance between experimental data and simulated data. It usually works as a part of a recursive process to develop mathematical models that are able to provide optimal estimated values for biological systems. Nevertheless, increasing number of unknown parameters and noisy experimental data of dynamic biochemical pathways cause most traditional estimation methods to generate inaccurate estimations [3].

Under the category of evolutionary algorithms, the Differential Evolution algorithm (DE) has been found to be the best estimation algorithm. It works to optimize a problem repetitively with the fixed objective function. The major advantages of DE are efficiency, high speed, ease of use, and simplicity [4]. It had been implemented by Moonchai *et al*. [5] to improve the production of bacteriocin by estimating the control parameters which were temperature and pH. Therefore, in this paper, it is implemented as the main estimation algorithm to solve the increasing number of unknown parameters. Kalman gain, *K*, value is used by the Kalman Filter (KF) in handling the noisy data through normalization. IDE (the Improved Differential Evolution algorithm) is the hybrid of these two algorithms [6].

Easily trapped in the local minimal due to faster convergence speed [4] and attempts to expose to worse results during the search process repeatedly are the disadvantages of DE. Therefore, to solve the mentioned disadvantages, IDE is then further combine with two modifications – the artificial bee colony algorithm (ABC) and a memory feature to generate the improved Bee Memory Differential Evolution algorithm (IBMDE). ABC capable to rise the probability in finding the optimal solutions by the food source possibility and this characteristic avoides the trapped in local minima [7]. The memory feature, however, capable to keep track of the best candidate ever during the search process with the extra memory named *gbest* and this prevents the worse result from being exposed again.

The paper is structured into four sections, where Section 2 introduce the method implemented, IBMDE, and subsequently Section 3 with experimental setup, results, and discussion. Finally, the conclusion and future works is showed in Section 4.

2. Methodology

An improved estimation algorithm, the Improved Bee Memory Differential Evolution algorithm (IBMDE) is presented in this paper. Previous work [5] only used DE while IBMDE uses a hybrid of DE, KF, ABC, and memory feature for parameter estimation. The details of the IBMDE is demostrated in Figure 1. As in conventional DE, a n x m population matrix, P, the initial population is produced in the initialization process. m is the number of unknown parameters and n is the number of generations. Each gene of the individuals in the initial population gained its own value based on Equation (1). Each gene represents a parameter value and individual, (Ind_i), indicates a set of estimated parameter values (possible solution) which i is the index variable where $1 \leq i \leq n$. For initialisation process, g indicates the ones matrix with dimension of n x 1, $I_{initial} = \{I_1, I_2, \ldots, I_m\}$ where I is the initial values for each parameter, and $rand(n,m)$ indicates a n x m matrix with random values between 0 and 1.

$$P = g * I_{initial} * 10^{0.5*rand(n*m)} \tag{1}$$

In the evaluation process, the fitness function, J, as shown:

$$J_i = \sum_{j=1}^{m} | f(D,E_{\exp}) - f(D,S_{sim}) |^2 \tag{2}$$

is implemented to analyse the fitness of each individual. D are the ordinary differential equations (ODEs) implemented to obtained the time series data values, $E_{exp} = \{E_1, E_2, \ldots, E_m\}$ where E is the experimental parameter values, $S_{sim} = \{S_1, S_2, \ldots, S_m\}$ where S is the optimal estimated parameter values, j is the index variable where $1 \leq j \leq m$, and $f(D, z)$ is the function to retrieve time series data values with fixed parameter values, z. The best candidate (candidate with the lowest fitness value) is stored in a memory named g*best*. Equation (3) is used to calculate the probability of each candidate, $prob(i)$, where $fit(i)$ is the fitness value with index i and max indicates the function to obtain the maximum value. The upper bound and lower bound are altered to individuals with the highest and lowest probability value respectively after obtaining the probability value of each individual as long as the fitness value is not converged.

