

## Research Article

# Tuning Genetic Algorithm Parameters to Improve Convergence Time

**Maria Angelova and Tania Pencheva**

*Institute of Biophysics and Biomedical Engineering, Bulgarian Academy of Sciences, 105 Acad. G. Bonchev Street, 1113 Sofia, Bulgaria*

Correspondence should be addressed to Tania Pencheva, [tania.pencheva@clbme.bas.bg](mailto:tania.pencheva@clbme.bas.bg)

Received 31 March 2011; Revised 17 June 2011; Accepted 5 July 2011

Academic Editor: Ahmet N. Palazoglu

Copyright © 2011 M. Angelova and T. Pencheva. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Fermentation processes by nature are complex, time-varying, and highly nonlinear. As dynamic systems their modeling and further high-quality control are a serious challenge. The conventional optimization methods cannot overcome the fermentation processes peculiarities and do not lead to a satisfying solution. As an alternative, genetic algorithms as a stochastic global optimization method can be applied. For the purpose of parameter identification of a fed-batch cultivation of *S. cerevisiae* altogether four kinds of simple and four kinds of multipopulation genetic algorithms have been considered. Each of them is characterized with a different sequence of implementation of main genetic operators, namely, selection, crossover, and mutation. The influence of the most important genetic algorithm parameters—generation gap, crossover, and mutation rates has—been investigated too. Among the considered genetic algorithm parameters, generation gap influences most significantly the algorithm convergence time, saving up to 40% of time without affecting the model accuracy.

## 1. Introduction

Fermentation processes (FP) are preferred and widely used in different branches of industry. The modeling and control of FP pose serious challenges as FP are complex, nonlinear dynamic systems with interdependent and time-varying process parameters. An important step for adequate modeling of nonlinear models of FP is the choice of a certain optimization procedure for model parameter identification. Different metaheuristic methods have been applied to surmount the parameter estimation difficulties [1–3]. Since the conventional optimization methods cannot overcome the limitations of FP [4], genetic algorithms (GAs), as a stochastic global optimization method, are quite promising. Among a number of searching tools, the genetic algorithms are one of the methods based on biological evolution and inspired by Darwin's theory of "survival of the fittest" [5]. GAs are directed random search techniques, based on the mechanics of natural selection and natural genetics. GAs find the global optimal solution in complex multidimensional search spaces simultaneously evaluating many points in the parameter space. They require only information concerning

the quality of the solution and do not require linearity in the parameters. GAs have been successfully applied in a variety of areas to solve many engineering and optimization problems [6–8]. Properties such as noise tolerance and ease of interfacing and hybridization make GA a suitable method for the identification of parameters in fermentation models [9–13].

Simple genetic algorithm (SGA) presented initially in Goldberg [5] searches for a global optimal solution using three main genetic operators in a sequence selection, crossover, and mutation. GAs work with a population of coded parameter set called "chromosome." Each of these artificial chromosomes is composed of binary strings (or genes) of certain length (number of binary digits). Each gene contains information for the corresponding parameter. Through selection chromosomes representing better possible solutions according to their own objective function values are chosen from the population. After the reproduction, crossover proceeds in order to form new offspring. Mutation is then applied with determinate probability. Even though selection and crossover effectively work, occasionally, a GA may lose some potentially useful information. That is why

mutation is needed to prevent falling of all solutions in the population into a local optimum of the solved problem. Then the algorithm evaluates the objective value of the individuals in the current population and according to that the new chromosome is created. The SGA is terminated when a certain number of generations is fulfilled. The basic multi-population genetic algorithm (MpGA) introduced in [5] has the same sequence of implementation of the main genetic operators as SGA. The difference in MpGA compared to SGA is the presence of many populations, called subpopulations [5, 14]. These subpopulations evolve independently from each other for a certain number of generations (isolation time), after that a number of individuals are distributed between the subpopulation.

