Influence of Genetic Algorithm Parameters on Their Performance for Parameter Identification of a Yeast Fed-batch Fermentation Process Model

Maria Angelova, Tania Pencheva*

Institute of Biophysics and Biomedical Engineering
Bulgarian Academy of Sciences
105 Acad. G. Bonchev Str., Sofia 1113, Bulgaria
E-mails: maria.angelova@biomed.bas.bg, tania.pencheva@biomed.bas.bg

**Corresponding author*

Received: September 05, 2024 **Accepted: December 12, 2024**

Published: December 31, 2024

Abstract: *Eight single (SGA) and eight multi-population (MGA) genetic algorithms (GA) differing in the sequence of implementation of the main genetic operators' selection, crossover and mutation, or omitting the mutation operator, have been examined for the purposes of parameter identification of a Saccharomyces cerevisiae fed-batch fermentation process model. The influence of some of the main genetic algorithm parameters, namely number of individuals, maximum number of generations, generation gap, crossover and mutation rates for both SGA and MGA, and insertion and migration probability for MGA only, have been investigated in depth. Almost all applied SGA and MGA led to similar values of the optimization criterion but the SGA with operators' sequence mutation, crossover and selection, and MGA with operators' sequence crossover, selection and mutation, are significantly faster than others while keeping the model accuracy. Among the considered GA parameters, generation gap influences most significantly to SGA and MGA convergence time, saving of about 40% of computational time of the algorithms without affecting the model accuracy.*

Keywords: *Single genetic algorithms, Multi-population genetic algorithms, Parameter identification, Fed-batch fermentation process model, Saccharomyces cerevisiae.*

Introduction

Genetic algorithms (GA) proposed by Holland [13] and further developed by Goldberg [11] are an optimization technique, frequently applied and proven to be very successful in solving of complicated optimization tasks [1, 15]. Modelling and further optimization of fermentation processes can be assumed as such a complex problem. The importance of fermentation processes comes from their numerous applications in different branches of industry, starting from the production of pharmaceuticals, chemicals and enzymes, and ending to the yield of yeast, foods and beverages. It is well known that living microorganisms participated in fermentation processes predetermine their specific characteristics as modelling and control objects. This is the reason fermentation processes to have a fame of complex, nonlinear, dynamic systems with interdependent and time-varying process variables. An important step for the adequate modelling of such a complicated and rather time-consuming task is the choice of a certain optimization procedure for model parameter identification. In general, the conventional optimization methods cannot meet the limitations of fermentation processes, while GA, as a representative of metaheuristic and stochastic techniques, have all prerequisites to overcome them [14]. The applicability of single-population genetic algorithm (SGA) and multi-population genetic algorithm (MGA), as well as some of their modifications, were demonstrated for identifying model parameters in

bacteria and yeast fermentation processes models [13, 14, 17, 18]. Thus, GA have been proven to be successful tools for solving the above discussed problem.

The aim of this investigation is to demonstrate sixteen different kinds of GA (eight SGA and eight MGA) as a powerful tool for parameter identification of a *S. cerevisiae* fed-batch fermentation process model. Moreover, the influence of the main genetic algorithm parameters, namely number of individuals, maximum number of generations, generation gap, crossover and mutation rates, for both SGA and MGA, as well as insertion and migration probability for MGA, is going to be thoroughly investigated.

Yeast fed-batch fermentation process model

For the purposes of model parameter identification, experimental data from *S. cerevisiae* fed-batch cultivation conducted in the *Institute of Technical Chemistry, University of Hannover, Germany* have been used. Process conditions and real experimental data consisting of on-line measurements of substrate (glucose) and dissolved oxygen, as well as off-line measurements of biomass and ethanol have been fully described in [14].

Mathematical model in which the dynamics of biomass, substrate, ethanol and dissolved oxygen concentrations of *S. cerevisiae* fed-batch fermentation process is commonly described by the system of non-linear differential equations [14], as given below:

$$\frac{dX}{dt} = \left(\mu_{2S} \frac{S}{S + k_S} + \mu_{2E} \frac{E}{E + k_E} \right) X - \frac{F}{V} X \tag{1}$$

$$\frac{dS}{dt} = -\frac{\mu_{2S}}{Y_{SX}} \frac{S}{S + k_S} X + \frac{F}{V} \left(S_{\text{in}} - S \right) \tag{2}$$

$$\frac{dE}{dt} = -\frac{\mu_{2E}}{Y_{EX}} \frac{E}{E + k_E} X - \frac{F}{V} E \tag{3}$$

