Estimation of Algae Growth Model Parameters by a Double Layer Genetic Algorithm

ARTORN NOKKAEW
Institute for Innovative Learning
Mahidol University
Nakorn Prathom 73170
THAILAND
krooart@gmail.com

BUSAYAMAS PIMPUNCHAT
Industrial Mathematics Research Unit and
Department of Mathematics
Faculty of Science, King Mongkut's Institute of Technology Ladkrabang
Bangkok 10520
and
Centre of Excellence in Mathematics, CHE
Bangkok 10400
THAILAND
knbusaya@kmitl.ac.th

CHARIN MODCHANG
Department of Physics
Faculty of Science, Mahidol University
Bangkok 10400
THAILAND
sccharin@mahidol.ac.th

SOMKID AMORNSAMANKUL
Department of Mathematics
Faculty of Science, Mahidol University
Bangkok 10400
and
Centre of Excellence in Mathematics, CHE
Bangkok 10400
THAILAND
somkid.amo@mahidol.ac.th

WANNAPONG TRIAMPO*
Department of Physics
Faculty of Science, Mahidol University
Bangkok 10400
and
Institute for Innovative Learning
Mahidol University
Nakorn Prathom 73170
and
Centre of Excellence in Mathematics, CHE
Bangkok 10400
THAILAND
scwtr@mahidol.ac.th
wtriampo@gmail.com

DARAPOND TRIAMPO
Department of Chemistry and Center of Excellence for Innovation in Chemistry
Faculty of Science, Mahidol University
Bangkok 10400
THAILAND
scdar@mahidol.ac.th

Abstract: - This paper presents a double layer genetic algorithm (DLGA) to improve performance of the information-constrained parameter estimations. When a simple genetic algorithm (SGA) fails, a DLGA is applied to the optimization problem in which the initial condition is missing. In this study, a DLGA is specifically designed. The two layers of the SGA serve different purposes. The two optimizations are applied separately but sequentially. The first layer determines the average value of a state variable as its derivative is zero. The knowledge from the first layer is utilized to guide search in the second layer. The second layer uses the obtained average to optimize model parameters. To construct a fitness function for the second layer, the relative derivative function of the average is combined into the fitness function of the ordinary least square problem as a value control. The result shows that the DLGA has better performance. When missing an initial condition, the DLGA provides more consistent numerical values for model parameters. Also, simulation produced by DLGA is more reasonable values than those produced by the SGA.

*Corresponding author: Wannapong Triampo. contact: scwtr@mahidol.ac.th and wtriampo@gmail.com

Key-Words: - Algae growth, genetic algorithms, initial values problem, optimization, ordinary differential equations, parameter estimation.

1 Introduction

To study the growth mechanism of algae, ordinary differential equations have been proposed and utilized [1]-[6]. It leads to optimization problems in which model parameters or constants need to be estimated. A number of optimization methods are available to generate feasible numerical values of model constants [7]. However, conventional methods often have difficulties with complex or undifferentiated problems. If possible, traditional optimization algorithms, such as gradient based methods or direct search methods [8,9], have limitations of reaching a global optimum. Also, the efficiency of these methods depends upon an initial point. Unless we know proper initial values, the traditional methods are inefficient.

To overcome these limitations, global optimization algorithms have been used. A genetic algorithm (GA) is a stochastic global optimization algorithm based on the evolutionary process [10]. Inspired by biological evolution, GA mimics natural selection and sexual reproduction to evolve solutions. GAs have been successfully applied to estimate constants of various models, for example [11]-[13].

However, their performances have been considerably limited by some problems, such as premature convergence problems.

To improve simple genetic algorithm (SGA) performance, more heuristic methods that multi SGA is implemented has been utilized [14]-[20]. They were produced with various purposes and names. Chang et al. [14] proposed two layers of SGA, named TPSPGA, in order to prevent local optimum trap. Not only to prevent premature convergence, Crevecoeur et al. [15] applied two layers of SGA to reduce computational burden for a complex problem. They named their multi-layers of SGA as "2LGA". Other variety names of multi-layers of SGA are such as two-phase genetic algorithm (TPGA) [16], double layer genetic algorithm [17], two-phase genetic local search algorithm [18], double genetic algorithm (DGA) [19] and Meta Genetic algorithm [20]. Multi-layers of SGA is successfully applied to number of problems: multi-model functions [16]; path planning [17]; scheduling problem [14], [18]-[20]; electromagnetic optimization [15] and multi-objective problems [14].