For the purpose of this investigation, SGA and MpGA with standard sequence of genetic operators, namely selection, crossover, and mutation, are denoted, respectively, as SGA-SCM and MpGA-SCM. Many improved variations of the SGA and MpGA have been developed [9, 13, 15, 16]. Among them are the modified genetic algorithm with a sequence crossover, mutation, and selection [9], here denoted as SGA-CMS, and consequent modification of MpGA based on such exchange, here denoted as MpGA-CMS. In these algorithms selection operator has been processed after performing crossover and mutation. The main idea for such operators' sequence is to prevent the loss of reached good solution by either crossover or mutation or both operators. Both algorithms SGA-CMS and MpGA-CMS have been tuned for a parameter identification of *S. cerevisiae* fed-batch cultivation, improving the optimization capability of the algorithm and decreasing decision time. Obtained promising results applying SGA-CMS and MpGA-CMS encouraged more investigations concerning genetic operators' sequences to be performed in order further improvements of the algorithms to be found. Moreover, since the basic idea of GA is to imitate the mechanics of natural selection and genetics, one can make an analogy with the processes occurring in the nature, saying that the probability mutation to come first, and then crossover is comparable to the idea both processes to occur in a reverse order. Thus, following this line of investigation, also the following modifications are obtained [9]: SGA-SMC and MpGA-SMC—with sequence selection, mutation and crossover; SGA-MCS and MpGA-MCS—with sequence mutation, crossover, and selection, all the modifications of the SGA and MpGA developed in [5].

The elaboration of SGA-MCS is briefly presented below as the fastest genetic algorithm among all investigated in this study. In the beginning, the SGA-MCS generates a random population of chromosomes, that is, suitable solutions for the problem. In order to prevent the loss of reached good solution by either crossover or mutation or both operators, selection operator has been processed after performing of crossover and mutation [9]. The modification in SGA-MCS is that the individuals are reproduced processing firstly mutation, followed by crossover. The elements of chromosome are a bit changed when a newly created offspring mutates, after that the genes from parents combine to form a whole new chromosome during the crossover. After

the reproduction, the SGA-MCS calculates the objective function for the offspring and the best fitted individuals from the offspring are selected to replace the parents, according to their objective values. When a certain number of generations is fulfilled the SGA-MCS is terminated.

There are many operators, functions, parameters, and settings in the genetic algorithms that can be ameliorated and implemented differently in various problems [5, 12, 14]. In this study three of the main genetic algorithms parameters are investigated, namely, generation gap (GGAP), crossover (XOVR), and mutation rates (MUTR) with values shown in Table 1, according to some statements [17]. Very big generation gap value does not improve performance of GA, especially regarding how fast the solution will be found. Mutation is randomly applied with low probability, typically in the range 0.01 and 0.1. Crossover rate is the parameter that affects the rate at which the crossover operator is applied. A higher crossover rate introduces new strings more quickly into the population. A low crossover rate may cause stagnation due to the lower exploration rate.

The aim of the paper is to study the influence of three of the main genetic algorithms parameters, namely, generation gap, crossover, and mutation rates, to be investigated towards algorithms convergence time with values shown in Table 1. Such genetic algorithms parameters are examined for altogether eight kinds of simple and multipopulation genetic algorithms. Their performances are demonstrated for *S. cerevisiae* fed-batch cultivation.

## 2. Parameter Identification of *S. cerevisiae* Fed-Batch Cultivation Using Simple and Multipopulation Genetic Algorithms

Experimental data of *S. cerevisiae* fed-batch cultivation is obtained in the *Institute of Technical Chemistry, University of Hannover, Germany* [10]. The cultivation of the yeast *S. cerevisiae* is performed in a 1.5 L reactor, using a Schatzmann medium. Glucose in feeding solution is 50 g·L<sup>-1</sup>. The temperature was controlled at 30°C, the pH at 5.7. The stirrer speed was set to 500 rpm. Biomass and ethanol were measured off-line, while substrate (glucose) and dissolved oxygen were measured on-line.

Mathematical model of *S. cerevisiae* fed-batch cultivation is commonly described as follows, according to the mass balance [18]:

$$\begin{aligned} \frac{dX}{dt} &= \mu X - \frac{F}{V} X, \\ \frac{dS}{dt} &= -q_s X + \frac{F}{V} (S_{in} - S), \\ \frac{dE}{dt} &= q_E X - \frac{F}{V} E, \\ \frac{dO_2}{dt} &= -q_{O_2} X + k_L^{O_2} a (O_2^* - O_2), \\ \frac{dV}{dt} &= F, \end{aligned} \quad (1)$$

TABLE 1: Range of investigated genetic algorithm parameters.