$$prob(i) = 0.1 * (fit(i) / \max(fit(i))) + 0.9 \tag{3}$$

$$M = P(i_3) + F * (P(i_1) - P(i_2)) \tag{4}$$

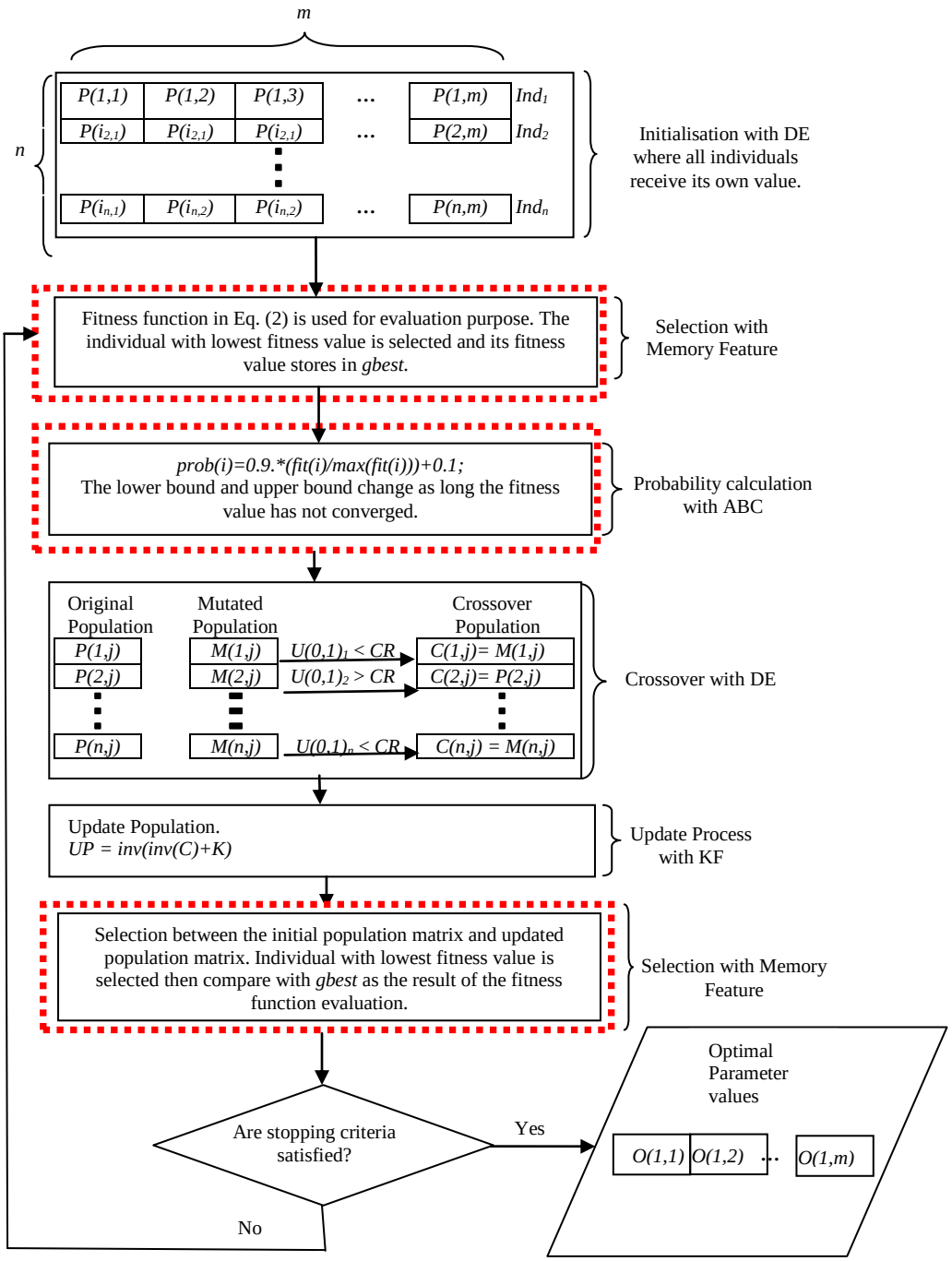
Three individuals (i_1,i_2,i_3) are chosen and then are replaced into the formula as presented in Equation (4). *M* indicates the mutated population matrix and *F* is the mutation factor. The subsequent crossover process is primarily executed according to the *CR* and $U(0,1)_i$ values. For mutation population matrix, each individual has its own *U(0,1)* value. *CR* indicates the crossover constant value, $U(0,1)_i$ is the uniform random value between 0 and 1 with index *i*, and *C* is the crossover population matrix. If the *CR* value is lower than the *U(0,1)* value of individual in the mutated population, then the mutation population's individual would become the resultant population's individual for the crossover process and vice versa.