The uses of the multi layers of SGA are objective-dependent. The SGA in the two layers is designed purposely. Several frameworks have been proposed and developed. Normally, the first layer is used as global optimization to narrow the search space in the second optimization [14,16,18]. Serving as local optimization, the second layer uses this promising search area to find an optimum. Namely, a set of elite solutions of the first phase is utilized to construct the initial population of the second phase [17,18]. Not only narrowing a searching space, the multi layers of SGA is applied as a meta-GA for choosing a proper set of SGA parameters and operators. For this purpose, the function of the first SGA is to provide the quasi-best SGA parameter vector values for particular instances to the second SGA [20]. The multi layers of SGA also were designed to work with models in an optimization.

The multi layers of SGA design is user-oriented design. Lin-Yu and Shih-Chieh [18] utilized the same genetic algorithm in both layers with different initial populations. Jin and Dongyong [17] and Barrios et al. [19] designed two SGA with different fitness evaluation functions. Representation, fitness, crossover operator were designed differently [19]. Not only adjusting genetic parameters and operators, Crevecoeur et al. [15] used two different models in the two layers. Working with time-demanding task, the multi layers of SGA implement two models, computationally demanding fine model and approximate model, in the optimization procedure [15]. An approximated model, which can be meta-model or theory-based model, is generated and optimized in the first layer. The model adjustment enables acceleration of optimization procedure.

2 Problem Description

To study algae growth, models proposed by [2,3] are adopted. The model described the evolutions of the ambient concentration of limiting nutrient, cell quota of limiting nutrient and growth of algae in continuous culture. In the proposed model, the three state equations are:

$$\frac{dS}{dt} = D(S_0 - S) - \rho X, \quad (1)$$

$$\frac{\mathrm{dQ}}{\mathrm{dt}} = \rho\text{-}\mu\mathrm{Q}, \quad (2)$$

$$\frac{dX}{dt} = \mu X - DX, \quad (3)$$

where S is the ambient concentration of limiting nutrient, Q is the cell quota of limiting nutrient, X is the cell density, S_0 is the influent concentration of limiting nutrient, D is the dilution rate, μ is the growth rate and ρ is defined as the net transfer of limiting nutrient across an average cell's membrane per unit time.

In [3], the model is made mathematically complete with the specification of two state dependent rate function, one for uptake rate and the other for growth rate. Therefore uptake rate is described by a Michaelis-Menten kinetics as

$$\rho = \rho_m \left(\frac{S}{K_\rho + S}\right) \quad (4)$$

where ρ_m is the maximum uptake rate and K_ρ is the half saturation constant for substrate uptake, namely the substrate concentration supporting an uptake rate one-half the maximum rate.

Note that the uptake rate of the limiting nutrient of the model depends upon the external concentration of the limiting nutrient (S).

The growth rate is based on a Droop function [2]:

$$\mu = \mu_\mathrm{m} \left(1 - \frac{\mathrm{q_m}}{\mathrm{Q}}\right) \quad (5)$$

where μ_m is the maximum growth rate and q_m is the subsistence quota in which algae do not grow under this threshold.

In this study, the model is applied to describe the growth of algae in a batch culture system. Therefore, the dilution rate of the continuous culture (D) is zero. In addition, this study takes account of the influence of the internal nutrient on uptake rate. Consequently, uptake rates are affected by external nutrients as well as internal nutrients. Uptake rate is represented as

$$\rho = \rho_m \left(\frac{SQ}{K_\rho + SQ}\right) \quad (6).$$

The model used in this study is composed of three ordinary differential equations hinging on the cell quota approach [2,4] as follows:

$$\frac{dS}{dt} = -\rho_m \left(\frac{SQ}{K_\rho + SQ}\right) X, \quad (7)$$

$$\frac{\mathrm{dQ}}{\mathrm{dt}} = \rho_\mathrm{m} \left(\frac{\mathrm{SQ}}{\mathrm{K_\rho + SQ}}\right) \text{-} \mu_\mathrm{m} \left(1 \text{-} \frac{\mathrm{q_m}}{\mathrm{Q}}\right) \mathrm{Q}, \quad (8)$$

$$\frac{dX}{dt} = \mu_m \left(1 - \frac{q_m}{Q}\right) X, \quad (9)$$

where S is the ambient concentration of the limiting nutrient (m·g·L^{-1}), Q is the cell quota of the limiting nutrient (dimensionless), X is the cell density (m·g·L^{-1}), ρ_m is the maximum uptake rate (hour^{-1}) and K_ρ is its associated half saturation constant (m·g·L^{-1}), namely the substrate concentration supporting an uptake rate one-half the maximum rate, μ_m is the maximum growth rate (hour^{-1}); q_m is the subsistence quota (dimensionless) in which algae do not grow under this threshold.

