InterCriteria Analysis of ACO and GA Hybrid Algorithms

Olympia Roeva[1], Stefka Fidanova[2], and Marcin Paprzycki[3]

[1]Institute of Biophysics and Biomedical Engineering
Bulgarian Academy of Science, Bulgaria
olympia@biomed.bas.bg
[2]Institute of Information and Communication Technology
Bulgarian Academy of Science, Bulgaria
stefka@parallel.bas.bg
[3]Systems Research Institute, Polish Academy of Sciences,
Warsaw and Management Academy, Warsaw, Poland
marcin.paprzycki@ibspan.waw.pl

Abstract. In this paper, the recently proposed approach for multicriteria decision making – InterCriteria Analysis (ICA) – is presented. The approach is based on the apparatus of the index matrices and the intuitionistic fuzzy sets. The idea of InterCriteria Analysis is applied to establish the relations and dependencies of considered parameters based on different criteria referred to various metaheuristic algorithms. A hybrid scheme using Genetic Algorithm (GA) and Ant Colony Optimization (ACO) is used for parameter identification of *E. coli* MC4110 fed-batch cultivation process model. In the hybrid GA-ACO, the GA is used to find feasible solutions to the considered optimization problem. Further ACO exploits the information gathered by GA. This process obtains a solution, which is at least as good as – but usually better than – the best solution devised by GA. Moreover, a comparison with both the conventional GA and ACO identification results is presented. Based on ICA the obtained results are examined and conclusions about existing relations and dependencies between model parameters of the *E. coli* process and algorithms parameters and outcomes, such as number of individuals, number of generations, value of the objective function and computational time, are discussed.

Keywords: InterCriteria Analysis, meta-heuristics, hybrid algorithm, ant colony optimization, genetic algorithm, *E. coli* cultivation process.

1 Introduction

To solve different optimization problems we can apply various techniques and approaches, namely exact algorithms (Branch-and-Bound, Dynamic Programming, local search techniques) [14, 20, 39], heuristics [27, 35], and metaheuristics (Genetic Algorithms, Ant Colony Optimization, Particle Swarm Optimization,

Simulated Annealing, Tabu Search, etc.) [15, 17, 23]. Today, the use of metaheuristics has received more and more attention. These methods offer good solutions, even global optima, within reasonable computing time [38]. An even more efficient behavior and higher flexibility when dealing with real-world and large-scale problems, can be achieved through a combination of a meta-heuristic with other optimization techniques, the so-called hybrid metaheuristic [15, 22, 29, 30, 37, 36, 40].

The main goal of the hybrid algorithms is to exploit the advantages of different optimization strategies, avoiding their disadvantages. Choosing an adequate combination of metaheuristic techniques we can achieve a better algorithm performance in solving hard optimization problems. Developing such effective hybrid algorithm requires expertise from different areas of optimization. There are many hybridization techniques that have shown to be successful for different applications.

In this paper, we investigate a hybrid metaheuristic method that combines Genetic Algorithms (GA) and Ant Colony Optimization (ACO), named GA-ACO. There are some applications of ACO-GA hybrid for several optimization problems. In [25, 26] a hybrid metaheuristic ACO-GA for the problem of sports competition scheduling is presented. In the proposed algorithm first, GA generates activity lists thus provides the initial population for ACO. Next, ACO is executed. In the next step GA, based on the crossover and mutation operations, generates new population. continuous engineering optimization. Authors in [18] presented hybrid algorithm in that ACO and GA search alternately and cooperatively in the solution space. Test examples show that hybrid algorithm can be more efficient and robust than the traditional population based heuristic methods. In [2] the problem of medical data classification is discussed. Authors propose a hybrid GA-ACO and show the usefulness of the proposed approach on a number of benchmark real-world medical datasets. For solving NP-hard combinatorial optimization problems in [1] a novel hybrid algorithm combining the search capabilities of the ACO and GA is introduced. As a result a faster and better search algorithm capabilities is achieved.

Provoked by the promising results obtained from the use of hybrid GA-ACO algorithms, we propose a hybrid algorithm, i.e. collaborative combination between ACO and GA for model parameters optimization of *E. coli* cultivation process. The effectiveness of GA and ACO have already been demonstrated for model parameter optimization considering fed-batch cultivation processes [32]. Moreover, parameter identification of cellular dynamics models has especially become a research field of great interest. Robust and efficient methods for parameter identification are of key importance.

On the other hand, the recently proposed approach for multicriteria decision making – InterCriteria Analysis (ICA) – is applied for additional exploring of the used metaheuristic techniques. In here discussed case the *E. coli* model parameter estimates, number of individuals (chromosomes and ants), number of algorithm generations, corresponding algorithm accuracy and computational time are considered as user criteria. The ICA is applied with the aim to more

profoundly understand the nature of the criteria involved and discover on this basis existing correlations between the criteria themselves. The theory of ICA has been presented in details in [4], and in [9–12] it was further discussed and developed.

The paper is organized as follows. The problem formulation is given in Section 2. The proposed hybrid GA-ACO technique is described in Section 3. The background of the ICA is presented in Section 4. The numerical results and a discussion are presented in Section 5. Conclusion remarks are done in Section 6.