The following is the updating process which is executing based on Equation (5). This step would update the population which is generated by the crossover process and it is performed according to Kalman gain value, *K*, retrieved from Equation (6). The *K* value from Equation (6) takes the process and measurement noise covariance into account and *UP* indicates the updated population matrix**.** In this study, *B* and *R* matrix are identity matrices while *H* is retrieved from Jacobian matrix and the ODEs information. Besides that, *H* matrix is invertible but it does not have to be a square matrix and its number of rows must be equivalent with the number of unknown parameters. For *in silico* approaches, Gaussian noise is used to simulate the noisy data so the model is close to the nature of biology. After a small number of trials are performed with the reasonable range of 0 to 1, the noisy data value implements in this study is 0.1. The evaluation process is performed where the results retrieved from the update process are analysed with the initial population's individual after normalizing the noisy data. Individual with lowest fitness value is chosen between initial and updated population. The results generate from the evaluation process would be replaced by *gbest* values and records in solution population matrix, *O*, if its fitness value is lower than the fitness value of *gbest* and vice versa. Next, the whole process is repeated until the stopping criterion is met. The stopping criteria are set via predefined maximum loop values or when the fitness function has converge. The modified selection and ABC are highlighted with the dotted box in Figure 1.

$$UP = inv(inv(C) + K) \tag{5}$$

$$K = P * H' * inv(H * P * H' + R) \tag{6}$$

where *K* = Kalman gain value, *P* = covariance of the state vector estimate, *H* = observation matrix, *R* = measurement noise covariance, *inv* = inverse function, and *H*' = inverse of matrix *H*,.



Note: The ABC and modify evaluation processes are highlighted in red.

Fig. 1. Flowchart of IBMDE.

3. Experimental Setup

The optimal values for kinetic parameters that consisted in the threonine biosynthesis pathway model for *E-coli* [8] are gathered then undergo IBMDE. These pathway consists of 11 metabolites, 11 ODEs, seven reactions involved and 46 unknown parameters; but only 1 ODE, two reactions and ten unknown parameters are used for generating metabolite O-Phospho-homoserine (HSP). HSP is a substrate for threonine biosynthesis while threonine uses as treament for several nervous system disorders. The control parameters' values used in this study are crossover constant, *CR*=0.9, mutation factor, *F*=0.5, and population size, *NP*=10. These parameter values showed better results than the other values after a small number of trials are conducted between the reasonable range of 0 to 10. In this study, the main software used are the Copasi and SBToolbox in Matlab 2008a. An online database, Biomodel, is managed by European Bioinformatics Institute (EMBL-EBI) is used to retrieve the metabolic pathways.

The Nelder Mead (NM), the Simulated Annealing (SA), the Genetic Algorithm (GA), DE, and IBMDE are five estimation algorithms implemented in this study to allow the comparisons to be performed. Table 1 shows the experimental kinetic parameter values collected from previous related work [8] and the simulated kinetic parameters are generated by all the mentioned estimation algorithms. Average of error rates which were produced from the time series data values for the concentration of the metabolite HSP are implemented to assess the accuracy of the algorithm. Moreover, the simulation is repeated 50 times to calculate its standard deviation (*STD*) value and the average of error rates are then tested statistically with the chi square test to evaluate the reliability of the algorithm.

Table 1. Kinetic parameter values of IBMDE compared with NM, SA, GA and DE for metabolite HSP.

Kinetic parameters	Measurement kinetic parameter values[8]	Simulated kinetic parameter values				
		NM	SA	GA	DE	IBMDE
vtsy_vm5	0.0434	0.038	0.089	2.849	0.038	0.181
vtsy_k5hsp	0.31	0.461	0.282	0.820	0.144	2.183
vhk_vm4f	0.1	0.204	0.057	16.577	1.690	62.174
vhk_lys	0.46	0.524	2.271	0.0564	0.524	1.750
vhk_k4lys	9.45	7.875	10.351	1.886	0.775	109.980
vhk_k4atp	0.072	0.052	0.026	15.507	0.013	0.110
vhk_k4ihs	4.7	5.952	1.532	3.2988	5.901	2.327
vhk_k4hs	0.11	0.2	0.017	6.6031	4.184	51.068
vhk_k4thr	1.09	1.2100	3.5	0.0224	1.307	4.164
vhk_k4iatp	4.35	3.1638	0.397	4.148	7.7305	248.351

Note: Table shows the kinetic parameter values implemened in the calculation of average of error metabolite HSP in Table 2.