In this case, the three differential equations are evolution equations specifying how the system will evolve with time in which the specified values at initial, called the initial conditions, are required for solving model. Specifically, initial conditions include S, Q and X at time zero. In this study, only the data of substrate (S) and Biomass (X) are available as reported in [21]. The problem becomes a fitting problem of an ordinary differential system in which an initial condition of a state variable, Q in this case, is missing. The model is fitted to the experimental data. Fitting the model by the least square method involves solving an initial value problem. Normally, without knowing the initial condition, a possible value will be assumed. Without the initial condition, alternatively, an SGA still can be applied. However, when implementing an SGA, we found that the SGA failed to consistently identify the value of the parameters of the missing initial condition problem.

To improve the performance of an SGA for fitting problems without initial conditions, this paper designs and proposes a multi layers of SGA to solve such a specific problem.

3 Simple Genetic Algorithm (SGA)

Genetic algorithm is an optimization technique that was developed by Holland and his colleagues in 1975. Simple genetic algorithm (SGA) produces optimum solutions by mimicking two biological mechanisms: natural selection and chromosome encoding. In nature, natural selection determines which individuals in a population survive or die. Through natural selection, living organisms with

greater fitness to the environment have a greater probability of surviving and reproducing. Organisms can evolve by sexual reproduction. Sexual reproduction introduces variation into the next generation population by a combination of parent chromosomes. However, errors can occur naturally during replication in the reproduction process, resulting in a mutation of offspring whose characteristics more or less shift from their parents. Both natural selection and sexual reproduction, including mutation, allow organisms to evolve as generations pass.

To adapt the process of biological evolution to a mathematical problem, candidate solutions have to be represented in the form of an array of variables. Borrowing a genetic term, the array is called "chromosomes", while each variable is called "gene".

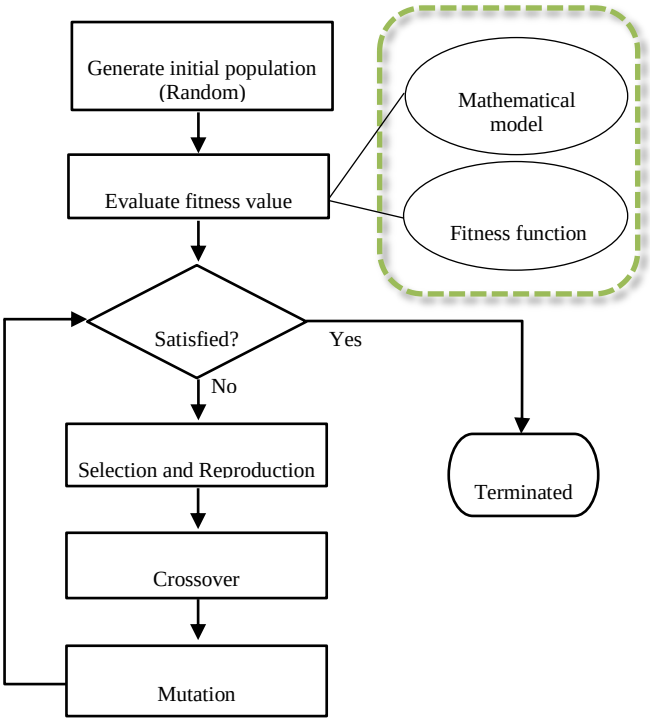

Fig. 1 a flowchart of a simple genetic algorithm (the highlighted dash-frame represents the adapted parts in a double genetic algorithm)

In an SGA, solutions evolve through three main operators: selection, crossover and mutation. The algorithm of an SGA is depicted in Fig. 1. Based on the 'Survival of the fittest' principle, the better fitness chromosomes are, the bigger chance to be selected to survive they have. It is a bias random process. The selection operator selects chromosomes from the new offspring according to the fitness. The fitness is evaluated by a mathematical function, called the fitness function. Applying it to the parameter determination problem, the fitness function works cooperatively with the mathematical model being studied (see Fig. 1). Crossover is a vital operator of the SGA process. It produces a chromosome by exchanging segments of a pair of selected parent chromosomes. It evolves new solutions by exploiting the profit material from the previous search. Therefore, the proper segments of a pair of parent are inherited by the descendants. Lastly, mutation introduces variety into populations. Mutation serves the crucial role of exploration. Without inheritance, mutation produces new members by altering the value of genes randomly. Consequently, unlike crossover, the offspring is different from the parents. Normally, the chance of mutation is set small. The SGA mechanism is an iterative procedure, and through a number of generations, it can evolve an optimal solution. The iterative algorithm to evolve a solution for a problem on the computer is summarized in Fig. 1.