$$\frac{dO_2}{dt} = \left(\frac{\mu_{2E}}{Y_{EX}} \frac{E}{E + k_E} Y_{OE} - \frac{\mu_{2S}}{Y_{SX}} \frac{S}{S + k_S} Y_{OS} \right) X + k_L^{O_2} a (O_2^* - O_2) \tag{4}$$

$$\frac{dV}{dt} = F \quad , \tag{5}$$

where X, S, S_{in}, E, O_2 are respectively the concentrations of biomass, [g·l^{-1}], substrate (glucose), [g·l^{-1}], initial glucose concentration in the feeding solution, [g·l^{-1}], ethanol, [g·l^{-1}], and dissolved oxygen, [%]; O_2^* – dissolved oxygen saturation concentration, [%]; F – feeding rate, [h^{-1}]; V – volume of bioreactor, [l]; $k_L^{O_2} a$ – volumetric oxygen transfer coefficient, [h^{-1}]; μ_{2S}, μ_{2E} – maximum growth rates of substrate and ethanol, [h^{-1}]; k_S, k_E – saturation constants of substrate and ethanol, [g·l^{-1}]; Y_{SX}, Y_{OS}, Y_{EX}, Y_{OE} – yield coefficients of substrate and ethanol, [g g^{-1}].

All functions in the model (Eqs. (1)-(5)) are continuous and differentiable. Also, nine model parameters, $p = [\mu_{2S}, \mu_{2E}, k_S, k_E, Y_{SX}, Y_{EX}, k_L^{O_2} a, Y_{OS}, Y_{OE}]$, going to be identified, fulfil the non-zero division requirement.

Mean square deviation between the model output and the experimental data obtained during the fermentation process is used as an optimization criterion:

$$J = \sum \left(Y - Y^* \right)^2 \rightarrow min \,, \tag{6}$$

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

Genetic algorithms – background and modifications

GA work in a stochastic way with a population of coded parameter sets known as chromosomes and search a global optimal solution through the three main GA operators – selection, crossover and mutation – implementing the idea of survival of the fittest. GA may act in a single-population and multi-population mode, with a certain number of individuals/populations involved. Standard single-population GA works with one population of individuals. Each randomly generated individual (solution) is evaluated, and based on the evaluation, a fitness value is assigned. The most suitable solutions are selected according to this value, then crossover proceeds to form a new offspring. Finally, mutation with a determined probability has occurred aiming to prevent falling of all solutions into a local optimum.

Multi-population GA is more similar to nature and works with many populations, called subpopulations, which proceed independently from each other. After a certain number of generations, known as an isolation time, individuals migrate between the subpopulations.

According to [11, 12], working principle for SGA and MGA might be presented in several steps as follows:

1. **[Start]**
   Generate a random population of *n* chromosomes in SGA
   or
   Generate *k* random subpopulations, each with *n* chromosomes in MGA
2. **[Object function]**
   Evaluate the object function of each chromosome *x* in the population/subpopulation
3. **[Fitness function]**
   Evaluate the fitness function of each chromosome *n* in the population/subpopulation
4. **[New population]**
   Create a new population by repeating following steps:
   4.1. **[Selection]**
   Select parent chromosomes from the population/subpopulation according to their fitness function
   4.2. **[Crossover]**
   Cross over the parents to form new offspring with a crossover probability
   4.3. **[Mutation]**
   Mutate new offspring at each locus with a mutation probability
5. [**Accepting**]
   Place a new offspring in a new population
6. **[Replace]**
   Use new generated population for a further run of the algorithm

7. **[Migrate] (for MGA only)**
   Migration of individuals between subpopulations after the time of isolation has expired
8. **[Test]**
   If the end condition is satisfied, stop and return the best solution in current population, else move to **Loop** step
9. **[Loop]**
   Go to **Step 2**.

SGA and MGA terminate when a certain criterion is met, e.g. number of generations performed, evaluation time passed, fitness threshold reached, fitness convergence satisfied, etc; in this investigation – when a certain number of generations is performed.

Modified SGA

SGA, originally developed in [11] works with the main GA operators in a sequence selection, crossover and mutation, thus further denoted as SGA_scm (comes from selection, crossover, mutation). Since the main idea of GA is to imitate the processes occurring in nature, one can assume that the probability crossover to occur before mutation is comparable to that mutation to come first, followed by crossover; also, selection could be performed before or after crossover and mutation, no matter of their order. This idea has been firstly developed as a modified simple genetic algorithm SGA_cms and applied to parameter identification of an *E. coli* fermentation process [20]. Following this logic, the authors of this investigation elaborated four new modifications of SGA, namely SGA_smc, SGA_mcs, SGA_csm and SGA_msc [3, 4, 6-8], along with the SGA_scm and SGA_cms [2]. All elaborated SGA modifications are successfully applied for the purposes of parameter identification of the *S. cerevisiae* fed-batch fermentation process model, aiming to improve the model accuracy and GA convergence time. Another SGA modification without performance of mutation operator [21], here denoted as SGA_sc, becomes an authors' inspiration for development of SGA_cs [8], also for the aforementioned purposes.