Both experimental and simulated kinetic parameter values are subsituted into the ordinary differential equations (ODEs) for the metabolite HSP, as shown below:

$$\frac{dHSP}{dt} = vtsy + vhk \tag{7}$$

where
vtsy = *compartment* * vtsy_vm5 * *hsp* / (*hsp* + vtsy_k5hsp),
vhk = *compartment* * (vhk_vm4f * *hs* * *atp* / ((1 + vhk_lys / vhk_k4lys) * (*atp* + vhk_k4atp * (1 + *hs* / vhk_k4ihs)) * (*hs* + vhk_k4hs * (1 + *thr* / vhk_k4thr) * (1 + *atp* / vhk_k4iatp)))),
compartment = constant value of 1,
adp = concentration for metabolite adenosine diphosphate which is equal to 0,
atp = concentration of metabolite adenosine triphosphate which is equal to 10,
hsp = concentration of metabolite HSP which is equal to 0,
hs = concentration of metabolite homoserine which is equal to 0,
thr = concentration of metabolite threonine which is equal to 2.

Equation (7) is used to retrieve the time series data values for the concentration of metabolite HSP. Experimental results, *y*, and simulated results y_{sim}, for NM, SA, GA, DE, and IBMDE are consisted in the time series data values respectively. Equation (8), Equation (9), and Equation (10) are used to calculate the error rate (*e*), average of error rate (*A*), and standard deviation (*STD*) value respectively.

$$e = \sum_{w=1}^{Q} (y - y_{sim})^2 \tag{8}$$

$$A = \frac{e}{Q} \tag{9}$$

$$STD = \frac{\sum_{w=1}^{Q} ((y - y_{sim})^2 - mu)^2}{Q - 1} \tag{10}$$

where *w*= the index variable, *mu*= the mean value, and *Q*= the number of rows of time series data values.

4. Experimental Results and Discussion

The average of error rate for each estimation algorithm is shown in Table 2. The results show that IBMDE has the lowest average of error rate with 0.001764 metabolite HSP. This proved that with the capability to keep tract of the best candidate ever during the search process by the memory named *gbest*, the accuracy of the estimation result has enhanced. This is due to the fact that with the memory, *gbest*, the search process has been prevented from being explored to worse results.

Table 2. Average of error rates for metabolite HSP.

Metabolite	NM	SA	GA	DE	IBMDE
HSP	0.002830	0.002699	0.0052727	0.001886	0.001764

Note: Shaded column represents the best results.

Table 3 shows the number of generations needed for each estimation algorithm to converge to its optimal fitness value for metabolite HSP whereas the execution time of each estimation algorithm on a Core i5 PC with 4GB main memory for metabolite HSP is shown in Table 4. Based on the results, SA requires the longest time, 698.2019 seconds with 5009 number of generations to converge to the optimal value for all kinetic parameters. On the contrary, the IBMDE requires the shortest time, 343.4834 seconds with 4 number of generations. This implied that less execution time and less number of generation required to converge able to enhance the effiency of the estimation algorithm in identifying the optimal solution as it can be retrieved in a shorter time. In short, the addition of a memory feature has avoided exposing space with worse results, and the addition of ABC has reduced the search space. Therefore, IBMDE showed higher accuracy and shorter computational time.

Table 3. Number of generations for metabolite HSP.

Metabolite	NM	SA	GA	DE	IBMDE
HSP	33	5009	91	86	4

Note: Shaded column represents the best results.

Table 4. Execution times in unit of second (s) for metabolite HSP.

Metabolite	NM	SA	GA	DE	IBMDE
HSP	531.1887	698.2019	400.5667	398.3966	343.4834

Note: Shaded column represents the best results.

