ICIC Express Letters
Part B: Applications
Volume **4**, Number **1**, February **2013**

 ISSN 2185-2766
pp. 1–6

A NEW HYBRID BEE EVOLUTION ALGORITHM FOR PARAMETER ESTIMATION IN BIOLOGICAL MODEL

Afnizanfaizal Abdullah, Safaai Deris and Mohd Saberi Mohamad

Artificial Intelligence and Bioinformatics Research Group
Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia
81310 UTM, Johor, Malaysia
{ afnizanfaizal; safaai; saberi }@utm.my

Received May 2012; accepted August 2012

Abstract. *The development of accurate and reliable models for biological systems plays an important role in both systems and synthetic biology. The models are constructed based on the ordinary differential equations to observe the concentration change of specific biochemical products. These formulations usually depend on a set of parameters that reflect the physical properties of the systems, such as reaction and kinetic rates. In most cases, these parameters are estimated by fitting the model prediction with the corresponding experimental data. Due to the noisy and incomplete experimental data, metaheuristics methods are utilized to find the optimal values of these parameters by minimizing the difference between both data. In this paper, a new optimization method is introduced for the biological model parameter estimation. The proposed method is developed based on the hybridization of Artificial Bee Colony (ABC) and Differential Evolution (DE) methods. In general, this method employs the evolutionary operations of the DE method to improve the neighboring searching strategy via the ABC method. The accuracy and reliability in estimating the parameters are demonstrated by using a model of lactose feedback regulation in a bacterial cell. The results showed that the performance of the proposed method has outperformed the existing methods.*
Keywords: Artificial bee colony, Differential evolution, Hybrid optimization, Parameter estimation, Biological models

1. **Introduction.** The key issue of systems biology is the development of the models that replicate the actual processes occurring within the cells. These models are commonly developed using ordinary differential equations (ODEs) to observe the concentration change of specific biochemical products. In order to construct an accurate model, the experimental data are needed to ensure the reliability of the model prediction. The models usually depend on a set of parameters to physically simulate the biological processes. These may include the reaction rates, affinity constants and kinetic velocity. However, most of these parameters are difficult to be extracted from the experimental data directly [1]. Thus, model fitting techniques are used to estimate these parameters by minimizing the difference between the experimental data and the model prediction. As the experimental data are frequently incomplete and noisy, it is a challenging task to estimate these parameters accurately [1-3].

In this paper, a new hybrid optimization method is introduced to estimate the parameters in the biological models. The proposed method, Differential Evolutionary Bee Colony (DEBCO) method, is developed to improve the neighboring searching strategy via the standard Artificial Bee Colony (ABC) method [4] using the combinatorial evolutionary operations in the Differential Evolution (DE) method [5]. The present method employs the differential mutation and crossover operations to enhance the searching capability performed by the employed bees in the ABC method. The performance of the

proposed DEBCO method for parameter estimation problem is evaluated using a complex biological model, namely the feedback regulation of lactose operons by bacterial cell [6]. The results showed that the proposed method was capable of finding the parameters with significant accuracy and acceptable amount of computational time compared with those produced by the existing methods.

2. **Problem Formulation.** The parameter estimation problem can be generalized as an optimization problem. This problem is aimed to find optimal parameters that can minimize the difference between the model prediction and its corresponding experimental data. The system is formulated using ODEs as follows:

$$\begin{cases} \frac{dS}{dt} = g(S, u, p, t_s) \\ S_{t_s} = S_0 \\ y = h(S(u, p), t_s) + \varepsilon \end{cases} \tag{1}$$

where S is the model state variable of the system, u is the input signal, $p = \{p_1, p_2, p_3, \ldots, p_N\}$ is the set of parameters, t_s is the time sampling and ε is the measurement noise. These equations show that the time-derivative of model state variable, $\frac{dS}{dt}$, is represented by a nonlinear function, g. The observable model output, y, is represented as another nonlinear function, h, that is formed by the model state variable of the corresponding time sampling and is associated with the measurement noise. Therefore, the parameters can be estimated by minimizing the difference between these data as follows:

$$f(p) = \arg\min \sum_{n=1}^{N} \sum_{m=1}^{M} \left(\hat{y}_m - y_m(p_n)\right)^2 \tag{2}$$

where p is the solution (parameter set), M and N are the sampling time and total number of parameters to be estimated, respectively, and $\hat{y}$ is the experimental data.