To illustrate the above-mentioned working principle of GA and elaborated SGA modifications, below are given the sequence of algorithm steps for SGA:

- SGA_scm (comes from selection, crossover, mutation): 1, 2, 3, *4.1*, *4.2*, *4.3*, 5, 6, 8, 9
- SGA_cms (comes from crossover, mutation, selection): 1, 2, 3, *4.2*, *4.3*, *4.1*, 5, 6, 8, 9
- SGA_smc (comes from selection, mutation, crossover): 1, 2, 3, *4.1*, *4.3*, *4.2*, 5, 6, 8, 9
- SGA_mcs (comes from mutation, crossover, selection): 1, 2, 3, *4.3*, *4.2*, *4.1*, 5, 6, 8, 9
- SGA_csm (comes from crossover, selection, mutation): 1, 2, 3, *4.2*, *4.1*, *4.3*, 5, 6, 8, 9
- SGA_msc (comes from mutation, selection, crossover): 1, 2, 3, *4.3*, *4.1*, *4.2*, 5, 6, 8, 9
- SGA_sc (comes from selection, crossover): 1, 2, 3, *4.1*, *4.2*, 5, 6, 8, 9
- SGA_cs (comes from crossover, selection): 1, 2, 3, *4.2*, *4.1*, 5, 6, 8, 9.

Following the same line of logics and inspired also from [16], the same modifications have been elaborated for the MGA as well, considering the obligate for MGA **Step 7** from the GA working principle, presented above. Additional explanations for elaboration of different MGA modifications, along with their applications to the *S. cerevisiae* fed-batch fermentation process model could be found in [5, 7, 9, 18].

The parameter identification of the model (1)-(5), applying (6) as an optimization criterion, was carried out in Matlab 7 environment using Genetic Algorithm Toolbox [10].

For the purposes of different algorithms comparison, all calculations were performed on Intel Pentium 4 PC (2.4 GHz) platform running Windows XP.

Results and discussion

Investigation of the influence of SGA parameters

For the purposes of parameter identification of the *S. cerevisiae* fed-batch fermentation process, altogether eight kinds of SGA were examined. The influence of five main GA parameters, namely number of individuals (NIND), number of generations (MAXGEN), generation gap (GGAP), crossover probability (XOVR) and mutation probability (MUTR) were investigated with the aim to explore their influence on the SGA solutions' accuracy and convergence time. Table 1 presents the values of GA parameters, chosen according to some recommendations in [17]:

Table 1. Values of the investigated GA parameters

NIND	MAXGEN	GGAP	XOVR	MUTR
20	100	0.5	0.65	0.02
40	200	0.67	0.75	0.04
60	500	0.8	0.85	0.06
80	1000	0.9	0.95	0.08
100	-	-	-	0.1

Each GA parameter has been examined separately, while the values of the other parameters remain unchanged. When any of the GA parameters is examined for its values (Table 1), the values of the other parameters are set to: NIND = 20, MAXGEN = 100, GGAP = 0.8, XOVR = 0.95 and MUTR = 0.05 according to recommendations in [17]. The type of GA operators used for different SGA are presented in Table 2.

Table 2. Type of GA operators

Operator	Type
encoding	binary
reinsertion	fitness-based
crossover	double point
mutation	bit inversion
selection	roulette wheel selection
fitness function	linear ranking

After a thorough analysis of the influence of the main GA parameters on the implementation of the eight kinds of SGA, no significant decrease of the time to find a global minimum has been observed when different values of XOVR have been explored [2]. However, XOVR = 0.85 might be recommended as more preferable value. Also, no clear trend might be drawn for the influence of MUTR on the convergence time and the solution accuracy of the eight SGA [2]. Meanwhile, it is worth to note that SGA_mcs executed at MUTR = 0.1 has led to save up to 20% of the algorithm convergence time [2, 7], thus recommended as a preferable value.

When the influence of MAXGEN has been explored for all kinds of SGA, the value of the objective function has decreased quite insignificantly with the increase of MAXGEN but for the expense of the algorithms computational time. As such, MAXGEN = 100 is accepted

as a suitable compromise for the degree of the model accuracy and computational time [2, 7]. Concerning NIND, the value of the objective function has been even increased a bit with the increase of the number of individuals in the population. The time for solution finding has also been grown. That is why NIND = 20 is considered as an acceptable compromise for the achieved model accuracy and the computational time required by the algorithm to reach an optimal solution [2, 7].