GGAP	XOVR	MUTR
0.5	0.65	0.02
0.67	0.75	0.04
0.8	0.85	0.06
0.9	0.95	0.08
—	—	0.1

where  $X$  is the concentration of biomass, [ $\text{g}\cdot\text{L}^{-1}$ ];  $S$ —concentration of substrate (glucose), [ $\text{g}\cdot\text{L}^{-1}$ ];  $E$ —concentration of ethanol, [ $\text{g}\cdot\text{L}^{-1}$ ];  $O_2$ —concentration of oxygen, [%];  $O_2^*$ —dissolved oxygen saturation concentration, [%];  $F$ —feeding rate, [ $\text{L}\cdot\text{h}^{-1}$ ];  $V$ —volume of bioreactor, [l];  $k_L^{O_2} a$ —volumetric oxygen transfer coefficient, [ $\text{h}^{-1}$ ];  $S_{\text{in}}$ —initial glucose concentration in the feeding solution, [ $\text{g}\cdot\text{L}^{-1}$ ];  $\mu$ ,  $q_S$ ,  $q_E$ ,  $q_{O_2}$ —specific growth/utilization rates of biomass, substrate, ethanol and dissolved oxygen, [ $\text{h}^{-1}$ ].

Fed-batch cultivation of *S. cerevisiae* considered here is characterized with keeping glucose concentration equal to or below to its critical level ( $S_{\text{crit}} = 0.05 \text{ g}\cdot\text{L}^{-1}$ ), sufficient dissolved oxygen  $O_2 \geq O_{2\text{crit}}$  ( $O_{2\text{crit}} = 18\%$ ) and availability of ethanol in the broth. This state corresponds to the so-called *mixed oxidative state* (FS II) according to functional state modeling approach [18]. As presented in [18], the specific growth rate is generally found to be a sum of two terms, one describing the contribution of sugar and the other the contribution of ethanol to yeast growth. Both terms have the structure of Monod model. Monod model is also used for the specific ethanol and sugar consumption rates. Dissolved oxygen consumption rate is obtained as a sum of two terms, which are directly proportional to the specific glucose rate and specific ethanol production rate, respectively. Hence, specific rates in (1) are presented as follows:

$$\begin{aligned} \mu &= \mu_{2S} \frac{S}{S + k_S} + \mu_{2E} \frac{E}{E + k_E}, & q_S &= \frac{\mu_{2S}}{Y_{SX}} \frac{S}{S + k_S}, \\ q_E &= -\frac{\mu_{2E}}{Y_{EX}} \frac{E}{E + k_E}, & q_{O_2} &= q_E Y_{OE} + q_S Y_{OS}, \end{aligned} \quad (2)$$

where  $\mu_{2S}$ ,  $\mu_{2E}$  are the maximum growth rates of substrate and ethanol, [ $\text{h}^{-1}$ ];  $k_S$ ,  $k_E$ —saturation constants of substrate and ethanol, [ $\text{g}\cdot\text{L}^{-1}$ ];  $Y_{ij}$ —yield coefficients, [ $\text{g}\cdot\text{g}^{-1}$ ].

As an optimization criterion, mean square deviation between the model output and the experimental data obtained during cultivation has been used:

$$J_Y = \sum (Y - Y^*)^2 \rightarrow \min, \quad (3)$$

where  $Y$  is the experimental data;  $Y^*$ —model predicted data;  $Y = [X, S, E, O_2]$ .

Parameter identification of the model (1) has been performed using *Genetic Algorithm Toolbox* [14] in *MATLAB 7* environment. All the computations are performed using a PC Intel Pentium 4 (2.4 GHz) platform running Windows XP. All kinds of genetic algorithms—four kinds of SGA and four kinds of MpGA—have been consequently applied for the purposes of parameter identification of

TABLE 2: Genetic algorithms parameters.

Parameter	Value
NVAR	9
PRECI	20
NIND	20
MAXGEN	100
MIGR	0.2
INSR	0.95
SUBPOP	5
MIGGEN	20

TABLE 3: Genetic operators.

