Using Particle Swarm Optimization for Estimating Kinetics Parameters on Essential Amino Acid Production of *Arabidopsis Thaliana*

Siew Teng Ng[1], Chuii Khim Chong[1], Yee Wen Choon[1], Lian En Chai[1], Safaai Deris[1], Rosli M. Illias[2], Mohd Shahir Shamsir[3], and Mohd Saberi Mohamad[1,*]

[1] Artificial Intelligence and Bioinformatics Research Group, Faculty of Computer Science and Information Systems, UniversitiTeknologi Malaysia, Skudai, 81310 Johor, Malaysia
evelyn_siewteng@hotmail.com,
{ckchong2,ywchoon2,lechai2}@live.utm.my, {safaai,saberi}@utm.my
[2] Department of Bioprocess Engineering, Faculty of Chemical Engineering, UniversitiTeknologi Malaysia, Skudai, 81310 Johor, Malaysia
r-rosli@utm.my
[3] Department of Biological Sciences, Faculty of Biosciences and Bioengineering, Universiti Teknologi Malaysia, Skudai 81310, Johor, Malaysia
shahir@fbb.utm.my

Abstract. Parameter estimation is one of nine phases in modelling, which is the most challenging task that is used to estimate the parameter values for biological system that is non-linear. There is no general solution for determining the nonlinearity of the dynamic model. Experimental measurement is expensive, hard and time consuming. Hence, the aim for this research is to implement PSO into SBToolbox to obtain optimum kinetic parameters for simulating essential amino acid metabolism in plant model *Arabidopsis Thaliana*. There are four performance measurements, namely computational time, average of error rate, standard deviation and production of graph. PSO has the smallest standard deviation and average of error rate. The computational time in parameter estimation is smaller in comparison with others, indicating that PSO is a consistent method to estimate parameter values compared to the performance of SA and downhill simplex method after the implementation into SBToolbox.

Keywords: Parameter Estimation; PSO; SBToolbox; *Arabidopsis Thaliana.*

1 Introduction

It is complex to understand the regulation, structure and organization of the underlying biological system because it needs quantitative assessment and reliable understanding of the system functions.

* Corresponding author.

A.S. Sidhu & S.K. Dhillon (Eds.): *Adv. in Biomedical Infrastructure 2013*, SCI 477, pp. 51–61.
DOI: 10.1007/978-3-642-37137-0_7

Modeling is a process to transform the symbol model into a numerical model which enables us to understand the model deeply. It converts the biological system into a simple analogue that is easier to analyze, interrogate, predict, extrapolate, manipulate, and optimize than the biological system itself. There are 9 phases in mathematical modelling as shown in Figure 1 according to Chou and Voit[1]. At molecular level, the variables represent the concentration of chemical species such as protein, mRNA and so on. With the known pathway structure, we are able to write down the equation, which depends on several parameters. The parameters might be the reaction rate, production and decay coefficient, approximation or reduction that is satisfied by the structure of the system. Normally, the parameters are unknown. The measurement, if done experimentally, is expensive, hard and time consuming.

Estimation of parameter values is one of the steps in the modelling process. Parameter estimation helps to determine appropriate numerical parameter values that can convert the symbolic model into a numerical model and makes the latter consistent with experimental observations [1]. Among the nine phases, parameter values estimation is the most challenging task. This is due to the previous phases of parameter estimation that will affect the difficulties of the estimation. Examples are like the selection of modelling framework, the size and complexity of the hypothesized model and so on. It will be easier if the model is an explicit linear model that uses linear regression methods. Nevertheless, as soon as the model becomes nonlinear, many of these methods will become inapplicable [1] .

In addition, biological model is nonlinear and dynamic. Hence, parameter estimation is complex because there is no general solution exists due to the model's nonlinearity. It is easier to analyze if it is a linear model since linear regression methods are used.

The model above describes the specific phenomena of biological system. It contains parameters that can alter the model behavior and it can be measured directly or inferred from the data. Parameter estimation is the process to determine appropriate numerical parameter values that can convert the symbolic model into a numerical model and makes the latter consistent with the experimental observations [1].

Optimization is a scientific discipline that deals with the detection of optimal solutions for a problem, among other alternatives. Optimization models the actual problem by building a proper mathematical function, or called as objective function. Among all feasible solutions where the solution fulfils all the constraints, global optimization tends to find the optimal one [2]. To estimate the parameter in a system, it is necessary to identify the objective function. Then, the objective function will be minimized by using appropriate optimization methods.