2 Problem Formulation

2.1 *E. coli* Fed-batch Fermentation Model

The mathematical model of the fed-batch cultivation process of *E. coli* is presented by the following non-linear differential equation system [33]:

$$\frac{dX}{dt} = \mu X - \frac{F_{in}}{V} X \tag{1}$$

$$\frac{dS}{dt} = -q_S X + \frac{F_{in}}{V}(S_{in} - S) \tag{2}$$

$$\frac{dV}{dt} = F_{in} \tag{3}$$

where

$$\mu = \mu_{max} \frac{S}{k_S + S} \tag{4}$$

$$q_S = \frac{1}{Y_{S/X}} \mu \tag{5}$$

X is the biomass concentration, [g/l];
S is the substrate concentration, [g/l];
F_{in} is the feeding rate, [l/h];
V is the bioreactor volume, [l];
S_{in} is the substrate concentration in the feeding solution, [g/l];
μ and q_S are the specific rate functions, [1/h];
μ_{max} is the maximum value of the specific growth rate, [1/h];
k_S is the saturation constant, [g/l];
$Y_{S/X}$ is the yield coefficient, [-].

For the model parameters identification, experimental data of an *E. coli* MC4110 fed-batch cultivation process are used. The experiments are performed in the Institute of Technical Chemistry, University of Hannover, Germany. The detailed description of the cultivation condition and experimental data could be found in [3, 31].

The fed-batch process starts at time $t = 6.68$ h, after batch phase. The initial liquid volume is 1350 ml. Before inoculation a glucose concentration of

2.5 g/l was established in the medium. Glucose in feeding solution is 100 g/l. The temperature was controlled at 35 °C, the pH at 6.9. The stirrer speed was set to 900 rpm and was increased to 1800 rpm, so that the dissolved oxygen concentration was never below 30%. The aeration rate was kept at 275 l/h and the carbon dioxide was measured in the exhaust gas. The process is stopped at time $t = 11.54$ h.

The bioreactor, as well as FIA measurement system is shown on Figure 1. The feed rate profile and the dynamics of the measured substrate concentration are presented, respectively on Figure 2 and Figure 3.

Fig. 1. Bioreactor and FIA measurement system

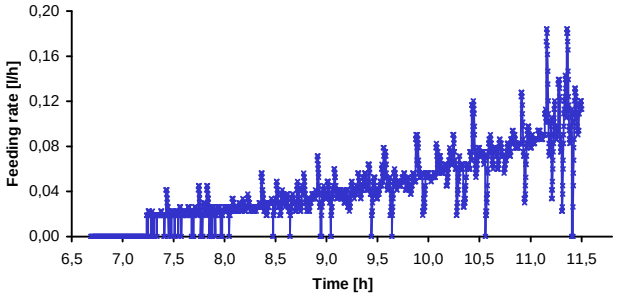

Fig. 2. Feed rate profile

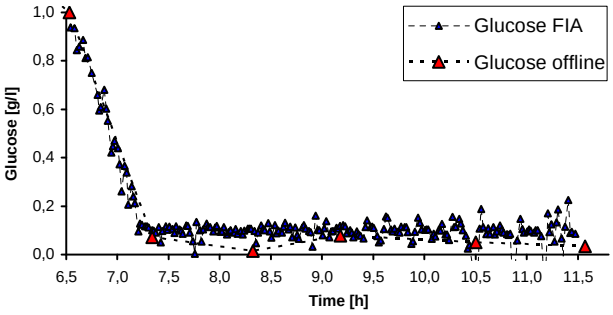

Fig. 3. Measured substrate concentration

For the considered non-linear mathematical model of *E. coli* fed-batch cultivation process (Eq. (1) - Eq. (5)) the parameters that should be identified are:

- maximum specific growth rate (μ_{max}),
- saturation constant (k_S),
- yield coefficient ($Y_{S/X}$).

The following upper and lower bounds of the model parameters are considered [32]:

$$0 < \mu_{max} < 0.7,$$

$$0 < k_S < 1,$$

$$0 < 1/Y_{S/X} < 30.$$

In the model identification procedures measurements of main process variables (biomass and glucose concentration) are used. For on-line glucose determination a FIA system has been employed. For biomass, off-line analysis are performed [3].

2.2 Optimization Criterion

The objective consists of adjusting the parameters (μ_{max}, k_S and $Y_{S/X}$) of the non-linear mathematical model function (Eq. (1) - Eq. (5)) to best fit a data set. The objective function is presented as a minimization of a distance measure J between experimental and model predicted values of the main state variables (biomass X and substrate S):

$$J = \sum_{i=1}^{m} \left(X_{\exp}(i) - X_{\,\mathrm{mod}}\,(i)\right)^2 + \\ + \sum_{i=1}^{m} \left(S_{\exp}(i) - S_{\,\mathrm{mod}}\,(i)\right)^2 \rightarrow \min \tag{6}$$

where m is the number of experimental data; $X_{\exp}$ and $S_{\exp}$ are the known experimental data for biomass and substrate; $X_{\,\mathrm{mod}}$ and $S_{\,\mathrm{mod}}$ are the model predictions for biomass and substrate with a given set of parameters (μ_{max}, k_S and $Y_{S/X}$).