Operator	Type
Encoding	Binary
Reinsertion	Fitness based
Crossover	Double point
Mutation	Bit inversion
Selection	Roulette wheel selection
Fitness function	Linear ranking

*S. cerevisiae* fed-batch cultivation. The values of genetic algorithms parameters except GGAP, XOVR, and MUTR for all considered here kinds of genetic algorithms, both simple and multipopulation, have been accepted as presented in Table 2, while the type of genetic operators is as listed in Table 3.

In Table 2 NVAR is the number of variables; PRECI—precision of binary representation; NIND—number of individuals; MAXGEN—maximum number of generations; MIGR—migration rate; INSR—insertion rate; SUBPOP—number of subpopulations; MIGGEN—number of generation, after which migration takes place between subpopulations.

The influence of main genetic algorithm operators considered here, namely, GGAP, XOVR, and MUTR, has been firstly examined for all four kinds of SGA, and a comparison in relation to model accuracy and convergence time is demonstrated in Table 4. When one of the parameters GGAP, XOVR, or MUTR is investigated according to Table 1, the basic values for the other two parameters are as follows, according to some statements [17]: GGAP = 0.8, XOVR = 0.95, and MUTR = 0.05.

As shown in Table 4, the optimization criterion values obtained with four kinds of standard genetic algorithms are very similar, varying between 0.0221 and 0.0230 which means about 4% variance. As one can see, the results obtained with SGA-SCM are very similar to results obtained with SGA-SMC. Also, results when SGA-CMS is applied are closed to those with SGA-MCS, but the convergence time is much less than the first group. One can summarize that proceeding selection operator before crossover and mutation (no matter their order) needs more computational time. This fact is valid for investigation of both parameters—GGAP and XOVR. It should be noted that the GGAP is the most sensitive from three investigated parameters concerning the

TABLE 4: Results from model parameter identification using different kinds of SGA.

	SGA-SCM		SGA-CMS		SGA-MCS		SGA-SMC	
	$J$	$t$ [s]	$J$	$t$ [s]	$J$	$t$ [s]	$J$	$t$ [s]
GGAP								
0.5	0.0223	43.8120	0.0222	38.1710	0.0221	37.2810	0.0221	44.5160
0.67	0.0221	52.7970	0.0225	37.1870	0.0223	39.7810	0.0221	55.0460
0.8	0.0221	67.9220	0.0222	50.2970	0.0224	53.4530	0.0221	67.6410
0.9	0.0222	70.6250	0.0224	52.3440	0.0230	61.4370	0.0222	70.9690
XOVR								
0.65	0.0222	65.5930	0.0223	45.9680	0.0221	50.0930	0.0222	69.6560
0.75	0.0223	60.7650	0.0223	44.8590	0.0223	47.7350	0.0221	65.9530
0.85	0.0221	65.6250	0.0224	45.9220	0.0227	41.5310	0.0221	61.6100
0.95	0.0221	65.6090	0.0224	45.9220	0.0225	50.5000	0.0222	63.1720
MUTR								
0.02	0.0222	56.2190	0.0228	46.5150	0.0222	48.6570	0.0222	55.1090
0.04	0.0221	62.0000	0.0221	44.3120	0.0223	50.1250	0.0221	56.1880
0.06	0.0223	70.9530	0.0222	42.7970	0.0221	45.5470	0.0222	65.3280
0.08	0.0222	81.0470	0.0224	49.0000	0.0222	43.8430	0.0221	76.2030
0.1	0.0221	82.5150	0.0221	42.5160	0.0226	40.9370	0.0222	83.0310

TABLE 5: Values of model parameters when SGA-MCS has been applied.

Parameter	$\mu_S$ [h <sup>-1</sup> ]	$\mu_E$ [h <sup>-1</sup> ]	$k_S$ [g·L <sup>-1</sup> ]	$k_E$ [g·L <sup>-1</sup> ]	$Y_{SX}$ [g·g <sup>-1</sup> ]	$Y_{EX}$ [g·g <sup>-1</sup> ]	$k_{La}$ [h <sup>-1</sup> ]	$Y_{OS}$ [g·g <sup>-1</sup> ]	$Y_{OE}$ [g·g <sup>-1</sup> ]
Value	0.95	0.12	0.12	0.80	0.4	1.47	113.02	894.88	254.95