Among all investigated five GA parameters, GGAP has been distinguished as the most sensitive one according to the algorithm's convergence time [2, 7]. Table 3 presents the results for the values of the optimization criterion *J*, as well as for the computational time *t* when eight types of SGA have been implemented at different GGAP values for the purposes of parameter identification of the *S. cerevisiae* fed-batch fermentation process model.

Table 3. Results from parameter identification with eight different types of SGA when GGAP is examined

GGAP	SGA_scm		SGA_smc		SGA_cms		SGA_mcs	
	J	*t*, s	*J*	*t*, s	*J*	*t*, s	*J*	*t*, s
0.5	0.0223	43.81	0.0221	44.52	0.0222	38.17	0.0221	37.28
0.67	0.0221	52.78	0.0221	55.05	0.0225	37.19	0.0223	39.78
0.8	0.0221	67.92	0.0221	67.64	0.0222	50.30	0.0224	53.45
0.9	0.0222	70.63	0.0222	70.97	0.0224	52.34	0.0230	61.44

Table 3. (continuation)

GGAP	SGA_csm		SGA_msc		SGA_cs		SGA_sc	
	J	*t*, s	*J*	*t*, s	*J*	*t*, s	*J*	*t*, s
0.5	0.0224	36.28	0.0223	36.27	0.0230	31.59	0.0223	30.74
0.67	0.0223	39.20	0.0228	53.55	0.0248	41.38	0.0235	43.95
0.8	0.0223	60.31	0.0223	47.49	0.0233	41.25	0.0228	42.58
0.9	0.0235	55.92	0.0223	77.61	0.0230	53.73	0.0237	39.36

As shown in Table 3, the values of the optimization criterion obtained after the application of four types of SGA executed the selection operator before and after the crossover and mutation operators (the upper part of Table 3) are quite similar and vary between 0.0221 and 0.0230, which means about 4% deviation in model accuracy. In the case of SGA_mcs, distinguished as simultaneously the most accurate and the fastest one in the considered group, the algorithm's convergence time can be reduced by almost 40% without loss of model accuracy when GGAP = 0.5 is used instead of 0.9. Even more, in the case of SGA_mcs the value of the objective function decreases from the highest 0.0230, among all observed values in this group, to the lowest one 0.0221.

Similar values of the optimization criterion obtained after the application of four types of SGA ensure that, from the accuracy point of view, the execution of mutation operator before the crossover operator does not lead to a decrease in the degree of the model adequacy. Moreover, in the most cases, the proposed modifications for SGA reduce the time for reaching global minimum. Presented comparison shows that the execution of GA operators in a sequence mutation, crossover and selection is optimal in terms of the GA convergence time, with guaranteed high adequacy of the found solution.

The trend observed for the previous four algorithms is outlined again, when the next four GA were examined (the lower part of Table 3). As evident from Table 3, similar structures of SGA lead to obtaining comparable results when considering the value of the objective function and the convergence time of the algorithms. Two algorithms without execution of the mutation operator – SGA_cs and SGA_sc can be distinguished as the fastest. It is quite logical that when GA are performed with two operators (selection and crossover), the computational time of the algorithm will decrease, but in most cases, at the expense of the accuracy of the solution. In the case of SGA_sc, which is distinguished as the fastest among the studied algorithms of this group, up to 28% of the solution time can be saved when using GGAP = 0.5 instead of 0.9, with a significant increase of degree of model accuracy. Two algorithms without execution of the mutation operator are logically the fastest, followed by the two algorithms in which the selection operator is executed between the operators for reproduction of the individuals. It can be concluded that, among the studied parameters, GGAP is the most sensitive parameter to SGA convergence time.

Based on this analysis, SGA_mcs can be distinguished as a "favorite" among all eight studied SGA. In order to demonstrate the obtained results, SGA_mcs is applied for parameter identification of the *S. cerevisiae* fed-batch fermentation process model, using the values of GA parameters recommended based on the above analysis: NIND = 20, XOVR = 0.85, MUTR = 0.1, GGAP = 0.5 and MAXGEN = 100. Table 4 presents the estimated values of model parameters after SGA_mcs implementation.