3 Methodology

3.1 Genetic Algorithm

GA is a metaheuristic technique based on an analogy with the genetic structure and behaviour of chromosomes within a population of individuals using the following foundations [21]:

- chromosomes in a population compete for resources and mates;
- those chromosomes most successful in each competition will produce more off-spring than those chromosomes that perform poorly;
- genes from good chromosomes propagate throughout the population so that two good parents will sometimes produce offspring that are better than either parent;
- thus each successive generation will become more suited to their environment.

The structure of the GA, shown by the pseudocode is presented in Figure 4.

GA mainly operating on binary strings and using a recombination operator with mutation. GA support a population of chromosomes, $Pop(t) = x_1^t, ..., x_n^t$ for generation t. Each chromosome introduces a potential solution to the problem and is implemented as some data structure S. Each solution is evaluated according its "fitness". Fitness of a chromosome is assigned proportionally to the value of the objective function of the chromosomes. Then, a new population (generation $t + 1$) is formed by selecting better chromosomes (selection step).

Roulette wheel, developed by Holland [28] is most used selection method. The probability, P_i, for each chromosome to be selected is defined by:

$$P[\text{Individual } i \text{ is chosen}] = \frac{F_i}{\sum_{j=1}^{PopSize} F_j}, \tag{7}$$

where F_i equals the fitness of chromosome i and $PopSize$ is the population size.

Selected members of the new population have been subjected to transformations by means of "genetic" operators to form new solution. There are unary

Genetic Algorithm
$i = 0$
Initial population $Pop(0)$
Evaluate $Pop(0)$
while (not done) **do** (test for termination criterion)
    $i = i + 1$
    Select $Pop(i)$ from $Pop(i-1)$
    Recombine $Pop(i)$
    Mutate $Pop(i)$
    Evaluate $Pop(i)$
end while
Final solution

Fig. 4. Pseudocode for GA

transformations m_i (mutation type), which create new chromosomes by a small change in a single chromosome ($m_i : S \to S$), and higher order transformations c_j (crossover type), which create new chromosomes by combining parts from several chromosomes ($c_j : S \times \ldots \times S \to S$). The combined effect of selection, crossover and mutation gives so-called reproductive scheme growth equation (the schema theorem) [24]:

$$\xi(S, t+1) \geq$$
$$\xi(S, t) \cdot eval(S, t) / \bar{F}(t) \left[1 - p_c \cdot \frac{\delta(S)}{m-1} - o(S) \cdot p_m\right].$$

Good schemata receive an exponentially increasing number of reproductive trials in successive generations.

3.2 Ant Colony Optimization

The ACO is a stochastic optimization method that mimics the social behavior of real ants colonies, which try to find shortest rout to feeding sources and back. Real ants lay down quantities of pheromone (chemical substance) marking the path that they follow. An isolated ant moves essentially at random but an ant encountering a previously laid pheromone will detect it and decide to follow it with high probability and reinforce it with a further quantity of pheromone. The repetition of the above mechanism represents the auto-catalytic behavior of a real ant colony, where the more the ants follow a trail, the more attractive that trail becomes. The idea comes from observing the exploitation of resources of food among ants, in which ants have collectively been able to find the shortest path between to the food.

The ACO is implemented as a team of intelligent agents, which simulate the ants behavior, walking around the graph representing the problem to solve. The requirements of the ACO algorithm are as follows [16, 19]:

- The problem needs to be represented appropriately, which would allow the ants to incrementally update the solutions through the use of a probabilistic transition rules, based on the amount of pheromone in the trail and other problem specific knowledge.
- A problem-dependent heuristic function, that measures the quality of components that can be added to the current partial solution.
- A rule set for pheromone updating, which specifies how to modify the pheromone value.
- A probabilistic transition rule based on the value of the heuristic function and the pheromone value, that is used to iteratively construct a solution.

The structure of the ACO algorithm, shown by the pseudocode is presented in Figure 5.

The transition probability $p_{i,j}$, to choose the node j when the current node is i, is based on the heuristic information $\eta_{i,j}$ and the pheromone trail level $\tau_{i,j}$ of the move, where $i, j = 1, \ldots, n$.

$$p_{i,j} = \frac{\tau_{i,j}^a \eta_{i,j}^b}{\sum\limits_{k \in Unused} \tau_{i,k}^a \eta_{i,k}^b}, \tag{8}$$

where $Unused$ is the set of unused nodes of the graph.

The higher the value of the pheromone and the heuristic information, the more profitable it is to select this move and resume the search. In the beginning, the initial pheromone level is set to a small positive constant value τ_0; later, the ants update this value after completing the construction stage. The ACO algorithms adopt different criteria to update the pheromone level.

Ant Colony Optimization
Initialize number of ants;
Initialize the ACO parameters;
while not end-condition **do**
    for $k = 0$ **to** number of ants
        ant k choses start node;
        while solution is not constructed **do**
            ant k selects higher probability node;
        end while
    end for
    Update-pheromone-trails;
end while

Fig. 5. Pseudocode for ACO

The pheromone trail update rule is given by:

$$\tau_{i,j} \leftarrow \rho \tau_{i,j} + \Delta \tau_{i,j}, \tag{9}$$

where ρ models evaporation in the nature and $\Delta\tau_{i,j}$ is a new added pheromone which is proportional to the quality of the solution. Better solutions will receive more pheromone than others and will be more desirable in a next iteration.