convergence time-up to almost 40% (in case of SGA-MCS, which is distinguished also as the fastest one algorithm) using GGAP = 0.5 instead of 0.9 can be saved without loss of accuracy. Even more exactly in case of SGA-MCS the objective function value decreases from the maximum observed in all performances value of 0.0230 to the lowest one—0.0221. Exploring different values of crossover rate no such time saving is realized but it should be pointed that values of 0.85 for XOVR can be assumed as more appropriate. Only in MUTR no tendency of influence can be drawn. For the “favorite” of the considered here algorithms SGA-MCS, value of 0.1 for MUTR can save up to 20%. It is also demonstrated that there is no loss of model accuracy when the operator mutation is performed before crossover. Moreover, proposed modification in SGA in most cases reduces convergence time. Presented here comparison shows that the implementation of the operators in a sequence of mutation, crossover, and then selection is the most optimal according to convergence time with guaranteed high accuracy of the decision.

Based on such analysis, as a “favorite,” SGA-MCS can be distinguished. Parameter identification of *S. cerevisiae* fed-batch cultivation has been performed applying SGA-MCS at chosen genetic parameter values GGAP = 0.5, XOVR = 0.85, and MUTR = 0.1. As a result of parameter identification, the values of model parameters are as presented in Table 5, while the CPU time = 38.6410 s and  $J = 0.0223$ .

Figure 1 presents results from experimental data and model prediction, respectively, for biomass, ethanol, substrate, and dissolved oxygen.

The same analysis has been performed also for four kinds of multipopulation algorithms. The tendencies described in SGA have been proven to multipopulation algorithms as well (results not shown because of similarity). As a “favorite” among MpGA the one with standard sequence, namely, selection, crossover, mutation, has been distinguished. GGAP is again the most sensitive parameter concerning the convergence time leading again to saving up to almost 40% using GGAP = 0.5 instead of 0.9 without loss of accuracy. Parameter identification of *S. cerevisiae* fed-batch cultivation has been performed applying MpGA-SCM at chosen genetic parameter values GGAP = 0.5, XOVR = 0.85, and MUTR = 0.02 (the only one difference to SGA). As a result of parameter identification, the values of model parameters are as presented in Table 6, while the CPU time = 97.5940 s and  $J = 0.0221$ .

Figure 2 presents results from experimental data and model prediction, respectively, for biomass, ethanol, substrate, and dissolved oxygen.

Presented in both figures, the results from SGA and MpGA application for parameter identification of *S. cerevisiae* fed-batch cultivation show the effectiveness of GA for solving complex nonlinear problems.

### 3. Conclusions

In this investigation altogether eight modifications of genetic algorithms—four kinds of simple and four kinds of multipopulation genetic algorithms—have been examined. Different modifications of SGA and MpGA are with exchanged

TABLE 6: Values of model parameters when MpGA-SCM has been applied.

Parameter	$\mu_S$ [ $\text{h}^{-1}$ ]	$\mu_E$ [ $\text{h}^{-1}$ ]	$k_S$ [ $\text{g}\cdot\text{L}^{-1}$ ]	$k_E$ [ $\text{g}\cdot\text{L}^{-1}$ ]	$Y_{SX}$ [ $\text{g}\cdot\text{g}^{-1}$ ]	$Y_{EX}$ [ $\text{g}\cdot\text{g}^{-1}$ ]	$k_L a$ [ $\text{h}^{-1}$ ]	$Y_{OS}$ [ $\text{g}\cdot\text{g}^{-1}$ ]	$Y_{OE}$ [ $\text{g}\cdot\text{g}^{-1}$ ]
Value	0.90	0.15	0.15	0.80	0.40	2.03	96.40	764.53	441.63