In order to simulate the biological system, parameter estimation is the most important phase because with complete and accurate set of parameter value, the system can be characterized. However, it is not always possible to measure these values in wet lab experiments due to high demands on cost and time, since there is no existing general solution to determine the nonlinearity of the dynamic model. Non-linear system is any problem that cannot be written as a linear combination of independent components and thus the result is not directly proportional to the input.

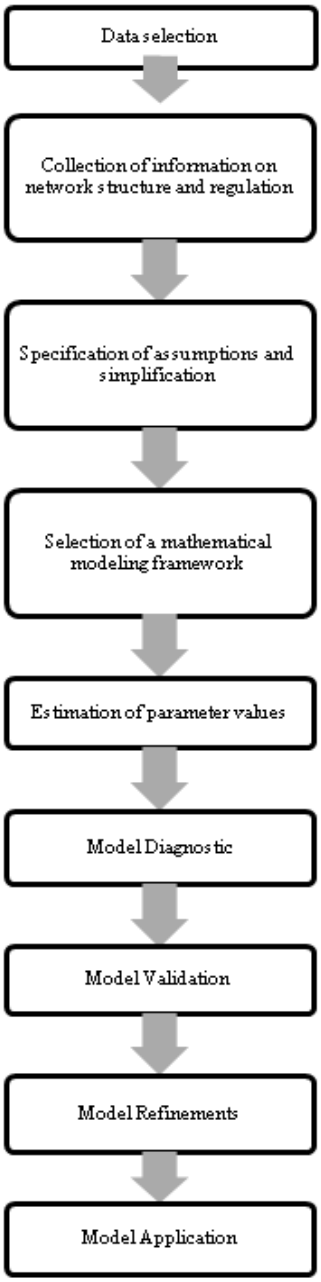

Fig. 1. Mathematical modelling [1]

As a result, it is difficult to obtain and researchers need to spend more time to solve the system since it needs to carry out the experiment within unknown time in order to get the best result. Furthermore, there are certain parameters which have no appropriate measurement method yet [3]. Exploration of several optimization techniques to minimize cost function is necessary to obtain the optimal value. Based on the research by Syed Murtuza Baker etalon on the estimation of the kinetic parameters of upper part of glycolysis process [3], comparison of several methods were performed and the result stated that SA took the longest time in order to converge to the best solution. Even though GA was able to complete the estimation in a shorter time, it tended to be stuck in local minima. Moreover, PSO was able to produce better result compared to other methods.

There are several optimization methods in the SBToolbox such as Genetic Algorithm (GA)[4], Simulated Annealing (SA)[5], downhill simplex method[6] and so on. However, there has been no implementation of Particle Swarm Optimization (PSO) [7] to estimate kinetic parameters to simulate the essential amino acid metabolism in plant model Arabidopsis Thaliana yet. Furthermore, most of the parameter estimations used other algorithms such as SA, GA, EP (Evolutionary Programming) [3] and so on, and completed the set of kinetics parameters for aspartate metabolism by using appropriate method to estimate the kinetic parameter of aspartate metabolism which was not presented.

PSO is one of the methods based on swarm intelligence to estimate the kinetic parameter values. The concept of PSO is that the particles will fly in limited number of directions and have flying experience by their own or with their companion along the search space in certain velocity; and they are expected to fly to the best position.

In this research, PSO is proposed and implemented into SBToolbox in MATLAB to estimate the parameter values of aspartate metabolism in plant model Arabidopsis Thaliana. This method is inspired by bird flocks, fish schools and animal herds when foraging. The significance of the study is that there is no implementation of Particle Swarm Optimization (PSO) into SBToolbox to estimate kinetic parameters to simulate essential amino acid metabolism in plant model, Arabidopsis Thaliana, yet. PSO is a consistent method in estimating parameter values. It takes a shorter time to converge to the best value. It has the ability to find the optima in fast pace. Besides that, very few parameters are needed to adjust in order to obtain the optimal value. PSO is computationally inexpensive in terms of memory requirements and speed [8].

2 Method

Previous works have implemented GA, SA, downhill simplex method, and so on in parameter estimation. In this paper, we propose PSO as a new approach for parameter estimation. In this section, the details of the proposed Particle Swarm Optimization for estimating parameter values are discussed. The steps involved to obtain optimal parameter values are summarized in Figure 2.