4 Double Level Genetic Algorithm (DLGA)

This study proposes a double layer genetic algorithm (DLGA) to information-constrained parameter estimation problem. Both two levels apply the same SGA. To enable SGA to identify the model parameters with a missing initial condition of a state variable, a DLGA is proposed as an alternative approach. A DLGA is designed to empower an SGA to estimate the possible values of the parameters. The proposed approach is composed of two phases of an SGA. Each layer corresponds to different aims. The mathematical model and the fitness function of each layer are designed differently to serve their intended purposes. For a parameter estimation problem, fitting models are designed differently in two layers. Firstly, a fitting model F is constructed based on theory. The model normally composes of a set of differential equations, expressed as following:

$$D_t \boldsymbol{x} = F(\boldsymbol{x}, \boldsymbol{t}), \tag{10}$$

where $\boldsymbol{x}$ is a vector of state variables and $\boldsymbol{t}$ is a vector of times. The second fitting model is created by approximating the original model. The F model is simplified by setting $D_t \boldsymbol{x_i} = 0$ where $\boldsymbol{x_i}$ data is deficient. Consequently, number of state variables x

of the simplified model decreases. The approximated model G can be expressed.

$$D_t \boldsymbol{y} = G(\boldsymbol{y}, \boldsymbol{t}); \quad \boldsymbol{y} \subset \boldsymbol{x} \; but \; \boldsymbol{x} \neq \boldsymbol{y}, \tag{11}$$

where $\boldsymbol{y}$ is a vector of state variables, which is a subset of $\boldsymbol{x}$ but $\boldsymbol{y}$ is not equal to $\boldsymbol{x}$, and $\boldsymbol{t}$ is a vector of times.

To apply DLGA, see Fig. 2, the first layer utilizes an SGA to generates a control data by using approximated model G. Then, the obtained control data will be used as knowledge for second search of DLGA in the second layer. The control data, constant $\boldsymbol{x_i}$ in this case, is used to clue the second optimization. To do so, the optimization evaluation function is adjusted. In the second layer, the obtained constant $\boldsymbol{x_i}$ is added into an optimization criterion in form of average deviation function. Consequently, a fitness function of the second layer composes of ordinary least squares of errors and deviation function of mean value of the missing state variable. Using new evaluation function, parameters of model F are estimated.

5 Implementation and Results

This section presents implementation and consequents of DLGA for parameter estimation of differential equations when an initial condition is missing. The same SGA is applied in two layers. In the first layer, approximated model is constructed. Cooperating with approximated model, the first SGA generates control data. The control data is utilized in the second layer. It is combined into fitness function to guide search of SGA with original model in the second layer.

5.1 The first layer

The first layer of DLGA serves for identifying the state variable constant when a state variable is assumed to have no change. The obtained constant is defined as an average of the state variable. In the first layer, the model, the equation (7)-(9), is simplified. Without knowing the proper value of the cell quota at time zero, state variable Q is assumed to be constant. Namely, the derivative of cell quota (Q), (8), is set at zero.

$$\frac{\mathrm{dQ}}{\mathrm{dt}} = \rho_m \left(\frac{SQ}{K_\rho + SQ}\right) - \mu_m \left(1 - \frac{q_m}{Q}\right) \cdot Q = 0 \tag{12}$$

$$\mu_m \left(1 - \frac{q_m}{Q}\right) = \rho_m \left(\frac{SQ}{K_\rho + SQ}\right) \cdot \frac{1}{Q} \tag{13}$$

Substitute (13) in (9)

$$\frac{dX}{dt} = \rho_m \left(\frac{SQ}{K_\rho + SQ}\right) \cdot \frac{1}{Q} \cdot X \tag{14}$$

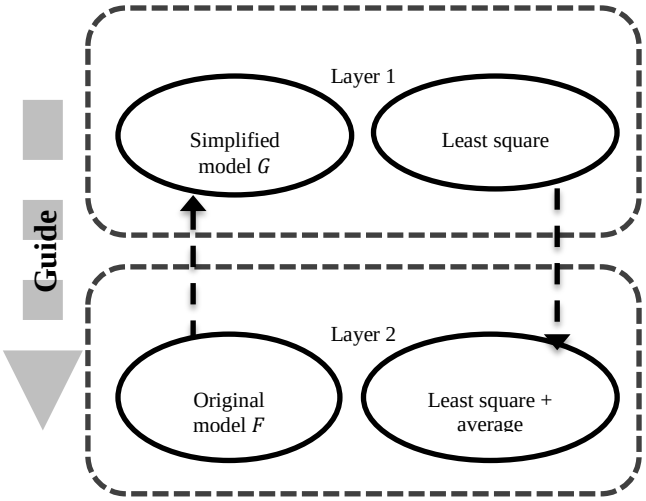

Fig. 2 interactions between two layers in a DLGA

An SGA is applied to estimate the constant Q, defined as an overall average of cell quotas. The SGA in the first layer uses the fitness function expressed in (15).

$$fitness \;\; function = \sum_{i=1}^{2} \sum_{j=1}^{10} \left(\frac{D_{\exp}^{i,j} - D_{sim}^{i,j}}{D_{\exp}^{i,j}} \right)^2 \tag{15}$$

where D_{exp} is experimental data and D_{sim} is the data obtained from solving differential equations (7) and (14). The equation (7) and (14) are fitted with 2 sets of data with 10 data points presented in [21]. The SGA in the first layer comes up with a consistent value of Q constant, 0.063.