Table 4. Estimated values of model parameters after SGA_mcs application

Parameter	μ_{2S} **[1/h]**	μ_{2E} **[1/h]**	k_S **[g/l]**	k_E **[g/l]**	Y_{SX} **[g/g]**	Y_{EX} **[g/g]**	k_La **[1/h]**	Y_{OS} **[g/g]**	Y_{OE} **[g/g]**
Value	0.95	0.12	0.12	0.80	0.4	1.47	113.02	894.88	254.95

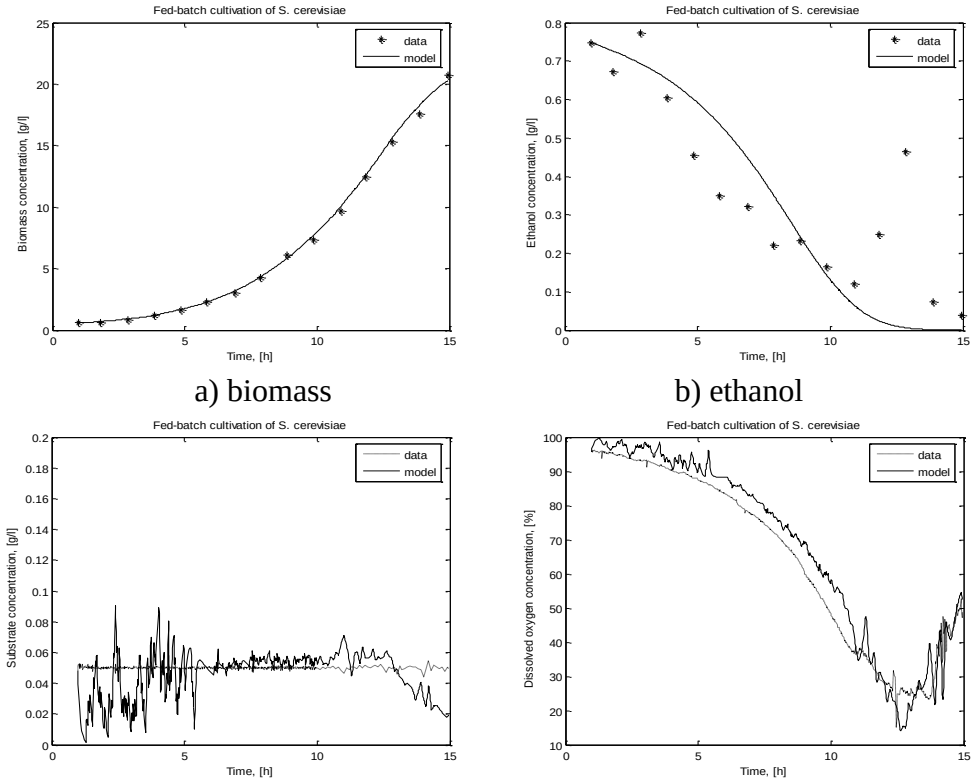
Fig. 1 presents experimental and model-predicted data, for the concentrations of biomass, ethanol, substrate and dissolved oxygen after SGA_mcs implementation for parameter identification of the *S. cerevisiae* fed-batch fermentation process model.

Investigation of the influence of MGA parameters

The influence of altogether seven MGA parameters on the accuracy of the solution and convergence time is investigated: five of them are identical to those considered for SGA (Table 1), and another two, specific for MPA, are added, namely insertion probability (INSR) and migration probability (MIGR). Table 5 presents different values for INSR and MIGR, based on some recommendations in [17].

Table 5. Values of the investigated parameters specific to MGA

INSR	**MIGR**
0.5	0.1
0.8	0.2
0.9	0.4
1	-



a) biomass b) ethanol

c) substrate d) dissolved oxygen

Fig. 1 Experimental and model-predicted data
for biomass, ethanol, substrate and dissolved oxygen concentrations
after SGA_mcs implementation

By analogy to SGA, each GA parameter has been examined separately, while the values of the other parameters remain unchanged. When any of the GA parameters is examined for its values (Tables 1 and 5), the values of the other parameters are set to: NIND = 20, XOVR = 0.95, MUTR = 0.05, GGAP = 0.8, INSR = 0.95, MIGR = 0.2, and MAXGEN = 100. The type of GA operators is identical to those used for SGA (Table 2). Additional parameters necessary only for MGA implementation are number of subpopulations (SUBPOP) and number of generations, after which migration takes place between subpopulations (MIGGEN), set to 5 and 20, respectively, for this investigation.

After a thorough analysis of the influence of the main GA parameters on the implementation of the eight kinds of MGA, no significant decrease of the time to find a global minimum has been again observed when different values of XOVR have been explored [2]. As for SGA, XOVR = 0.85 might be recommended as the most suitable value. Again, no clear trend might be drawn for the influence of MUTR on the convergence time and the solution accuracy of the eight MGA [2]. However, MUTR = 0.02 can be accepted as more preferable value for MGA, saving up to 24% of the computational time in the case of MGA_msc, without loss of model accuracy [2].