3.3 Hybrid GA-ACO Algorithm

We proposed to combine two metaheuristics, namely GA [24, 28] and ACO [19]. GA is a population-based method where initial population is randomly generated. Thus generated initial solutions, further is genetically evaluated. ACO algorithm is a population-based too. The difference with GA is that ACO do not need initial population. ACO is a constructive method and we manage the ants to look for good solutions by parameter called pheromone. At the beginning the initial pheromone is the same for the elements of the all potential solutions. After every iteration the pheromone is updated. The elements of better solutions receive more pheromone then others and become more desirable in a next iterations. In our hybrid algorithm the solutions achieved by GA are like solutions achieved by ACO from some previous iteration and we update the initial pheromone according them. After that we continue with ACO algorithm.

The pseudocode of the proposed GA-ACO algorithm is shown in Figure 6.

```
GA-ACO hybrid algorithm
i = 0
Initial population Pop(0)
Evaluate Pop(0)
while not end-condition do
        i = i + 1
        Select Pop(i) from Pop(i − 1)
        Recombine Pop(i)
        Mutate Pop(i)
        Evaluate Pop(i)
end while
Final GA solution for ACO
Initialize number of ants;
Initialize the ACO parameters;
while not end-condition do
        for k = 0 to number of ants
            ant k choses start node;
            while solution is not constructed do
                    ant k selects higher probability node;
            end while
        end for
        Update-pheromone-trails;
end while
Final solution
```

Fig. 6. Pseudocode for Hybrid GA-ACO

4 InterCriteria Analysis

4.1 Short Remarks on Intuitionistic Fuzzy Pairs and Index Matrices

The Intuitionistic Fuzzy Pairs (IFPs) is an object in the form of an ordered pair

$$\langle a, b\rangle,$$

where $a, b \in [0, 1]$ and $a + b \leq 1$.

IFPs are used as an evaluation of some object or process, and the components (a and b) are interpreted, respectively, as degrees of membership and non-membership to a given set, or degrees of validity and non-validity, or degree of correctness and non-correctness, etc. [5].

Let us have two IFPs $x = \langle a, b\rangle$ and $y = \langle c, d\rangle$.

In [5] the following relations are defined:

$$\begin{array}{l} x < y \text{ iff } a < c \text{ and } b > d \\ x \leq y \text{ iff } a \leq c \text{ and } b \geq d \\ x = y \text{ iff } a = c \text{ and } b = d \\ x \geq y \text{ iff } a \geq c \text{ and } b \leq d \\ x > y \text{ iff } a > c \text{ and } b < d \end{array}$$

The concept of Index Matrix (IM) was introduced in [6] and discusssed in more details in [7, 8].

The basic definitions and properties related to IMs are follows [7]:

Let I be a fixed set of indices and $\mathcal{R}$ be the set of all real numbers. By IM with index sets K and L ($K, L \subset I$), we mean the object,

$$[K, L, \{a_{k_i,l_j}\}] \equiv \begin{array}{c|cccc} & l_1 & l_2 & \ldots & l_n \\ \hline k_1 & a_{k_1,l_1} & a_{k_1,l_2} & \ldots & a_{k_1,l_n} \\ k_2 & a_{k_2,l_1} & a_{k_2,l_2} & \ldots & a_{k_2,l_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_m & a_{k_m,l_1} & a_{k_m,l_2} & \ldots & a_{k_m,l_n} \end{array},$$

where

$$K = \{k_1, k_2, ..., k_m\},\ L = \{l_1, l_2, ..., l_n\},$$

and for $1 \leq i \leq m$, and $1 \leq j \leq n : a_{k_i,l_j} \in \mathcal{R}$.

On the basis of the above definition, in [8] the new object – the Intuitionistic Fuzzy IM (IFIM) – was introduced in the form

$$[K, L, \{\langle \mu_{k_i,l_j}, \nu_{k_i,l_j}\rangle\}]$$

$$\equiv \begin{array}{c|cccc} & l_1 & l_2 & \dots & l_n \\ \hline k_1 & \langle \mu_{k_1,l_1}, \nu_{k_1,l_1} \rangle & \langle \mu_{k_1,l_2}, \nu_{k_1,l_2} \rangle & \dots & \langle \mu_{k_1,l_n}, \nu_{k_1,l_n} \rangle \\ k_2 & \langle \mu_{k_2,l_1}, \nu_{k_2,l_1} \rangle & \langle \mu_{k_2,l_2}, \nu_{k_2,l_2} \rangle & \dots & \langle \mu_{k_2,l_n}, \nu_{k_2,l_n} \rangle \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_m & \langle \mu_{k_m,l_1}, \nu_{k_m,l_1} \rangle & \langle \mu_{k_m,l_2}, \nu_{k_m,l_2} \rangle & \dots & \langle \mu_{k_m,l_n}, \nu_{k_m,l_n} \rangle \end{array},$$

where for every $1 \leq i \leq m, 1 \leq j \leq n$: $0 \leq \mu_{k_i,l_j}, \nu_{k_i,l_j}, \mu_{k_i,l_j} + \nu_{k_i,l_j} \leq 1$, i.e., $\langle \mu_{k_i,l_j}, \nu_{k_i,l_j} \rangle$ is an IFP.