5.2 The second layer

The second layer of DLGA serves for identifying parameters of original model. To do so, knowledge from the first layer is utilized. Second optimization

is guided by the constant obtained in the first layer. Not only the least square of errors function, the constant is used to evaluate the search. It becomes multi objective problems. In this study, the multiple objectives are scalarized into a single objective. Therefore, objective function, the equation (15), is adjusted. The deviation function of the average cell quota is combined into normal fitness function of least squares of errors. The new fitness function is obtained as follows:

$$fitness\ \ function = \sum_{i=1}^{2}\sum_{j=1}^{10}\left(\frac{D_{\exp}^{i,j} - D_{sim}^{i,j}}{D_{\exp}^{i,j}}\right)^2 + \frac{\left(\overline{Q}_{1stGA} - \overline{Q}_{2ndGA}\right)^2}{\overline{Q}_{1stGA}} \times w \quad (16)$$

where D_{exp} is the experimental data, D_{sim} is the data obtained from solving differential equations, $\overline{Q}_{1stGA}$ is the average cell quota obtained from the first layer, $\overline{Q}_{2ndGA}$is the average cell quota obtained from the current layer and w is the weight of average control. In this case, w is set of 10, in order to adjust its order.

The results from applying a DLGA are presented in Table 1.

Table 1 selected final results using DLGA

Parameter	Run 2	Run 3	Run 4
ρ_m	0.099107	0.067061	0.169138
K_ρ	41.537445	28.004112	71.129544
q_m	0.058901	0.058993	0.058865
μ_m	0.13026	0.132694	0.129224
$Q(t=0)$	0.061199	0.061216	0.061199
Fitness value	1.444976	1.450784	1.439973

.

Improvements can be gained by using DLGA. Quantitatively, a DLGA improves the optimal approach. Fig. 3 plots the final values of the obtained parameters. The left column of Fig. 3 plots the results from the SGA, while the right column shows the results of a DLGA. Except $Q(t=0)$, clearly, parameters obtained by the DLGA smoothly approaches a certain value as the fitness value drops, while smooth behavior can be found for ρ_m and K_ρ in the SGA.

See Fig. 3, plot of $Q(t=0)$ versus the fitness value shows a weak correlation. It implies that the initial condition of the cell quota does not influence approaching an optimum. However, the DLGA suggests a consistent initial condition for the cell quota (Q), around 0.061.

Qualitatively, a DLGA provides a reasonable cell quota evolution in a batch culture system [22]. Theoretically, in a static culture in which algae is added to a known amount of medium, algae requires a brief adaptation period (lag phase). Then, the number of algae increases exponentially (log phase) and continues to grow at a maximum rate until resources become limiting (transitional phase). During the growth of the algae, resources contained in each algae decrease, namely the cell quota drops. The decline in the cell quota leads to a drop in the growth rate until the cell quota reaches its minimum value, at which point there can be no further growth (stationary phase).

Cell quota trajectory obtained by a DLGA seems more reasonable than an SGA. In addition, it also corresponds to the growth curve of algae as presented in the upper row of Fig. 4. Cell quota obtained by a DLGA show a sharp increase of cell quota after the addition of algae to the culture medium, see Fig. 4 bottom right. Rapid uptake of algae results in increased cell quotas. The growth curve presents a slow increase in the amount of algae in this period. This behaviour is consistent to the lag phase which is a brief adaptation period of algae, see Fig. 4 upper right. Contrarily, increase of cell quota due to greedy consumption behaviour cannot be found in an SGA simulation, see Fig. 4 bottom left. The curve simulated by the SGA sharply and immediately decreases after adding the algae to the culture medium.