By analogy to the analysis for the results obtained when SGA have been implemented, MAXGEN = 100 and NIND = 20 are recommended for MGA implementation as an acceptable compromise for the degree of model accuracy and algorithm computational time [2]. No significant influence has been observed when exploring different values of INSR and MIGR [2]. However, INSR = 0.9 and MIGR = 0.1 are recommended as preferable values

for MGA implementation for the purposes of parameter identification of the yeast fed-batch fermentation process model.

Again, as for SGA, GGAP has been distinguished as the most sensitive one among the studied MGA parameters in terms of the algorithm's convergence time. Table 6 presents the results for the values of the optimization criterion *J*, as well as for the computational time *t* when eight types of MGA have been implemented at different GGAP values for the purposes of parameter identification of the *S. cerevisiae* fed-batch fermentation process model.

Table 6. Results from parameter identification with eight different types of MGA when GGAP is examined

GGAP	MGA_scm		MGA_smc		MGA_cms		MGA_mcs	
	J	***t*, s**	***J***	***t*, s**	***J***	***t*, s**	***J***	***t*, s**
0.5	0.0220	100.89	0.0220	111.78	0.0221	273.91	0.0220	307.84
0.67	0.0221	112.17	0.0220	141.09	0.0221	325.58	0.0220	332.06
0.8	0.0221	155.47	0.0220	178.97	0.0221	321.02	0.0221	373.16
0.9	0.0220	170.27	0.0220	340.67	0.0221	343.69	0.0221	349.75

Table 6. (continuation)

GGAP	MGA_csm		MGA_msc		MGA_cs		MGA_sc	
	J	***t*, s**	***J***	***t*, s**	***J***	***t*, s**	***J***	***t*, s**
0.5	0.0221	97.69	0.0220	98.30	0.0223	267.92	0.0222	111.53
0.67	0.0221	128.88	0.0221	121.86	0.0222	331.97	0.0224	119.73
0.8	0.0221	163.86	0.0220	145.67	0.0223	333.63	0.0221	153.39
0.9	0.0221	165.67	0.0220	166.02	0.0221	357.02	0.0220	168.22

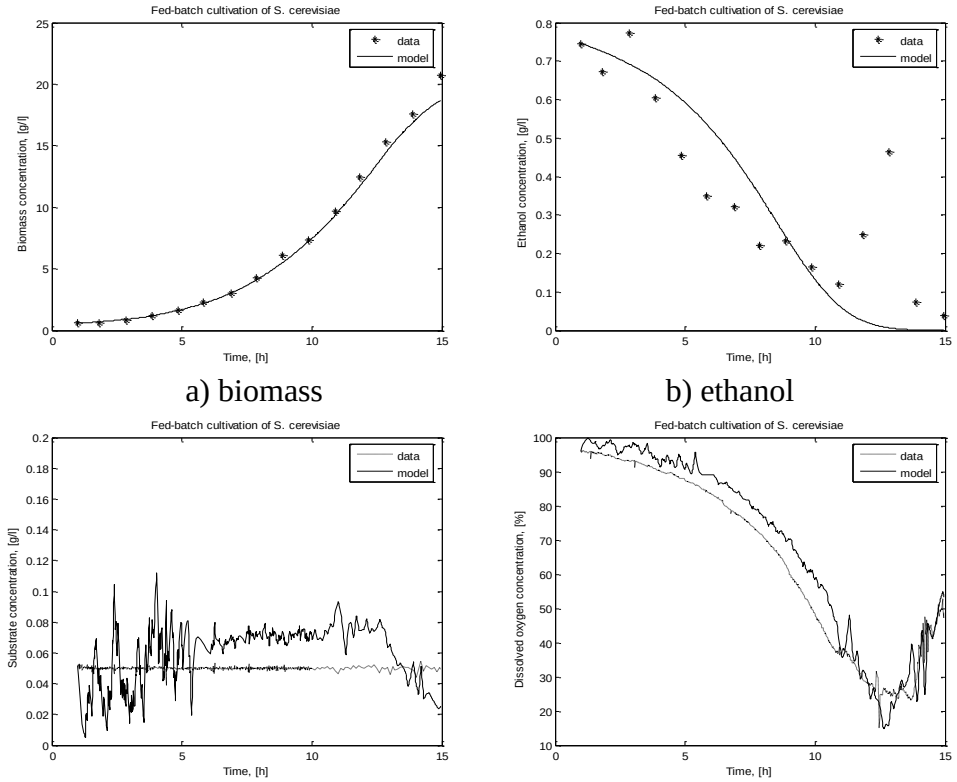
As seen from Table 6, the results obtained by MGA_scm are very similar to the results obtained by MGA_smc. Also, the results when MGA_cms is applied are close to those when MGA_mcs is implemented, but the convergence time for this couple of algorithms is much longer than the first MGA couple. It is obvious that there is no loss of model accuracy when the mutation operator is executed before the crossover operator. Also, it can be summarized that the execution of the selection operator before and between the crossover and mutation operators (no matter of their order) needs less computational time. The accuracy of the two algorithms without performing the mutation operator, namely MGA_sc and MGA_cs, is logically decreased, as previously observed for SGA. In the case of MGA_csm, distinguished as the fastest among the eight studied kinds of MGA, up to 41% of the algorithm's convergence time can be saved without loss of model accuracy if GGAP = 0.5 is used instead of 0.9.