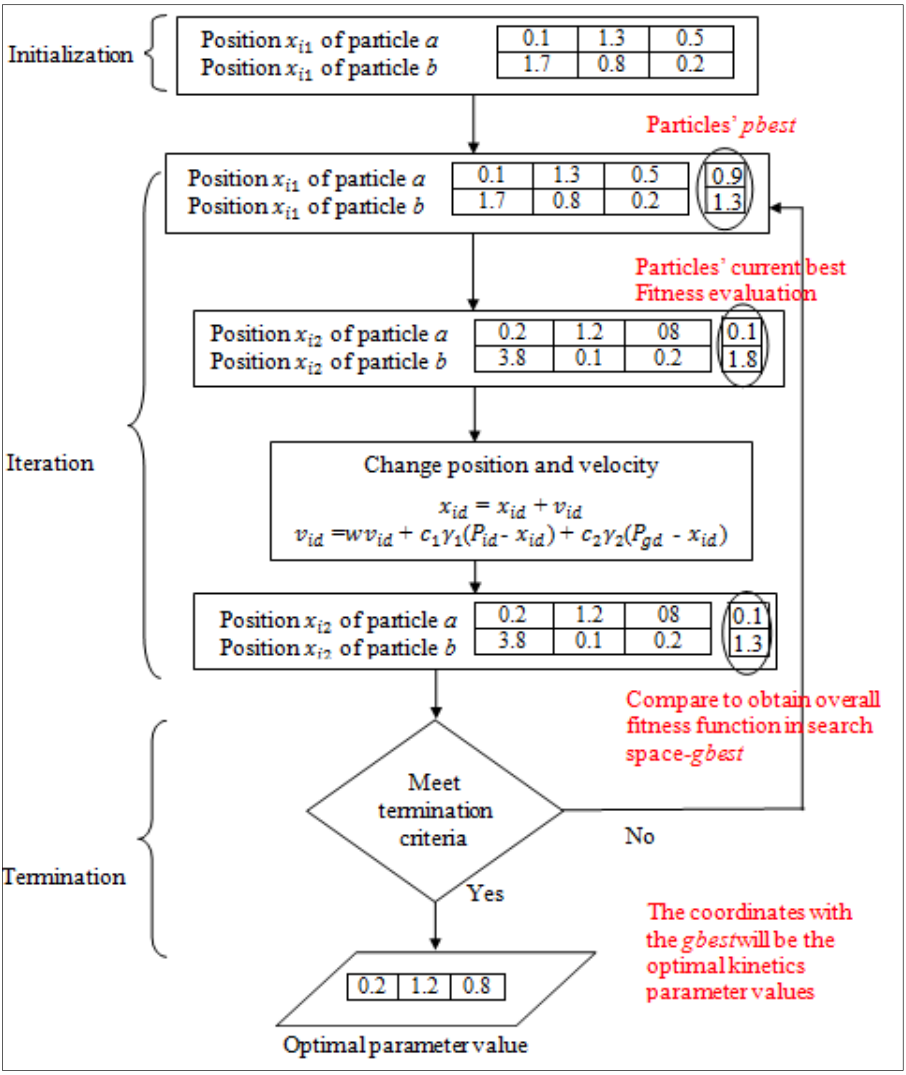

Fig. 2. Three steps involved to estimate parameter values using PSO

2.1 Initialization

Initially, the population array of particles with random positions and velocities on D dimensions in search space was initialized. Then, we defined the number of iterations, inertia weight, positive constant and swarm size. In this study, the inertia weight was 1.0, the positive constant was 2.0, and the number of iteration was 100. Next, the desired optimization fitness function in d variables for each particle was evaluated.

2.2 Iteration

In this part, a loop function was used to search and update the best position. There were two values being updated if best values were found in each iteration which were global best- gbest and best solution (fitness solution)- pbest value.

Initially, the particles' fitness evaluation was compared with particles'spbest. If current value is better than pbest, then set pbest value is equal with the current value and the pbest location equal to the current location in d-dimensional space. Then, we compared fitness evaluation with the population's overall previous best. If current value is better than gbest, then the gbest is reset to the current value. After being updated using Equation 1 and 2, the optimization fitness function in d variables for each particle was evaluated again.

$$x_{id} = x_{id} + v_{id} \tag{1}$$

$$v_{id} = wv_{id} + c_1\gamma_1(P_{id} - x_{id}) + c_2\gamma_2(P_{gd} - x_{id}) \tag{2}$$