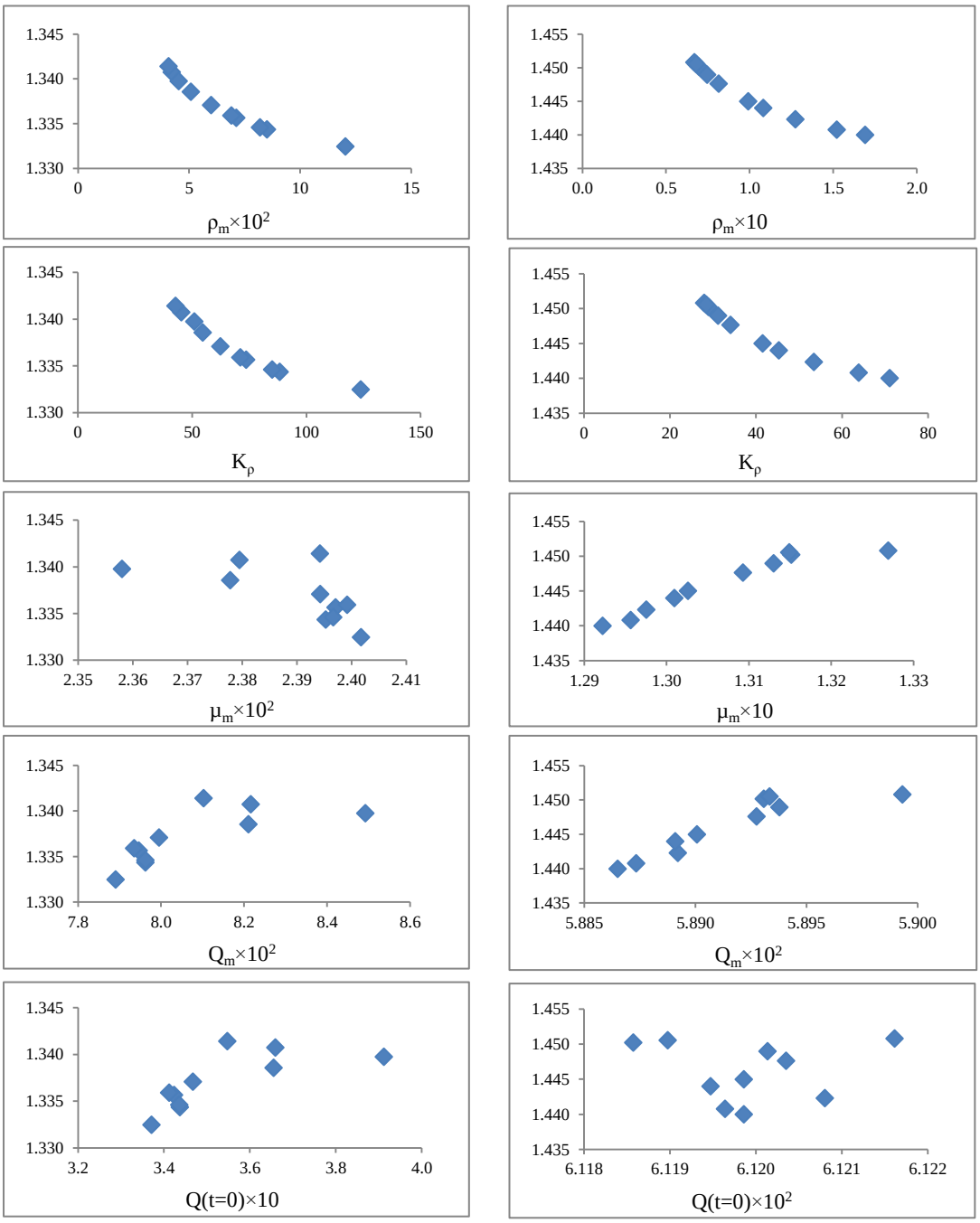

Fig. 3 plots of final parameter values obtained versus final fitness values. The left column presents the results from an SGA. The right presents the results from a DLGA

Not only is there the lag phase coincidence, a sharp decrease of cell quota simulated by the DLGA also agrees well with the dramatic growth of algae in the log phase. When the growth phase of algae population is dominant, cell quotas drop as biomass increases. Obtained by a DLGA, the cell quota trajectory dramatically decreases from 50 to 150 hours. The DLGA simulation presents the existence of the minimal cell quota for growth (q_m). It reaches stationary phase as growth rate becomes zero when the cell quota reaches its subsistent level.

Contrarily, cell quota curves simulated by SGA conflict to growth of biomass. The cell quota decreases dramatically from 0-50 hrs. The cell quota continues to decrease slowly until the limitation of the nutrients is reached. But, the growth of algae still continues as the cell quota drops nearly to its minimum value.