3. **Differential Evolutionary Bee Colony Optimization Method.** In this paper, a new hybrid optimization method called DEBCO is introduced. The method is basically an improvement of the neighboring searching strategy of ABC method using evolutionary operations adopted from DE method. Initially, the method starts with a randomly distributed population of solutions. Each ith solution is composed of a set of vectors that represent the parameters of the model, $x_i = \{p_{i1}, p_{i2}, p_{i3}, \ldots, p_{iN}\}$, where N is the total number of parameters. The fitness of each ith solution, $f(x_i)$, is evaluated. As the iteration progresses, the method divides the population into two main groups: employed and onlooker solutions. In a standard ABC method, the employed solutions are subjected to neighborhood improvement. This is performed by comparing the currently observed solution with its neighboring solutions [4]. For the ith employed solution, the searching process is executed based on the following equation:

$$x_i = x_i + r1(x_i - x_j) \tag{3}$$

where x_j is the jth neighboring solution of the current solution x_i, and $r1$ is a uniformly distributed random number between -1 and 1. In the present method, the employed solutions are subjected to evolutionary improvement before the neighboring searching process is performed. The evolutionary improvement involves the use of evolutionary operations from DE method. Firstly, each ith solution is submitted for a mutation operation to produce trivial solution, v_i. The process is carried out using the following rule [5]:

$$v_i = \begin{cases} x_{\min} + MR(x_i - x_{\min}) & \text{if } r2 \geq 0.7 \\ x_i & \text{Otherwise} \end{cases} \tag{4}$$

where $x_{\min}$ is the vector of the current global best solution, MR is the predefined mutation rate and $r2$ is a uniformly distributed random number between 0 and 1. After that, the

trivial solution is subjected for a crossover operation to produce the offspring solution. The operation is performed as follows [5]:

$$x_{i(t+1)} = \begin{cases} v_{i(t)} & \text{if } r3 \leq CR \\ x_{i(t)} & \text{if } r3 > CR \end{cases} \tag{5}$$

where t is the iteration step, CR is the predefined crossover rate and $r3$ is a uniformly distributed random number between 0 and 1. As the offspring solution is generated, a selection operation is executed between the parent solution and its offspring counterpart, using the following rule:

$$x_{i(t+1)} = \begin{cases} x_{i(t+1)} & \text{if } f(x_{i(t+1)}) \leq f(x_{i(t)}) \\ x_{i(t)} & \text{if } f(x_{i(t+1)}) > f(x_{i(t)}) \end{cases} \tag{6}$$

Therefore, the solutions with better fitness are retained in the population while those with least substantial fitness are disregarded [5].

Conversely, the onlooker solutions are subjected to the neighboring improvement by probability. The probability of each ith solution, $prob_i \Pr_1 \Pr_i$, is calculated as follows [4]:

$$\Pr_i = \frac{fit_i}{\sum_{i=1}^{NP} fit_i} \tag{7}$$

where fit_i is the relative fitness of ith solution and NP is the population size. The relative fitness is computed by using the following rule [4]:

$$fit_{i(t+1)} = \begin{cases} \frac{1}{1+f(x_{i(t+1)})} & \text{if } f(x_{i(t+1)}) \geq 0 \\ 1 + abs(f(x_{i(t+1)})) & \text{if } f(x_{i(t+1)}) < 0 \end{cases} \tag{8}$$