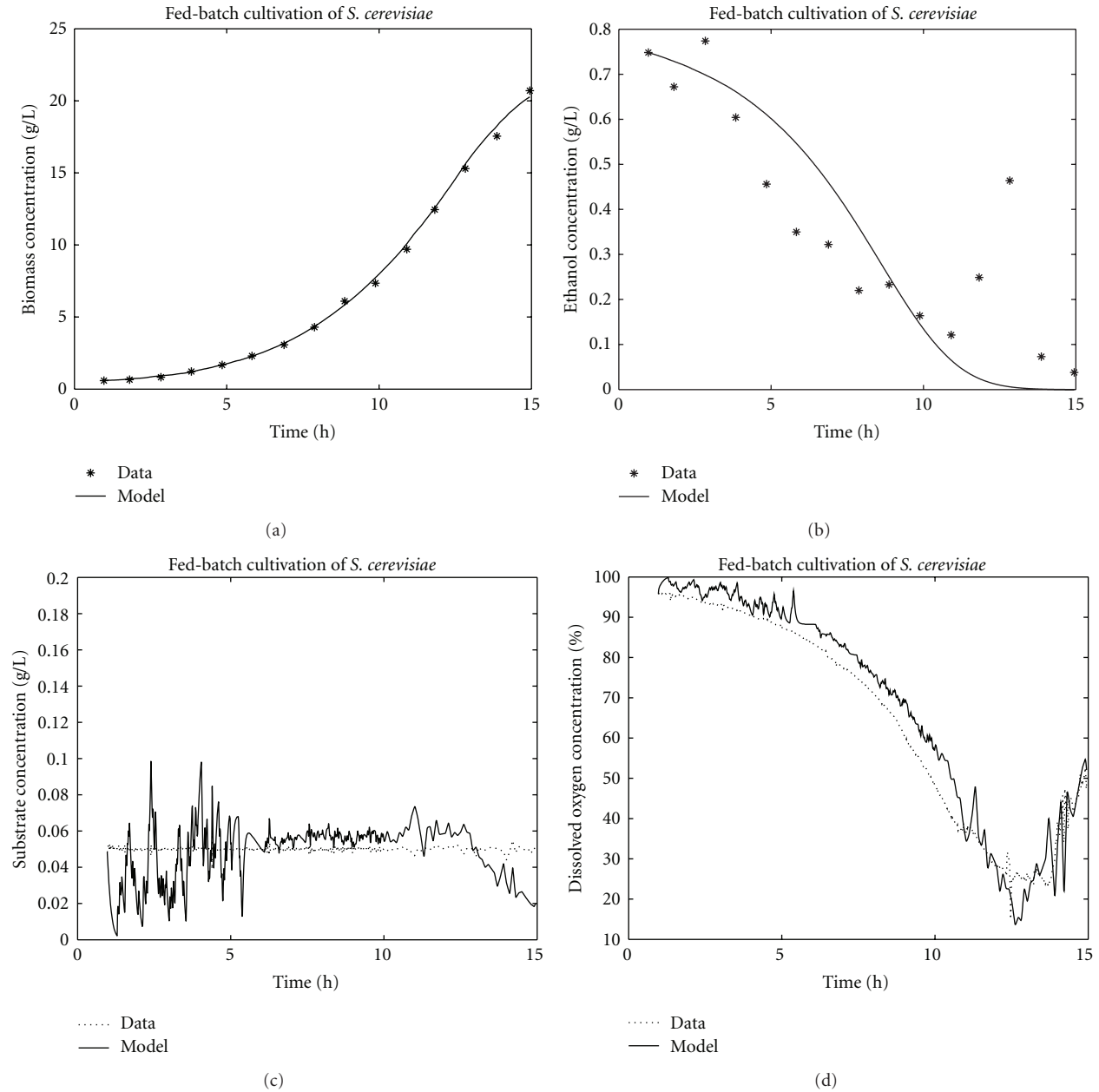


FIGURE 1: Model prediction compared to experimental data, respectively, for biomass, ethanol, substrate, and dissolved oxygen concentrations.

operators' sequence of selection, crossover, and mutation operators. The influence of some of genetic algorithm parameters, namely, generation gap, crossover, and mutation rates, has been explored for all eight kinds of genetic algorithms aiming to improve the convergence time. Among the three investigated parameters, the generation gap is the most sensitive one towards to convergence time. As

“favorites” among the considered here algorithms, SGA-MCS and MpGA-SCM have been distinguished. Up to almost 40% from calculation time can be saved in cases of SGA-MCS and MpGA-SCM application using  $\text{GGAP} = 0.5$  instead of 0.9 without loss of model accuracy. Exploring different values of crossover and mutation rates no such time saving is realized but it should be pointed that values of