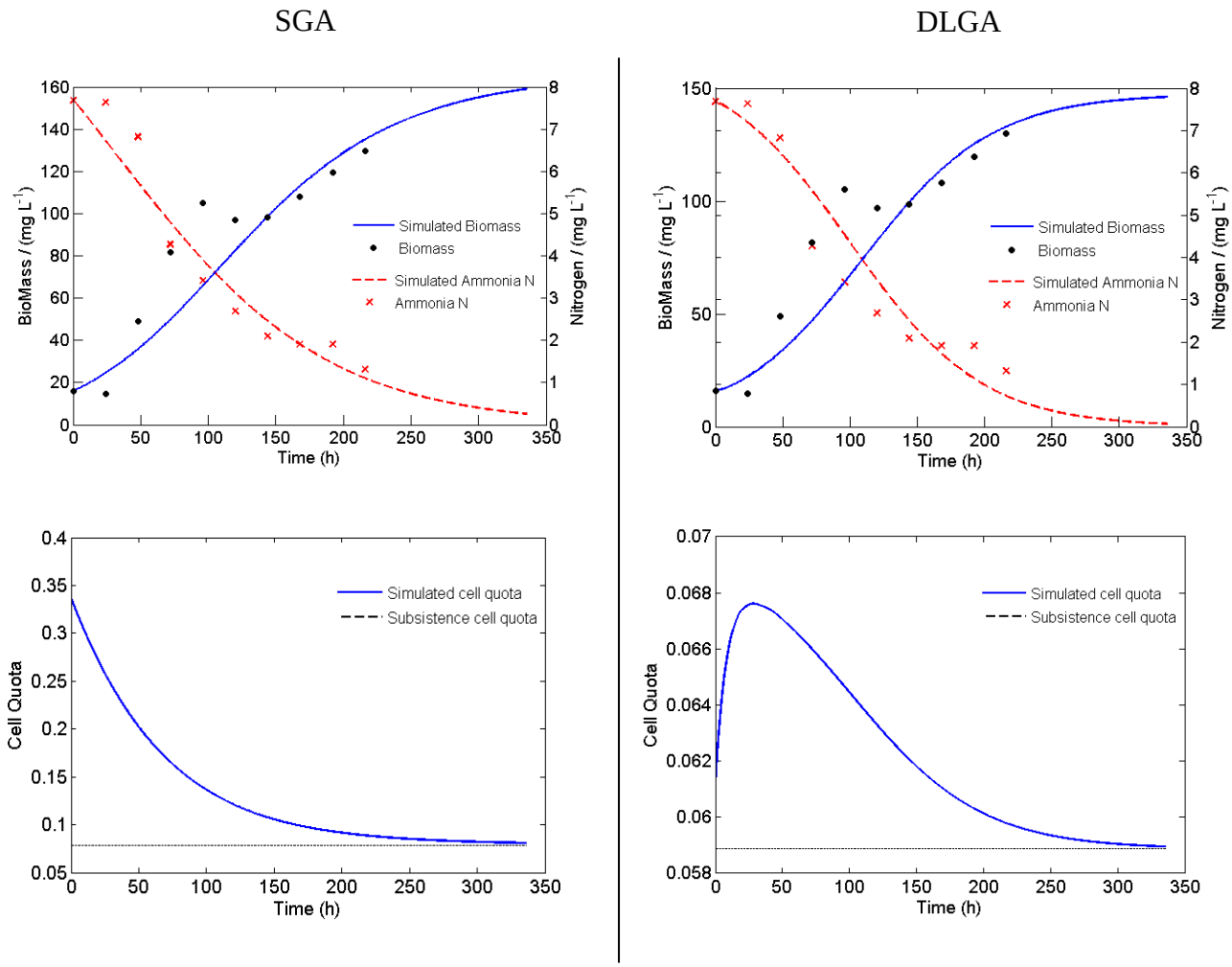

Fig. 4 plots of experimental data and simulated curves of Nitrogen concentration, biomass and cell quota versus time obtained from an SGA (left column) and a DLGA (right column)

6 Conclusion

When an SGA fails, a DLGA is utilized in optimization problems where the initial condition is missing. A DLGA is designed specifically. The two SGA have different aims and are implemented separately. The knowledge from the first layer is utilized to guide the search in the second layer of the DLGA. The knowledge is combined into the fitness function of ordinary least square problem to evaluate search in the second layer. The DLGA improves the performance of the SGA for fitting the ordinary differential equation model when the initial condition is missing. Numerical values estimated by the DLGA are more consistent. Also, simulation produced by the DLGA is more reasonable than the one produced by the SGA.

ACKNOWLEDGMENTS

The first author would like to thank The Institute for the Promotion of Teaching Science and Technology (IPST) for a scholarship. Also, this work is partially supported by Centre of Excellence in Mathematics, The Commission on Higher Education, Thailand.

References:

[1] Ruiz, J., Álvarez, P., Arbib, Z., Garrido, C., Barragán, J., Perales, J. A., Effect of Nitrogen and Phosphorus Concentration on Their Removal Kinetic in Treated Urban Wastewater by Chlorella Vulgaris, *International Journal of Phytoremediation*, Vol. 13, No. 9, 2011, pp. 884-896.