Where $X_i = (X_{i1}, X_{i2}, \ldots, X_{iD})$: ithparticle's position in search space, $V_i = (V_{i1}, V_{i2}, \ldots, V_{iD})$: ith particle's velocity, $P_i = (P_{i1}, P_{i2}, \ldots, P_{iD})$: Best position of the ith, $P_g = (P_{g1}, P_{g2}, \ldots, P_{gD})$: Best position in the whole swarm,i = 1, 2 ,…, m, indicates each particle in one population. d = 1,2,…, D, indicates the number of dimension, c_1, c_2: Acceleration constant representing the pulling of each particle toward pbest and gbest. γ_1 , γ_2 : Random number between 0 and 1, $v_{id} \in [-v_{max}, v_{max}]$, v_{max}: maximum velocity decided by user and w = inertia weight that provides the balance between global and local exploration and exploitation to find a sufficient optimal solution.

2.3 Termination

The loop continues until a criterion is met where optimum parameter values are obtained or a maximum number of iteration is reached.

2.4 Dataset

In this research, the dataset used was the aspartate metabolism [9] of Arabidopsis Thaliana. In this research, the kinetic parameters for Lysine, Threonine and Isoleusine were estimated using PSO in SBToolbox [10]. There were 9 kinetic parameters, 16 kinetics parameters, 6 kinetic parameters respectively. Table 1 shows the list of kinetic parameters that needed to be estimated, experimental values, the kinetic parameters values estimated using SA, simplex and PSO.

Table 1. List of kinetic parameters with measured kinetic parameter values for Lysine

Kinetic parameter	Measured kinetic parameter values	SA	Simplex	PSO
Vdhdps1_DHDPS1_k_app_exp	1	0.7019	0.9384	0.4726
Vdhdps1_DHDPS1_Lys_Ki_app_exp	10	10.1627	12.0480	10
Vdhdps1_DHDPS1_nH_exp	2	1.8208	1.9279	1.7768
Vdhdps2_DHDPS2_k_app_exp	1	1.0846	10	1
Vdhdps2_DHDPS2_Lys_Ki_app_exp	33	33.3325	34.5784	32.0637
Vdhdps2_DHDPS2_nH_exp	2	2	20	0.9687
VlysTRNA_Lys_tRNAS_Lys_Km	25	15.0701	22.8179	35.1274
VlysKR_LKR_kcat_exp	3.1000	0.3430	3.1305	10.0065
VlysKR_LKR_Lys_Km_exp	13000	121600	12350	60575

3 Result and Discussion

In this study, PSO was implemented into SBToolbox in MATLAB to estimate parameter value. Three algorithms; SA, downhill simplex method and PSO were used to estimate the parameters and the result produced by two algorithms were compared. To evaluate the consistency and accuracy of both algorithms, the average of error rate and standard deviation were compared. There were 50 runs for estimating all the kinetic parameters and the formulas used to calculate the standard deviation are as follow:

$$e = \sum_{i=1}^{N}(y - y_i)^2 \tag{3}$$

$$A = \frac{e}{N} \tag{4}$$

$$STD = \sqrt{\frac{e}{N}} \tag{5}$$

The Equation 3 and 4 were used to calculate the error rate and average of error rate. Then, the standard deviation was obtained using Equation 5, where y_i is simulated results, y is measurement result, e is error rate, A is average of error rate and N is the number of sample. This equation was used to compare the performance of PSO with other methods. The best performance among the methods could be the method with the lower average of error rate and the standard deviation value close to 0 which indicated that PSO was able to produce high accuracy result.

After the discussion on the performance of PSO in estimating kinetics parameter of three amino acids, this section discusses and compares the performance of the three methods including PSO, SA and downhill simplex method. Based on Table 2, the standard deviation values of SA and downhill simplex method did not get close to 0 compared to standard deviation value of PSO. The values were 0.0733, 0.1211 and 0.0113 respectively. Meanwhile, the standard deviation values were 0.0733, 0.1211 and 0.0113 which PSO had the value that was the closest to 0. Based on Figure 3, the simulated line produced by PSO that was the closest to experimental line compared to SA and downhill simplex method. Having the smallest average of error rate, standard deviation value closer to 0 and simulated line closest to experimental line shows that PSO is a more consistent method to estimate parameter values compared to SA and downhill simplex method. In addition, the computational time for PSO to estimate 9 kinetics parameters was 315.9816 seconds which took a shorter time to complete compared to SA which took 4834.0581 seconds and 585.9037 seconds for downhill simplex method. The smaller average of error rate, standard deviation value closer to 0 and simulated line closest to experimental line shows that PSO is a more consistent method to estimate parameter values compared to SA and downhill simplex method. In addition, the computational time for PSO to estimate 9 kinetics parameters was 315.9816 seconds which took a shorter time to complete compared to SA and downhill simplex method. In addition, the computational time for PSO to estimate 9 kinetics parameters was 315.9816 seconds which took a shorter time to complete compared to SA which took 4834.0581 seconds and 585.9037 seconds for downhill simplex method. The smaller average of error rate, standard deviation value closer to 0 and simulated line closest to experimental line shows that PSO is a more consistent method to estimate parameter values compared to SA and downhill simplex method. In addition, the computational time for PSO to estimate 9 kinetics parameters was 315.9816 seconds which took a shorter time to complete compared to SA and downhill simplex method. We have conducted 50 runs with three algorithms and the STD values are shown in Table 2. The results showed that PSO has the lowest STD value; this indicates that the different between each run is small and this proved that it is a reliable estimation algorithm.