Let us have an IM

$$A = \begin{array}{c|cccccccc} & O_1 & \dots & O_k & \dots & O_l & \dots & O_n \\ \hline C_1 & a_{C_1,O_1} & \dots & a_{C_1,O_k} & \dots & a_{C_1,O_l} & \dots & a_{C_1,O_n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ C_i & a_{C_i,O_1} & \dots & a_{C_i,O_k} & \dots & a_{C_i,O_l} & \dots & a_{C_i,O_n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ C_j & a_{C_j,O_1} & \dots & a_{C_j,O_k} & \dots & a_{C_j,O_l} & \dots & a_{C_j,O_n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ C_m & a_{C_m,O_1} & \dots & a_{C_m,O_k} & \dots & a_{C_m,O_l} & \dots & a_{C_m,O_n} \end{array}, \tag{10}$$

where for every p, q, $(1 \leq p \leq m, 1 \leq q \leq n)$:

- C_p is a criterion, taking part in the evaluation,
- O_q is an object, being evaluated.
- a_{C_p,O_q} is a real number or another object, that is comparable about relation R with the other a-objects, so that for each i, j, k: $R(a_{C_k,O_i}, a_{C_k,O_j})$ is defined. Let $\overline{R}$ be the dual relation of R in the sense that if R is satisfied, then $\overline{R}$ is not satisfied and vice versa. For example, if "R" is the relation "$<$", then $\overline{R}$ is the relation "$>$", and vice versa.

Let $S^{\mu}_{k,l}$ be the number of cases is which $R(a_{C_k,O_i}, a_{C_k,O_j})$ and $R(a_{C_l,O_i}, a_{C_l,O_j})$ are simultaneously satisfied. Let $S^{\nu}_{k,l}$ be the number of cases is which $R(a_{C_k,O_i}, a_{C_k,O_j})$ and $\overline{R}(a_{C_l,O_i}, a_{C_l,O_j})$ are simultaneously satisfied.

Obviously,

$$S^{\mu}_{k,l} + S^{\nu}_{k,l} \leq \frac{n(n-1)}{2}.$$

Now, for every k, l, such that $1 \leq k < l \leq m$ and for $n \geq 2$, we define

$$\mu_{C_k,C_l} = 2\frac{S^{\mu}_{k,l}}{n(n-1)}, \quad \nu_{C_k,C_l} = 2\frac{S^{\nu}_{k,l}}{n(n-1)}. \tag{11}$$

Therefore, $\langle \mu_{C_k,C_l}, \nu_{C_k,C_l} \rangle$ is an IFP. Now, we can construct the IM

$$\begin{array}{c|ccc} & C_1 & \dots & C_m \\ \hline C_1 & \langle \mu_{C_1,C_1}, \nu_{C_1,C_1} \rangle & \dots & \langle \mu_{C_1,C_m}, \nu_{C_1,C_m} \rangle \\ \vdots & \vdots & \ddots & \vdots \\ C_m & \langle \mu_{C_m,C_1}, \nu_{C_m,C_1} \rangle & \dots & \langle \mu_{C_m,C_m}, \nu_{C_m,C_m} \rangle \end{array}, \tag{12}$$

that determine the degrees of correspondence between criteria $C_1, ..., C_m$.

5 Numerical Results and Discussion

5.1 Model Parameters Identification of *E. coli* Fed-batch Fermentation Process

The theoretical background of the GA and ACO is presented in details [32]. For the considered here model parameter identification, we used real-value coded GA instead binary encoding. The type of the basic operators in GA are as follows:

- encoding – real-value,
- fitness function – linear ranking,
- selection function – roulette wheel selection,
- crossover function – extended intermediate recombination,
- mutation function – real-value mutation,
- reinsertion – fitness-based.

In the applied here ACO algorithm the problem is represented by graph and the artificial ants try to construct shortest path under some conditions. In our case the graph of the problem is represented by tripartite graph. There are not arcs inside a level and there are arcs between levels. Every level corresponds to one of the model parameters we identify (μ_{max}, k_S and $Y_{S/X}$).

To set to the optimal settings the parameters of the GA and ACO, several pre-tests, according considered here optimization problem, are performed.

The optimal settings of the GA and ACO parameters are summarized in Table 1 and Table 2.

Table 1. Parameters of GA

Parameter	**Value**
ggap	0.97
xovr	0.7
mutr	0.05
maximum generations (maxgen)	40
number of individuals (nind)	20
number of variables	3
inserted rate	100 %

Table 2. Parameters of ACO algorithm

Parameter	**Value**
number of ants (nind)	20
initial pheromone	0.5
evaporation	0.1
maximum generations (maxgen)	100

Computer specification to run all identification procedures are Intel Core i5-2329 3.0 GHz, 8 GB Memory, Windows 7 (64bit) operating system and Matlab 7.5 environment.

We perform 30 independent runs of the hybrid GA-ACO. The hybrid algorithm starts with population of 20 chromosomes. We use 40 generation to find solution. We take the achieved best GA solution to update ACO initial pheromone. Further ACO is used to obtain the best model parameters vector using 30 ants for 100 generations.

For comparison of hybrid performance pure GA and pure ACO are run (30 times) with parameters shown in Table 1 and Table 2.

The main numerical results, from parameter identification, are summarized in Table 3. The obtained average values of the model parameters (μ_{max}, k_S and $Y_{S/X}$) are summarized in Table 4.