[2] Bernard, O., Hurdles and challenges for modelling and control of microalgae for CO 2 mitigation and biofuel production, *Journal of Process Control*, Vol. 21, No. 10, 2011, pp. 1378-1389.

[3] Burmaster, D. E., Chisholm, S. W., A comparison of two methods for measuring phosphate uptake by Monochrysis lutheri droop grown in continuous culture, *Journal of Experimental Marine Biology and Ecology*, Vol. 39, No. 2, 1979, pp. 187-202.

[4] Droop, M. R., The nutrient status of algal cells in continuous culture, *Journal of the Marine Biological Association of the U.K.*, Vol. 54, 1974, pp. 825-855.

[5] Caperon, J., Population Growth Response of Isochrysis Galbana to Nitrate Variation at Limiting Concentrations, *Ecology*, Vol. 49, No. 5, 1968, pp. 866-872.

[6] Lemesle, V., Mailleret, L., A Mechanistic Investigation of the Algae Growth "Droop" Model, *Acta Biotheoretica*, Vol. 56, 2008, pp. 87-102.

[7] Zwillinger D., Ed., *CRC Standard mathematical tables and formulas (30 ed.)*, CRC Press, 2003.

[8] Lewis, R. M., Torczon, V., Trosset, M. W., Direct search methods: then and now, *Journal of Computational and Applied Mathematics*, Vol. 124, 2000, pp. 191-207.

[9] Carson, Y., & Maria, A, Simulation optimization: Methods and applications, *29th conferance on Winter simulation*, 1997, pp. 118-126.

[10] Goldberg, D., *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wiley Publishing Company, 1989.

[11] Modchang, C., Triampo, W., Lenbury, Y., Mathematical modeling and application of genetic algorithm to parameter estimation in signal transduction: Trafficking and promiscuous coupling of G-protein coupled receptors, *Computers in Biology and Medicine*, Vol. 38, 2008, pp. 574-582.

[12] Leelawattanachai, J., Modchang, C., Triampo, W., Triampo, D., Lenbury, Y., Modeling and genetic algorithm optimization of early events in signal transduction via dynamics of G-protein-coupled receptors: Internalization consideration, *Applied Mathematics and Computation*, Vol. 207, No. 2, 2009, pp. 528-544.

[13] Roeva, O., Parameter Estimation of a Monod-Type Model Based on Genetic Algorithms and Sensitivity Analysis Large-Scale Scientific Computing, *Large-Scale Scientific Computing.* Vol. 4818, Lirkov I., *et al.*, Eds., Springer, 2008, pp. 601-608.

[14] Chang, Pei-Chann, Chen, Shih-Hsin, Lin, Kun-Lin, Two-phase sub population genetic algorithm for parallel machine-scheduling problem, *Expert Systems with Applications*, Vol. 29, No. 3, 2005, pp. 705-712.

[15] Crevecoeur, G., Sergeant, P., Dupre, L., Van de Walle, R., A Two-Level Genetic Algorithm for Electromagnetic Optimization, *IEEE Transactions on Magnetics*, Vol. 46, 2010, pp. 2585-2595.

[16] Yu-zhen, H., Li-shanf, K., Ai-min, Two-phase genetic algorithm applied in the optimization of multi-modal function, *Wuhan University Journal of Natural Sciences*, Vol. 8, No. 1B, 2003, pp. 259-264.

[17] Jin, L., Dongyong, Y., Path Planning Based on Double-Layer Genetic Algorithm, *Third International Conferenceon Natural Computation*, Haikou, Hainan, 2007, pp. 357-361.

[18] Lin-Yu, T., Shih-Chieh, C., Two-Phase Genetic Local Search Algorithm for the Multimode Resource-Constrained Project Scheduling Problem, *IEEE Transactions on Evolutionary Computation*, Vol. 13, No.4, 2009, pp. 848-857.

[19] Barrios, A., Ballestín, F., Valls, V., A double genetic algorithm for the MRCPSP/max, *Computers & Operations Research*, Vol. 38, No. 1, 2011, pp. 33-43.

[20] Borodin, D., Vreckem, B. V., Bruyn, W. D., Meta Genetic Algorithm for the Batch Scheduling Problem with A Few Permutation Solution Vectors. A Case Study, presented at *the IEEE Symposium Series on Computational Intelligence 2011*, Paris, April 11-15, 2011.

[21] Kim, J., Lingaraju, B. P., Rheaume, R., Lee, J.-Y., Siddiqui, K. F., Removal of Ammonia from Wastewater Effluent by Chlorella Vulgaris, *Tsinghua Science & Technology*, Vol. 15, No. 4, 2010, pp. 391-396.

[22] Lampert, W., Sommer, U., *Limnoecology*, 2nd ed. Oxford University Press Inc., 2007, pp. 50-79.