PSO had the smallest average of error rate, standard deviation values closer to 0 and the simulated line closer to the experimental line. The results obtained show that PSO outperformed SA and simplex in estimating kinetics parameters of Lysine, threonine and Isoleucine. It also shows that PSO is the most consistent method used in this research. The use of GA to estimate the kinetics parameters easily gets stuck in local minima and as a result, the accuracy of the kinetics parameters values will be low. This can be solved by using PSO due to the inertia weight taken into account in

PSO which was able to avoid being stuck into local minima by increasing the global search ability. The inertia weight produced the balance between the local and global exploration and exploitation. The computational time used to estimate the kinetics parameters is higher by using other algorithms and this can be solved by using PSO, proven by the short time taken in this research. This is the result of PSO which is inspired by bird flocking, fish schooling etc which does not require generation of new population for each iteration, which is time-consuming, but each particle from the same population will fly to better solution in each iteration. Hence, this decreases the time complexity. Furthermore, the steps involved in PSO are less complex compared to other algorithms such as GA which need to undergo selection, mutation and crossover. Besides that, the appropriate acceleration constant in PSO is able to ensure each particle fly towards pbest and gbest, which then lets PSO be able to converge to the best solution faster compared to other algorithms. If the constant value is too low, the particle will tend to fly away from the best solution; at the same time the high value of acceleration constant will make the particle pass the target.

Table 2. Comparison of average of error rate, standard deviation and execution time in seconds between SA, downhill simplex method and PSO for Lysine production from *Arabidopsis Thaliana*

Method / Feature	SA	Downhill simplex method	PSO
Computational time (second)	4834.0581	585.9037	315.9816
Average of error rate	0.0318	0.1520	0.0057
Standard deviation	0.0733	0.1211	0.0113

Note: Shaded column represents the best results.

PSO had the smallest average of error rate, standard deviation values closer to 0 and the simulated line closer to the experimental line. The results obtained show that PSO outperformed SA and simplex in estimating kinetics parameters of Lysine, threonine and Isoleucine. It also shows that PSO is the most consistent method used in this research. The use of GA to estimate the kinetics parameters easily gets stuck in local minima and as a result, the accuracy of the kinetics parameters values will be low. This can be solved by using PSO due to the inertia weight taken into account in PSO which was able to avoid being stuck into local minima by increasing the global search ability. The inertia weight produced the balance between the local and global exploration and exploitation. The computational time used to estimate the kinetics

parameters is higher by using other algorithms and this can be solved by using PSO, proven by the short time taken in this research. This is the result of PSO which is inspired by bird flocking, fish schooling etc which does not require generation of new population for each iteration, which is time-consuming, but each particle from the same population will fly to better solution in each iteration. Hence, this decreases the time complexity. Furthermore, the steps involved in PSO are less complex compared to other algorithms such as GA which need to undergo selection, mutation and crossover. Besides that, the appropriate acceleration constant in PSO is able to ensure each particle fly towards pbest and gbest, which then lets PSO be able to converge to the best solution faster compared to other algorithms. If the constant value is too low, the particle will tend to fly away from the best solution; at the same time the high value of acceleration constant will make the particle pass the target.