Table 3. Results from model parameters identification procedures

Value	**Algorithm**	**Algorithm performance**	
		T, [s]	J
best	GA	67.5172	4.4396
	ACO	67.3456	4.9190
	GA-ACO	38.7812	4.3803
	ACO-GA	35.5212	4.4903
worst	GA	66.5968	4.6920
	ACO	66.6280	6.6774
	GA-ACO	41.4495	4.6949
	ACO-GA	35.3498	4.6865
average	GA	67.1370	4.5341
	ACO	69.5379	5.5903
	GA-ACO	39.4620	4.5706
	ACO-GA	36.1313	4.5765

Table 4. Parameters' estimations of the *E. coli* fed-batch cultivation process model

Value	**Algorithm**	**Model parameters**		
		μ_{max}	k_S	$1/Y_{S/X}$
average	GA	0.4857	0.0115	2.0215
	ACO	0.5154	0.0151	2.0220
	GA-ACO	0.4946	0.0123	2.0204
	ACO-GA	0.4976	0.0135	2.0221

As it can be seen from Table 3 the hybrid GA-ACO achieves similar to pure GA and pure ACO algorithm values of the objective function. In the same time, the running time of the proposed hybrid algorithm is about two times less. The pure ACO algorithm starts with equal initial pheromone for all problem

elements. In the case of hybrid GA-ACO we use the best found solution by the GA to update the ACO pheromone. Thus our ACO algorithm uses the GA "experience" and starts from "better" pheromone. This strategy helps to the ants to find good solutions using less computational resources like time and memory. In result our hybrid algorithm uses more than three times less memory than pure ACO and pure GA algorithms.

Moreover, in Table 3 we compare achieved in this work results with results in our previous work [34]. There we run the ACO algorithm for several iterations and thus we generate initial populations for GA algorithm. Thus the GA starts from population closer to the good (optimal) solution than the randomly generated population. We observe that ACO-GA and GA-ACO algorithms achieves very similar results for a similar running time. We run the ANOVA test to measure the relative difference between two algorithms. The two hybrid algorithms achieves statistically equivalent results, but the GA-ACO algorithm uses 30% less memory. Thus we can conclude that hybrid GA-ACO algorithm performs better than ACO-GA hybrid algorithm.

On Figure 7 the comparison of the dynamics of measured and modeled biomass concentration is shown. With a line we show the modeled biomass during the cultivation process and with stars we show the measured biomass concentration. We put only several stars because the two line are almost overlapped. On Figure 8 the comparison between time profiles of measured and modeled substrate concentration during the cultivation process is shown. On the both figures we observe how close are the modeled and measured data. Thus we show the quality of our hybrid GA-ACO algorithm.

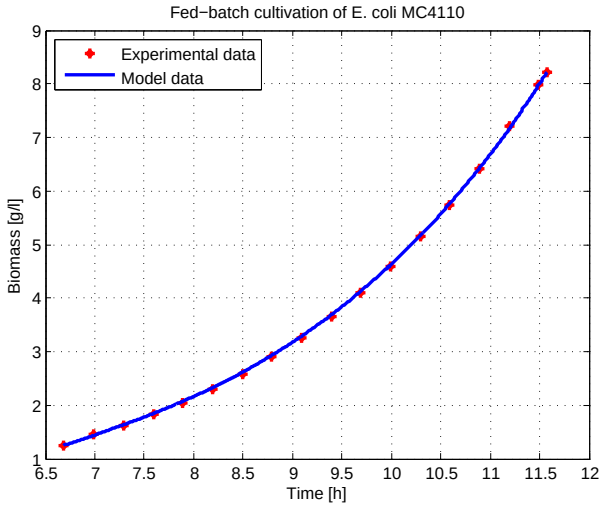

Fig. 7. Comparison between measured and modeled biomass concentration

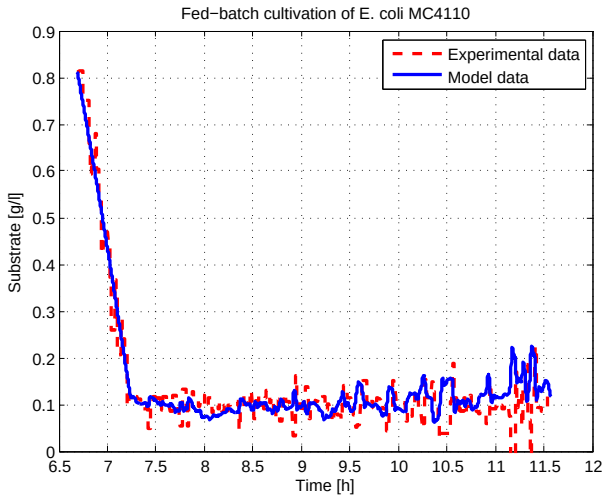

Fig. 8. Comparison between measured and modeled substrate concentration

5.2 InterCriteria Analysis

Based on the obtained results from the identification procedures the following IM is defined:

$$A = \begin{array}{c|cccc} & \text{GA} & \text{ACO} & \text{GA-ACO} & \text{ACO-GA} \\ \hline T_{ave} & 67.1370 & 69.5379 & 39.4620 & 36.1313 \\ J_{ave} & 4.5341 & 5.5903 & 4.5706 & 4.5765 \\ J_{best} & 4.4396 & 4.9190 & 4.3803 & 4.4903 \\ J_{worst} & 4.6920 & 6.6774 & 4.6949 & 4.6865 \\ \mu_{max_{ave}} & 0.4857 & 0.5154 & 0.4946 & 0.4976 \\ k_{S_{ave}} & 0.0115 & 0.0151 & 0.0123 & 0.0135 \\ 1/Y_{S/X_{ave}} & 2.0215 & 2.0220 & 2.0204 & 2.0221 \\ \text{nind} & 100 & 20 & 30 & 10 \\ \text{maxgen} & 100 & 20 & 30 & 10 \end{array} \tag{13}$$