Figure 2 shows the metabolite production graph for the metabolite HSP based on the kinetic parameters collected from previous related work [8] and produced by the mentioned estimation algorithms. The results illustrate that the kinetic parameters generated by IBMDE have enhanced the production rate as the experimental line is

lower than its dotted simulation line. This enhancement is supported by the increase in speed and concentration kinetic parameters for metabolite HSP as compared to previous related work [8]. The speed kinetic parameters named vm5 and vm4f, rise by 0.138 $\mu mol/(l*\min)$ and 62.0741 $\mu mol/(l*\min)$ respectively while for the concentration kinetic parameters - vtsy_k5hsp, vhk_k4atp, vhk_k4hs, vhk_k4thr, and vhk_k4iatp, the values rise by 1.8733 $mmol/l$, 0.383 $mmol/l$, 50.9584 $mmol/l$, 3.0741 $mmol/l$, and 244.0015 $mmol/l$ respectively. μmol is micromole, $mmol$is milimole, l is liter, s is seconds, and min is minutes. In Figure 2, ORI indicates the production graph that is generated with the kinetic parameters obtained from previous related work [8] whereas IBMDE, DE, GA, NM, and SA indicate the production graphs that are generated by IBMDE, DE, GA, NM, and SA. Speed is assumed to be the crucial kinetic parameter which can be increased to enhance the the interested metabolites' production [10]. The reasons that cause the increase in the speed parameter are increase in the surface area, substrate concentration, temperature, and the addition of catalyst. The enzyme and temperature fail to be the reasons to increase the speed parameter under particular conditions. When none of the reaction presents the flux coefficient value is equal to one and the temperature implemented exceeds the optimal temperature of the enzyme, then the addition of temperature and enzyme cause no effect. Other than that, the production of the interested metabolite can be improved by rising the concentration of the reactants. This implies that the product increase as more sources.

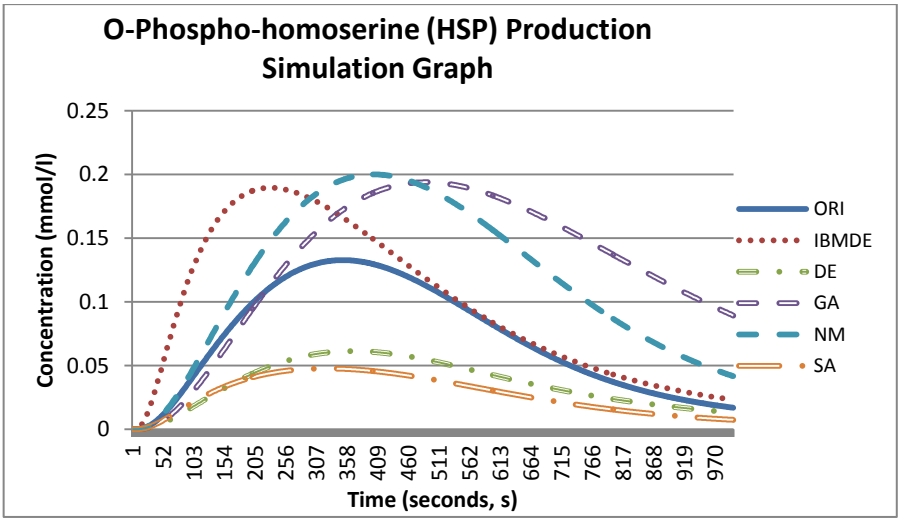


Fig. 2. Production Simulation Graph for NM, SA, GA, DE and IBMDE of metabolite O-Phospho-homoserine (HSP).

Table 5 shows the benckmark functions tested with Particle Swarm Optimisation (PSO), DE, and IBMDE. For these benchmark function tests, control parameters used for PSO are swarm size= 40, inertia weight= 0.7290, particle's best weight= 1.4945, and swarm's best weight= 1.4945 while for DE and IBMDE are *NP*=20, *CR*= 0.9, and *F*=0.5. Based on the results, IBMDE passes four tests out of five tests. Once again, it showed its ability in obtaining the optimal solution but it failed in the Rosenbrock test. This is due to the fact that Rosenbrock's landscape modifies from simple to complex and this implies that the diversity of control parameters is getting lesser. The mentioned problem can be solved by requiring large number of generations.

Table 5. Performance evaluations among PSO, DE, and IBMDE on benchmark functions.

Optimisation algorithm	Ackley	Griewank	Rastrigin	Rosenbrock	Sphere
IBMDE	9.47E-07	1.50E-05	4.44E-14	1.18E-08	8.98E-18
PSO	1.96E-06	9.15E-04	2.38E-10	2.97E-12	3.48E-14
DE	2.08E-06	5.14E-04	1.21E-12	4.33E-09	1.83E-17

Note: Shaded column represents the best results.