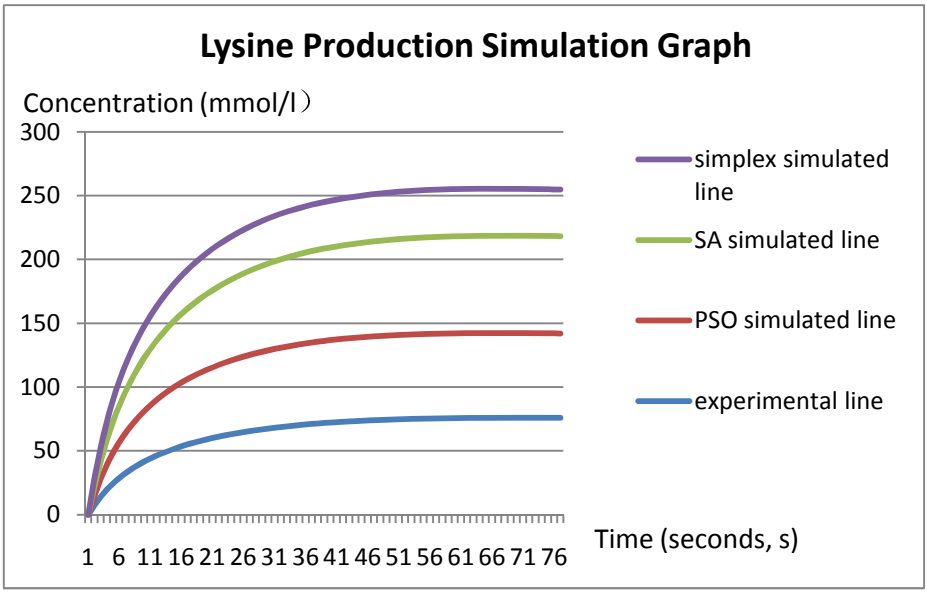

Fig. 3. Comparison of simulated line of SA, downhill simplex method and PSO with experimental line for Lysine production

4 Conclusion

In conclusion, the performance of PSO in estimating parameter values is better than SA and downhill simplex method after the implementation of PSO into SBToolbox in MATLAB. The simulated results generated by PSO are more consistent, as the standard deviation value is closer to 0 compared to SA and downhill simplex methods. The graph also shows that the simulated line of PSO is closer to experimental line. Moreover, the computational time to estimate parameter values for SA and downhill simplex method are longer compared to PSO. This is due to PSO which applies inertia weight to obtain a balance between the local and global exploration and exploitation to

avoid getting stuck into the local minima. In addition, PSO takes a shorter time to converge to best solution. Besides that, the acceleration constant that is taken into account in the equation ensures that each particle is pulled towards the pbestandgbest positions. In this research, value 2 was applied. In conclusion, Parameter Estimation through experiment is time consuming, hard and expensive. However, the implementation of PSO into SBToolbox manages to reduce the computational time for parameter estimation. It also reduces the complexity and the cost needed to use to estimate the kinetics parameters since the estimation only involves the use of computer. For future work, the number of run may be increased to ensure the accuracy of the method and more different weight parameters can be implemented to enhance the performance of PSO.

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. 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)
2. Parsopoulos, K.E., Vrahatis, M.N.: Particle Swarm Optimization and Intelligence. Hersey, New York (2010)
3. Baker, S.M., Schallau, K., Junker, B.H.: Comparison of different algorithms for simultaneous estimation of multiple parameters in kinetic metabolic models. Journal of Integrative Bioinformatics 7(3), 133 (2010)
4. Houck, C.R., Joines, J.A., Kay, M.G.: A genetic algorithm for function optimization: a Matlab implementation. Technical Report: NCSU-IE-TR-95-09, North Carolina State University, Raleigh, NC (1995)
5. Aarts, E., Korst, J.: Simulated Annealing and Boltzmann Machines. A Stochastic Approach to Combinatorial Optimization and Neural Computing. John Wiley & Sons, New York (1989)
6. Nelder, J., Mead, R.: The downhill simplex method. Computer Journal 7, 308–313 (1965)
7. Eberhart, R., Shi, Y.: Particle swarm optimization: developments, applications and resources. In: Proc. Congress on Evolutionary Computation, Service Center, Piscataway, NJ, Seoul, Korea (2001)
8. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proc. of IEEE International Conference on Neural Networks (ICNN), Perth, Australia (1995)
9. Curien, G., Bastien, O., Robert-Genthon, M., Cornish-Bowden, A., Cárdenas, M.L., Dumas, R.: Understanding the regulation of aspartate metabolism using a model based on measured kinetic parameters. Mol. Syst. Biol. 5(271), 1–32 (2009)
10. Schmidt, H., Jirstrand, M.: Systems Biology Toolbox for MATLAB: A computational platform for research in Systems Biology. Bioinformatics 22(4), 514–515 (2005)