As for the solutions that are subjected under this rule, the neighboring improvement is performed. A greedy selection operation is executed to choose between the original solution and its onlooker counterpart in terms of fitness values. Lastly, solutions that show lack of improvement within the given number of iterations are submitted for repopulation. The process basically relocates these solutions to a new location randomly within the searching space by following equation [5]:

$$x_{i(t+1)} = x_{\min} + r4(x_{\max} - x_{\min}) \tag{9}$$

where $x_{\max}$ and $x_{\min}$ are the maximum and minimum boundaries of the searching space, respectively, while $r4$ is a uniformly distributed random number between 0 and 1. The procedure is repeated until the maximum number of iterations is reached. At this point, the solution with the best fitness value is selected as the global best solution.

4. **Results and Discussion.**

To demonstrate the effectiveness of the proposed method for a more complex problem, a biological model consisting of 12 parameters is used. The model is constructed by using [5] to simulate the feedback regulation of lactose operon in *Escherichia coli* bacterium. Fundamentally, this model presents the cellular metabolism of the bacterium in the absence of glucose but with the availability of external lactose. The lactose is transported into the cell by the permease, which later will be broken down into allolactose by the β-galactosidase enzyme. The allolactose feeds back to bind with the lactose repressor and permits the transcription process by messenger ribonucleic acid (mRNA). The model is constructed by the following equations [6]:

$$\frac{dA}{dt} = k_g - (k_m + k_{gm})A + \frac{B}{k_{tm}} \tag{10}$$

$$\frac{dB}{dt} = (k_m + k_{gb})C + \frac{D}{k_{tb}} \tag{11}$$

$$\frac{dF}{dt} = k_a A \frac{E}{k_l + E} - k_b C \frac{F}{k_l + F} = (k_m + k_{ga})F \tag{12}$$

$$\frac{dG}{dt} = (k_m + k_{gp})G + \frac{H}{k_{tb} + k_{tp}} \tag{13}$$

where A, B, C, D, E, F, G, and H are mRNA, partial mRNA, β-galactosidase, partial β-galactosidase, lactose internal, allolactose, permease, and partial permease concentrations, respectively. In total, a number of 12 parameters are required to be estimated. The experimental data of this model is generated *in silico*, in which the time-series data is superimposed with white Gaussian noise [7].

The performance of the proposed DEBCO method is compared with those by the standard PSO, GA, ABC, and two newly proposed methods, namely Artificial Bee Colony Differential Evolution (ABCDE) [9] and Local Evolutionary PSO [10] methods. The population sizes are set to 20, 40, and 60. The maximum numbers of iterations are fixed to 100, 500, and 1000. For PSO method, the inertia weight is tuned to 0.5, and the self-exploitation and swarm-exploration rates are both set to 3.5 [8]. For the GA method, the mutation and crossover rate are set to 0.1 and 1.5. All methods are executed for 100 independent times and the mean best fitness values and the standard deviations are computed to demonstrate the consistency of the methods in finding the best solutions. Table 1 shows the comparison of the mean best parameter values found by the respective methods and the corresponding best fitness values. It is clearly presented that the proposed DEBCO method has obtained better parameter values with the overall best fitness values. Figure 1 illustrates the convergence behavior of the methods using population size of 60. This figure showed that the proposed DEBCO method is managed to converge relatively faster than other methods. These results suggested that the proposed DEBCO method is capable of estimating parameters accurately and reliably using the noisy and incomplete experimental data.

5. **Conclusion.** In this paper, a new hybrid optimization method, named DEBCO, is proposed. The aim of this contribution is to introduce the exploitation of information among neighboring solutions so that this information is beneficially utilized to find a better solution through the iterations. The effectiveness of the proposed DEBCO method is measured by solving the nonlinear parameter estimation problem in the biological model.