A fitness function is used to reduce the variance between experimental and simulated results in this study. Based the results obtained from this experiment, the *STD* value and mean for the metabolite HSP are 0.001753 and 0.00136 respectively. Standard deviation is a calculation of how widely are the values being distributed from the mean (the average value). The results generated by IBMDE were consistent and the difference between each 50 run was small as the *STD* value for IBMDE was close to the mean value. Chi-square test (X^2 test), a statistical test has to be performed as stated by Lillacci and Khammash, 2010 [2], in order to assure that the simulated results are statistically consistent with the experimental results. The confidence coefficient, γ, and degree of freedom, *s*, used in this paper are 0.995 and 1. Interval estimates, σ^2, produced based on *s*, γ, and the formula found in Lillacci and Khammash, 2010 [2] are $0.00004 < \sigma^2 < 9.550$. The hypothesis proposed is the simulated results are statistically consistent with the experimental results. The X^2 value for the metabolite HSP is 0.00004 based on the chi-square equation; which implies that the X^2 value is existed in within the range of σ^2. Thus, the hypothesis is accepted as the IBMDE passed the X^2 test. The estimated results demonstrated to be statistically consistent with the experimental results.

5. Conclusion

IBMDE, a hybrid of DE, KF, ABC, and memory feature effectively reduced the search space through the probability value obtained from ABC in the search process and ultimately resulted in faster convergence time while only DE was used in previous work [5]. The exposing of worse search spaces has avoided with the ability to store of the best candidate ever with the *gbest* value during the search process and consequently helped in enhancing the accuracy of the estimated results. In short, IBMDE performed better than SA, NM, GA, and DE in terms of computational time and accuracy. Furthermore, IBMDE also has proved that it is a reliable estimation algorithm as it passed the chi square test and can be used in used the areas that contains noisy data for example in the electrical and electronic engineering field. Besides that, IBMDE can be implemented to other metabolic pathways to improve the interested metabolites which are essential for medical and industrial use.

DE is very sensitive towards its control parameters: mutation factor (*F*), crossover constant (*CR*), and population size (*NP*) [9]. Thus, as future work, the self-adapting approach to these control parameters can be added to improve the performance of the conventional DE as well as the IBMDE.

Acknowledgments

We would like to thank Malaysian Ministry of Science, Technology and Innovation for supporting this research by an e-science research grant (Grant number: 06-01-06-SF1029). This research is also supported by UTM GUP research grant that was sponsored by Universiti Teknologi Malaysia.

References

1. Steuer R.: Exploring the Dynamics of Large-Scale Biochemical Networks: A Computational Perspective. The Open Bioinformatics Journal 5, 4-15 (2011)
2. Lillacci G., Khammash M.: Parameter Estimation and Model Selection in Computational Biology. PLoS Computational Biology 6(3), 1-17 (2010)
3. Zhong J., Martin B.: Joint State and Parameter Estimation For Biochemical Dynamic Pathways With Iterative Extended Kalman Filter: Comparison With Dual State and Parameter Estimation. The Open Automation and Control Systems Journal 2, 69-77 (2009)

4. Chiou J.P., Wang F.S.: Estimation of Monod model parameters by hybrid differential evolution. Bioprocess and Biosystems Engineering 24, 109-113 (2001)
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 and Biosystems Engineering 28, 1–17 (2005)
6. Chong C.K., Mohamad M.S., Deris S., Shamsir M.S., Choon Y.W., Chai L.E.: Improved Differential Evolution Algorithm for Parameter Estimation to Improve the Production of Biochemical Pathway. International Journal of Interactive Multimedia and Artificial Intelligence 1(5), 22-29 (2012)
7. Karaboga D.: An idea based on honey bee swarm for numerical optimization. Technical Report TR06, (2005)
8. Chassagnole C., Fell DA., Rais B., Kudla B., Mazat JP.: Control of the threonine-synthesis pathway in Escherichia coli: a theoretical and experimental approach, Biochem J. 356 (2), 433-444 (2001)
9. Feng L., Yang Y.F., Wang Y.X.: A New Approach to Adapting Control Parameters in Differential Evolution Algorithm. Lecture Notes in Computer Science 5361/2008, 433-444 (2008)
10. Ferchichi M., Crabble E., Hintz W., Gil G.H., Almadidy A.: Influence of culture parameters on biological hydrogen production by Clostridium saccharoperbutylacetonicum ATCC 27021. World Journal of Microbiology & Biotechnology 21, 855-862 (2005)