In the IM A (13) the average values for computation time (T_{ave}) and for the three model parameters estimations ($\mu_{max_{ave}}$, $k_{S_{ave}}$ and $1/Y_{S/X_{ave}}$), in case of GA, ACO, GA-ACO and ACO-GA, are presented. Moreover, population number (individuals and/or ants) (nind), algorithm generations (maxgen) and the average, best and worst value of the objective function ($J_{ave}, J_{best}, J_{worst}$) (Eq. (6)) are considered.

Resulting IMs that determine the degrees of "agreement" (μ) and "disagreement" (ν) between criteria are follows:

μ	T_{ave}	J_{ave}	J_{best}	J_{worst}	$\mu_{max_{ave}}$	k_{Save}	$1/Y_{S/X_{ave}}$	nind	maxgen
T_{ave}	**1**	0.5	0.67	0.83	0.5	0.5	0.5	0.67	0.67
J_{ave}	0.5	**1**	0.83	0.67	1	1	0.67	0.17	0.33
J_{best}	0.67	0.83	**1**	0.5	0.83	0.83	0.83	0.33	0.5
J_{worst}	0.83	0.67	0.5	**1**	0.67	0.67	0.33	0.5	0.5
$\mu_{max_{ave}}$	0.5	1	0.83	0.67	**1**	1	0.67	0.17	0.33
k_{Save}	0.5	1	0.83	0.67	1	**1**	0.67	0.17	0.33
$1/Y_{S/X_{ave}}$	0.5	0.67	0.83	0.33	0.67	0.67	**1**	0.17	0.33
nind	0.67	0.17	0.33	0.5	0.17	0.17	0.17	**1**	0.5
maxgen	0.67	0.33	0.5	0.5	0.33	0.33	0.33	0.5	**1**

(14)

ν	T_{ave}	J_{ave}	J_{best}	J_{worst}	$\mu_{max_{ave}}$	k_{Save}	$1/Y_{S/X_{ave}}$	nind	maxgen
T_{ave}	**0**	0.5	0.33	0.17	0.5	0.5	0.5	0.33	0
J_{ave}	0.5	**0**	0.17	0.33	0	0	0.33	0.33	0.33
J_{best}	0.33	0.17	**0**	0.5	0.17	0.17	0.17	0.67	0.17
J_{worst}	0.17	0.33	0.5	**0**	0.33	0.33	0.67	0.5	0.17
$\mu_{max_{ave}}$	0.5	0	0.17	0.33	**0**	0	0.33	0.33	0.33
k_{Save}	0.5	0	0.17	0.33	0	**0**	0.33	0.33	0.33
$1/Y_{S/X_{ave}}$	0.5	0.33	0.17	0.67	0.33	0.33	**0**	0.33	0.33
nind	0.33	0.83	0.67	0.5	0.83	0.83	0.83	**0**	0.17
maxgen	0	0.33	0.17	0.17	0.33	0.33	0.33	0.17	**0**

(15)

As expected, every criteria perfectly correlates with itself, so the value μ is always 1, and $\nu = 0$. Also, the two matrices are obviously symmetrical according to the main diagonal. Observing obtained values of "agreement" (μ, Eq. (14)) and "disagreement" (ν, Eq. (15)), we can group the pairs of defined criteria in the following 6 groups.

- $\mu = 1$ and $\nu = 0$
  $\mu_{max_{ave}} - J_{ave}$, $k_{Save} - J_{ave}$, $k_{Save} - \mu_{max_{ave}}$

- $\mu = 0.83$ and $\nu = 0.17$
  $J_{worst} - T_{ave}$, $J_{best} - J_{ave}$, $\mu_{max_{ave}} - J_{best}$, $k_{Save} - J_{best}$, $1/Y_{S/X_{ave}} - J_{best}$

- $\mu = 0.67$ and $0 <= \nu <= 0.33$
  $J_{worst} - J_{ave}$, $nind - T_{ave}$, maxgen $- T_{ave}$, $1/Y_{S/X_{ave}} - J_{ave}$, $J_{best} - T_{ave}$

- $\mu = 0.5$ and $0.17 <= \nu <= 0.5$
  $J_{ave} - T_{ave}$, $\mu_{max_{ave}} - T_{ave}$, $k_{Save} - T_{ave}$, $1/Y_{S/X_{ave}} - T_{ave}$, $J_{worst} - J_{best}$, nind $- J_{worst}$, maxgen $- J_{best}$, maxgen $- J_{worst}$, maxgen $-$ nind

- $\mu = 0.33$ and $0.33 <= \nu <= 0.67$
  maxgen $- J_{ave}$, nind $- J_{best}$, $1/Y_{S/X_{ave}} - J_{worst}$, $\mu_{max_{ave}} -$ maxgen, $k_{Save} -$ maxgen, $1/Y_{S/X_{ave}} -$ maxgen