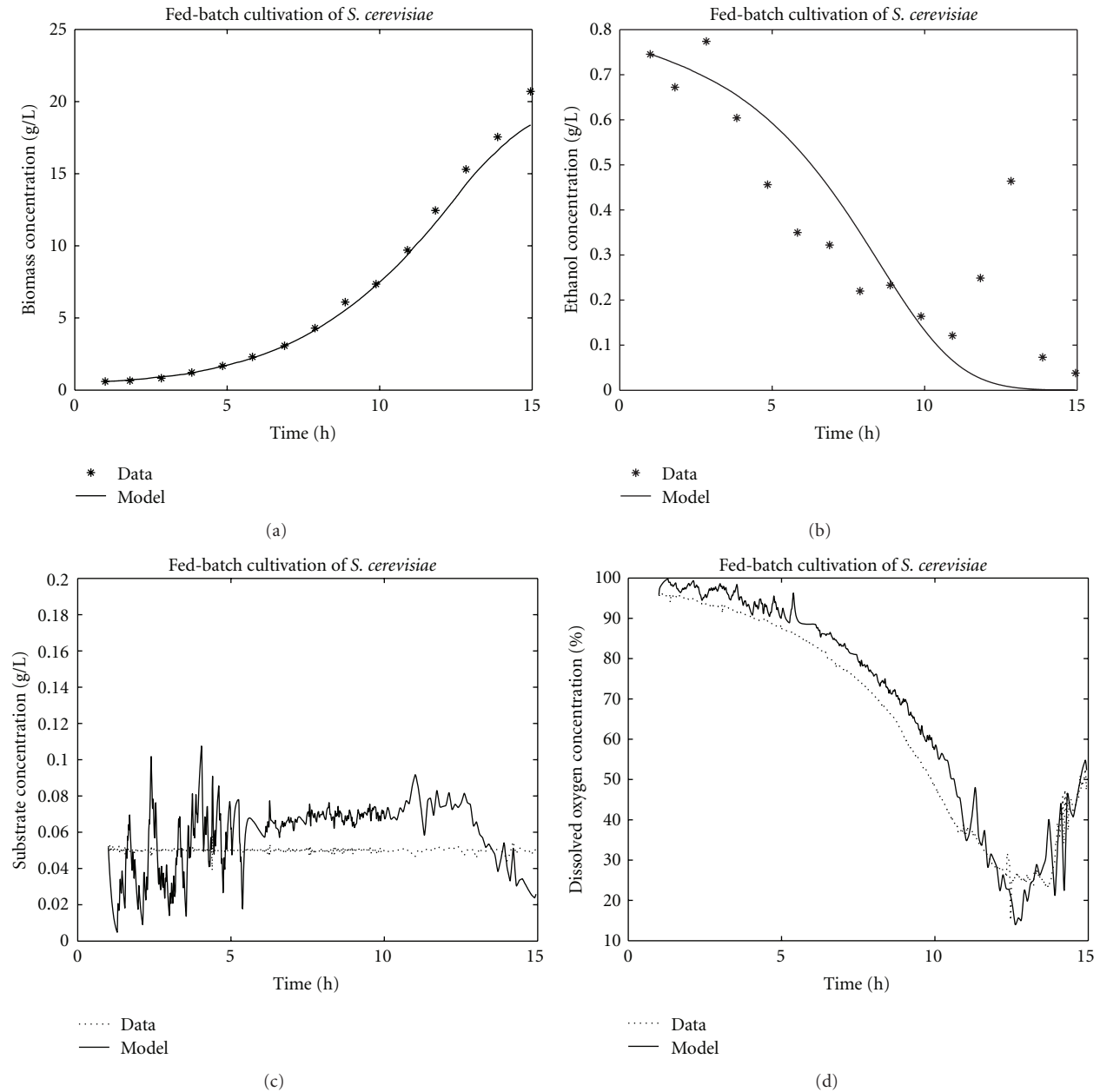


FIGURE 2: Model prediction compared to experimental data, respectively, for biomass, ethanol, substrate, and dissolved oxygen concentrations.

0.85 for crossover rate can be assumed as more appropriate. Employing such values of genetic algorithm parameters, both distinguished algorithms, as well as all others modification of SGA and MpGA, show the effectiveness of genetic algorithms for solving complex nonlinear problems.