Based on this analysis, MGA_csm can be distinguished as a "favorite" among all eight studied MGA. In order to demonstrate the obtained results, MGA_csm is applied for parameter identification of the *S. cerevisiae* fed-batch fermentation process model, using the values of GA parameters recommended based on the above analysis: NIND = 20, XOVR = 0.85, MUTR = 0.2, GGAP = 0.5, INSR = 0.9 and MIGR = 0.1, and MAXGEN = 100. Table 7 presents the estimated values of model parameters after MGA_csm implementation.

Table 7. Results from parameter identification when MGA_csm is applied

Parameter	μ_{2S} **[1/h]**	μ_{2E} **[1/h]**	k_S **[g/l]**	k_E **[g/l]**	Y_{SX} **[g/g]**	Y_{EX} **[g/g]**	k_La **[1/h]**	Y_{OS} **[g/g]**	Y_{OE} **[g/g]**
Value	0.90	0.12	0.15	0.80	0.41	1.64	65.20	509.82	360.17

Fig. 2 shows results from experimental and model predicted data, for biomass, ethanol, substrate and dissolved oxygen, respectively when MGA_csm is implemented.



a) biomass b) ethanol

c) substrate d) dissolved oxygen

Fig. 2 Experimental and model-predicted data
for biomass, ethanol, substrate and dissolved oxygen concentrations
after MGA_csm implementation

Presented in this investigation results from the implementation of altogether eight kinds of SGA and eight kinds of MGA, demonstrate their effectiveness for solving complex nonlinear problems, such as parameter identification of the *S. cerevisiae* fed-batch fermentation process model.

Conclusion

In this study, different kinds of GA as well as the influence of the main GA parameters have been explored toward convergence time and the degree of model accuracy. Altogether sixteen GA – eight single GA and eight multi-population GA have been implemented for the purposes of parameter identification of the *S. cerevisiae* fed-batch fermentation process model. Different SGA and MGA diverse from each other by the execution order of the main genetic operators – selection, crossover and mutation, or in the absence of mutation operator. The influence of the most important parameters of GA, namely NIND, MAXGEN, GGAP, XOVR and MUTR has been investigated for SGA, aiming at improving the accuracy of the solution and the convergence time of the algorithms. In addition,

the influence of INSR and MIGR has been explored only in the case of MGA. Among all investigated parameters, GGAP has been distinguished as the most sensitive one towards the convergence time, both for SGA and MGA. About 40% of the computational time can be saved without loss of model accuracy in the cases of SGA_mcs and MGA_csm, when GGAP = 0.5 is used instead of 0.9. No such significant influence on the algorithm's convergence time is observed when other GA parameters have been examined. However, NIND = 20, XOVR = 0.85, MUTR = 0.1/0.02 (for SGA/MGA), MAXGEN = 100, MIGR = 0.1 (for MGA only) and INSR = 0.9 (for MGA only) might be considered as a good compromise for model accuracy and time for finding the solution. Selected values of the GA along with the proposed modifications for SGA and MGA, improve the effectiveness of the algorithms and proof their successful implementation for solving complex nonlinear problems, as the parameter identification of fermentation process models.