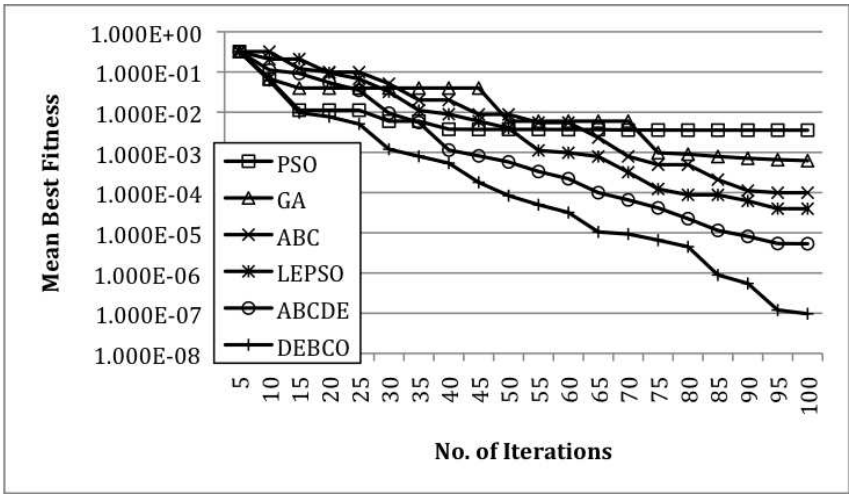

FIGURE 1. Convergence behavior of the optimization methods (for population size 60)

TABLE 1. Mean best fitness value and standard deviation (in brackets) found by the methods

Method	**No. of Iterations**	**Population Size**		
		20	40	60
PSO	100	1.39×10^{1} (5.81×10^{1})	9.97×10^{-1} (2.12×10^{-1})	5.56×10^{-2} (3.11×10^{-2})
	500	3.11×10^{-1} (2.08×10^{-1})	5.75×10^{-1} (4.37×10^{-1})	2.28×10^{-2} (4.45×10^{-2})
	1000	1.01×10^{-1} (2.11×10^{-1})	3.75×10^{-1} (2.81×10^{-1})	7.88×10^{-3} (4.34×10^{-3})
GA	100	1.42×10^{-1} (2.20×10^{-1})	7.22×10^{-2} (3.12×10^{-2})	8.22×10^{-2} (4.43×10^{-2})
	500	7.72×10^{-1} (3.12×10^{-1})	4.75×10^{-2} (4.01×10^{-2})	5.58×10^{-2} (4.00×10^{-2})
	1000	3.40×10^{-1} (5.11×10^{-1})	2.12×10^{-2} (3.36×10^{-2})	7.71×10^{-3} (2.09×10^{-3})
ABC	100	5.13×10^{-1} (3.75×10^{-1})	5.21×10^{-2} (4.10×10^{-2})	4.47×10^{-2} (3.25×10^{-2})
	500	3.39×10^{-2} (2.99×10^{-2})	3.55×10^{-2} (2.22×10^{-2})	1.10×10^{-2} (2.79×10^{-2})
	1000	1.01×10^{-2} (2.95×10^{-2})	1.25×10^{-2} (2.57×10^{-2})	7.26×10^{-3} (3.77×10^{-3})
LEPSO	100	5.71×10^{-2} (1.12×10^{-2})	8.91×10^{-3} (1.12×10^{-3})	6.01×10^{-3} (3.81×10^{-3})
	500	7.28×10^{-2} (5.09×10^{-2})	4.40×10^{-3} (2.32×10^{-3})	4.56×10^{-3} (3.35×10^{-3})
	1000	3.78×10^{-3} (2.01×10^{-2})	1.01×10^{-3} (1.78×10^{-3})	8.21×10^{-4} (5.55×10^{-4})
ABCDE	100	3.12×10^{-3} (2.08×10^{-3})	4.41×10^{-4} (3.77×10^{-4})	7.22×10^{-5} (5.51×10^{-5})
	500	1.33×10^{-3} (5.18×10^{-3})	1.91×10^{-4} (2.61×10^{-4})	3.01×10^{-5} (1.22×10^{-5})
	1000	9.01×10^{-4} (4.22×10^{-4})	9.44×10^{-5} (3.17×10^{-5})	5.56×10^{-6} (1.01×10^{-6})
DEBCO	100	$\mathbf{1.71 \times 10^{-5}}$ ($\mathbf{1.85 \times 10^{-5}}$)	$\mathbf{2.95 \times 10^{-6}}$ ($\mathbf{1.10 \times 10^{-6}}$)	$\mathbf{4.91 \times 10^{-8}}$ ($\mathbf{1.01 \times 10^{-8}}$)
	500	$\mathbf{2.13 \times 10^{-6}}$ ($\mathbf{4.55 \times 10^{-6}}$)	$\mathbf{8.99 \times 10^{-7}}$ ($\mathbf{3.21 \times 10^{-7}}$)	$\mathbf{9.07 \times 10^{-9}}$ ($\mathbf{3.81 \times 10^{-9}}$)
	1000	$\mathbf{1.88 \times 10^{-7}}$ ($\mathbf{3.35 \times 10^{-7}}$)	$\mathbf{3.35 \times 10^{-8}}$ ($\mathbf{1.23 \times 10^{-8}}$)	$\mathbf{5.77 \times 10^{-9}}$ ($\mathbf{2.71 \times 10^{-9}}$)