## Acknowledgments

This work is partially supported by the European Social Fund and Bulgarian Ministry of Education, Youth and Science under Operative Program "Human Resources Development," Grant BG051PO001-3.3.04/40 and National Science

Fund of Bulgaria, Grant no. DID 02-29 "Modeling Processes with Fixed Development Rules."

## References

- [1] S. Fidanova, "An heuristic method for GPS surveying problem, computational science," *Lecture Notes in Computer Science*, vol. 4490, no. 4, pp. 1084–1090, 2007.
- [2] F. Glover and G. A. Kochenberger, *Handbook of Metaheuristics*, Kluwer Academic, Boston, Mass, USA, 2003.
- [3] G. R. Raidl, "A unified view on hybrid metaheuristics," *Lecture Notes in Computer Science*, vol. 4030, pp. 1–12, 2006.

- [4] I. Hristozov, T. Pencheva, B. Iliev, and St. Tzonkov, "A Comparative analysis of genetic algorithms and conventional optimization procedures," in *Proceedings of the Scientific-Technical Union of Mechanical Engineering*, V(3-25), pp. 187–192, 1998.
- [5] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wiley Publishing Company, Massachusetts, Mass, USA, 1989.
- [6] G. E. Carrillo-Ureta, P. D. Roberts, and V. M. Becerra, "Genetic algorithms for optimal control of beer fermentation," in *Proceedings of the 2001 IEEE International Symposium on Intelligent Control (ISIC '01)*, pp. 391–396, Mexico City, Mexico, September 2001.
- [7] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, Berlin, Heidelberg, Germany, 2nd edition, 1994.
- [8] J.-G. Na, Y. K. Chang, B. H. Chung, and H. C. Lim, "Adaptive optimization of fed-batch culture of yeast by using genetic algorithms," *Bioprocess and Biosystems Engineering*, vol. 24, no. 5, pp. 299–308, 2001.
- [9] M. Angelova, S. Tzonkov, and T. Pencheva, "Genetic algorithms based parameter identification of Yeast Fed-Batch cultivation," in *Proceedings of the Conference on "Numerical Methods and Applications"*, vol. 6046 of *Lecture Notes in Computer Science*, pp. 224–231, 2011.
- [10] T. Pencheva, O. Roeva, and I. Hristozov, *Functional State Approach to Fermentation Processes Modelling*, Prof. Marin Drinov Academic Publishing House, Sofia, Bulgaria, 2006.
- [11] M. Ranganath, S. Renganathan, and C. Gokulnath, "Identification of bioprocesses using genetic algorithm," *Bioprocess Engineering*, vol. 21, no. 2, pp. 123–127, 1999.
- [12] O. Roeva, *Improvement of Genetic Algorithm Performance for Identification of Cultivation Process Models*, *Advanced Topics on Evolutionary Computing*, Artificial Intelligence Series-WSEAS, 2008.
- [13] O. Roeva, "A modified genetic algorithm for a parameter identification of fermentation processes," *Biotechnology and Biotechnological Equipment*, vol. 20, no. 1, pp. 202–209, 2006.
- [14] A. J. Chipperfield, P. Fleming, H. Pohlheim, and C. M. Fonseca, *Genetic Algorithm Toolbox for Use with MATLAB*, User's Guide, Version 1.2, Department of Automatic Control and System Engineering, University of Sheffield, Sheffield, UK, 1994.
- [15] O. Cordon and F. Herrera, "Hybridizing genetic algorithms with sharing scheme and evolution strategies for designing approximate fuzzy rule-based systems," *Fuzzy Sets and Systems*, vol. 118, no. 2, pp. 235–255, 2001.
- [16] R. J. Kuo, C. H. Chen, and Y. C. Hwang, "An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network," *Fuzzy Sets and Systems*, vol. 118, no. 1, pp. 21–45, 2001.
- [17] M. Obittko, "Genetic Algorithm," 2005, <http://cs.felk.cvut.cz/~xobitko/ga/main.html>.
- [18] X.-C. Zhang, A. Visala, A. Halme, and P. Linco, "Functional state modelling approach for bioprocesses: local models for aerobic yeast growth processes," *Journal of Process Control*, vol. 4, no. 3, pp. 127–134, 1994.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