References

1. Anwaar A., A. Ashraf, W. H. K. Bangyal, M. Iqbal (2022). Genetic Algorithms: Brief Review on Genetic Algorithms for Global Optimization Problems, Human-Centered Cognitive Systems (HCCS), Shanghai, China, 1-6.
2. Angelova M. (2014). Modified Genetic Algorithms and Intuitionistic Fuzzy Logic for Parameter Identification of Fed-batch Fermentation Process Model, PhD Thesis, Sofia. (in Bulgarian)
3. Angelova M., P. Melo-Pinto, T. Pencheva (2012). Modified Simple Genetic Algorithms Improving Convergence Time for the Purposes of Fermentation Process Parameter Identification, WSEAS Transactions on Systems, 11(7), 256-267.
4. Angelova M., S. Tzonkov, T. Pencheva (2010). Parameter Identification of a Fed-batch Cultivation of *S. cerevisiae* Using Genetic Algorithms, Serdica Journal of Computing, 4(1), 11-18.
5. Angelova M., S. Tzonkov, T. Pencheva (2009). Modified Multi-population Genetic Algorithm for Yeast Fed-batch Cultivation Parameter Identification, International Journal Bioautomation, 13(4), 163-172.
6. Angelova M., S. Tzonkov, T. Pencheva (2011). Genetic Algorithms Based Parameter Identification of Yeast Fed-batch Cultivation, Lecture Notes in Computer Science, 6046, 224-231.
7. Angelova M., T. Pencheva (2011). Tuning Genetic Algorithm Parameters to Improve Convergence Time, International Journal of Chemical Engineering, Article ID 646917.
8. Angelova M., T. Pencheva (2012). Algorithms Improving Convergence Time in Parameter Identification of Fed-batch Cultivation, Comptes rendus de l'Académie bulgare des Sciences, 65(3), 299-306.
9. Angelova M., T. Pencheva (2013). Improvement of Multi-population Genetic Algorithm Convergence Time, Monte Carlo Methods and Applications, 1-9.
10. Chipperfield A. J., P. Fleming, H. Pohlheim, C. M. Fonseca (1994). Genetic Algorithm Toolbox for Use with MATLAB, User's Guide v. 1.2, Department of Automatic Control and System Engineering, University of Sheffield, UK.
11. Goldberg D. (1989). Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wiley Publishing Company, Massachusetts.
12. Gupta D., S. Ghafir (2012). An Overview of Methods Maintaining Diversity in Genetic Algorithms, International Journal of Emerging Technology and Advanced Engineering, 2(5), 56-60.
13. Holland J. (1975). Adaptation in Natural and Artificial Systems: An introductory analysis with Application to Biology, University of Michigan Press, Control and Artificial Intelligence.

14. Hristozov I., T. Pencheva, B. Iliev, S. Tzonkov (1998). A Comparative Analysis of Genetic Algorithms and Conventional Optimization Procedures, Scientific Proceeding of Scientific-Technical Union of Mechanical Engineering, 3-25, 187-192. (in Bulgarian)
15. Katoch S., S. S. Chauhan, V. Kumar (2021). A Review on Genetic Algorithm: Past, Present, and Future, Multimedia Tools and Applications, 80, 8091-8126.
16. Kosev K., O. Roeva (2010). Modified Multi-population Genetic Algorithm with Improved Convergence, 8th National Young Scientific-Practical Session, Sofia, Bulgaria, May 10-11, 7-12. (in Bulgarian)
17. Obittko M., Genetic Algorithms, https://www.obitko.com/tutorials/genetic-algorithms/, (last accessed 2024/12/17).
18. Pencheva T., M. Angelova (2016). Modified Multi-population Genetic Algorithms for Parameter Identification of Yeast Fed-Batch Cultivation, Bulgarian Chemical Communications, 48(4), 713-719.
19. Pencheva T., O. Roeva, I. Hristozov (2006). Functional State Approach to Fermentation Processes Modelling, Prof. Marin Drinov Academic Publishing House, Sofia.
20. Roeva O. (2006). A Modified Genetic Algorithm for a Parameter Identification of Fermentation Processes, Biotechnology and Biotechnological Equipment, 20, 202-209.
21. Rowe J., I. East (1995). Direct Replacement: A Genetic Algorithm without Mutation Which Avoids Deception, Lecture Notes in Artificial Intelligence, 956, 41-48.

Sen. Assist. Prof. Maria Angelova, Ph.D.
E-mail: maria.angelova@biomed.bas.bg

Maria Angelova received M.Sc. Degree in Bioautomation from the Technical University of Sofia (2006) and Ph.D. Degree (2014) from the Institute of Biophysics and Biomedical Engineering (IBPhBME), Bulgarian Academy of Sciences (BAS). She was a coordinator of one and participant in six scientific projects. She has published more than 50 scientific papers in renowned international journals with about 400 known citations. Her scientific interests are in the area of artificial intelligence, *in silico* drug design, and modelling and optimization of biotechnological processes.

Prof. Tania Pencheva, Ph.D.
E-mail: tania.pencheva@biomed.bas.bg

Tania Pencheva received M.Sc. Degree in Bioengineering (1994) and Ph.D. Degree (2003) from the Technical University – Sofia. At present, she is a Professor at IBPhBME – BAS. She has published more than 230 publications including 25 books and book chapters, with more than 1300 citations. Her current scientific interests are in the fields of *in silico* drug design, modelling and optimization of bioprocess systems, bioinspired algorithms, data analysis.