The results showed that the convergence speed of the DEBCO method has outperformed those produced by the existing methods. Moreover, the convergence behavior of the method suggests that the DEBCO method is also capable of escaping the local optima more effectively. In the future, the proposed method may be practical to be implemented for stochastic and spatial temporal modeling as the method can utilize the computational cost consumption more effectively compared with the existing methods. This is important as such modeling techniques require more extensive computational cost than kinetic modeling used in this paper.

REFERENCES

[1] J. Sun, J. Garibaldi and C. Hodgman, Parameter estimation using meta-heuristics in systems biology: A comprehensive review, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol.9, no.1, pp.185-202, 2012.

[2] T. Miyake, T. Matsumoto, T. Imamura and Z. Zhang, Estimation of facial expression from its change in time, *ICIC Express Letters, Part B: Applications*, vol.2, no.3, pp.641-645, 2011.

[3] A. Abdullah, S. Deris, M. Mohamad and S. Hashim, A new hybrid firefly algorithm for complex and nonlinear problem, *Distributed Computing and Artificial Intelligence*, vol.151, pp.673-680, 2012.

[4] D. Karaboga and B. Basturk, On the performance of artificial bee colony (ABC) algorithm, *Applied Soft Computing*, vol.8, no.1, pp.687-697, 2008.

[5] 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, pp.341-359, 1997.

[6] N. Yildirim and M. C. Mackey, Feedback regulation in the lactose operon: A mathematical modeling study and comparison with experimental data, *Biophysical Journal*, vol.84, no.5, pp.2841-2851, 2003.

[7] G. Lillacci and M. Khammash, Parameter estimation and model selection in computational biology, *PLoS Computational Biology*, vol.6, no.3, 2010.

[8] S. Das, A. Abraham and A. Konar, Particle swarm optimization and differential evolution algorithms: Technical analysis, applications and hybridization perspectives, *Advances of Computational Intelligence in Industrial Systems*, vol.116, pp.1-38, 2008.

[9] A. Abraham, R. K. Jatoth and A. Rajasekhar, Hybrid differential artificial bee colony algorithm, *Journal of Computational and Theoretical Nanoscience*, vol.9, pp.1-9, 2012.

[10] A. Abdullah, S. Deris, S. Z. M. Hashim, M. S. Mohamad and S. N. V. Arjunan, An improved local best searching in particle swarm optimization using differential evolution, *International Conference on Hybrid Intelligent Systems*, pp.115-120, 2011.