- $\mu = 0.17$ and $\nu <= 0.83$
  nind $- J_{ave}$, $\mu_{max_{ave}} -$ nind, $k_{Save} -$ nind, $1/Y_{S/X_{ave}} -$ nind

The high value of "agreement" of the parameters values from the first two groups confirms the robustness of the proposed algorithms. At the same time the strong connection in pair $k_{Save} - \mu_{max_{ave}}$ derives from the physical meaning of these model parameters [13]. In most of the cases the sum of the values of μ and ν is 1. However, sometimes this sum is less than 1, therefore there is some uncertainty. This uncertainty could be explained with the stochastic nature of the applied algorithms. The values of μ in the 4^{th} group show the correctness of the algorithms. The value of "agreement" between the average value of the objective function and computation time, or worst and best value of the objective function is not very high, but it exists. The last group shows that the dependence between number of population and achieved objective function value is less important than with running time. The running time depends on both – the number of population and the number of algorithm generations. Thus we can conclude that for achieving good results the balance between number of population and number of generations is very important. We observe that the worst value of the objective function depends much more of the running time than the best objective function value. If we run the algorithm for a short running time with high probability we will achieve bad solutions only, at the same time long running time can not guarantee achieving of good solutions.

On Figure 9 with stars are shown the pairs $\langle \mu, \nu \rangle$. When there is not uncertainty, the stars are on the diagonal. In the case of uncertainty the stars are under the diagonal. The uncertainties are in 30% of the cases and thus the robustness of our algorithms is confirmed.

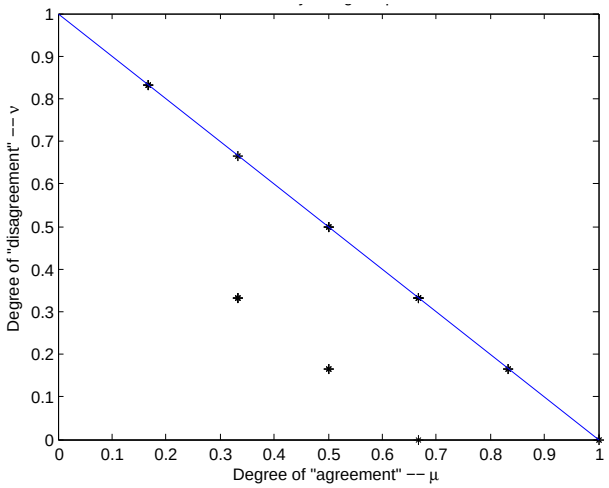

Fig. 9. Intuitionistic fuzzy triangle representation

6 Conclusion

In this paper we apply a new approach – InterCriteria analysis – for establishing relations and dependencies between different algorithm parameters and model parameters. We considered two hybrid algorithms – GA-ACO and ACO-GA – as well as pure GA and pure ACO algorithms.

First, algorithms are applied for parameter identification of nonlinear mathematical model of *E. coli* fed-batch cultivation process. We observe that our hybrid algorithms (GA-ACO and ACO-GA) achieve similar to pure GA and ACO algorithms solutions using less computational resources like time and memory. Both hybrid algorithms achieve statistically similar results for a similar running time, but GA-ACO algorithm uses 30% less memory, which is important when we solve large problems.

Second, InterCriteria analysis is performed to determine the levels of dependence between *E. coli* process model parameters themselves. Then between algorithms outcomes as computational time and accuracy, as well as the number of used populations and maximum number of algorithms generations. Next we determine the levels of dependence between *E. coli* process model parameters and the considered algorithms' parameters. This analysis shows some relations and dependencies that result from the physical meaning of the model parameters - on the one hand, and from stochastic nature of the considered metaheuristics - on the other hand. Moreover, the results show the robustness of the proposed algorithms (both hybrid and pure techniques) and confirm their correctness.

Acknowledgment

Work presented here is a part of the Poland-Bulgarian collaborative Grant "Parallel and distributed computing practices" and the Bulgarian National Scientific Fund under the grants DFNI-I02/20 Efficient ParallelAlgorithms for Large Scale Computational Problems" and DFNI-I02/5 "InterCriteria Analysis. A New Approach to Decision Making.

References

1. A. Acan, "A GA + ACO Hybrid for Faster and Better Search Capability", In: Ant Algorithms: Proc. of the Third International Workshop, ANTS 2002, Lecture Notes in Computer Science, 2002.
2. S. AlMuhaideb and M. El B. Menai, "A New Hybrid Metaheuristic for Medical Data Classification", Int. J. of Metaheuristics, Vol. 3(1), 2014, pp. 59–80.
3. M. Arndt and B. Hitzmann, "Feed Forward/feedback Control of Glucose Concentration during Cultivation of *Escherichia coli*", 8th IFAC Int. Conf. on Comp. Appl. in Biotechn, Canada, 2001, pp. 425–429.
4. K. Atanassov, D. Mavrov, V. Atanassova, "Intercriteria Decision Making: A New Approach for Multicriteria Decision Making, Based on Index Matrices and Intuitionistic Fuzzy Sets", Issues in IFSs and GNs, Vol. 11, 2014, 1-8.

5. K. Atanassov, E. Szmidt, J. Kacprzyk, "In intuitionistic fuzzy pairs", Notes on Intuitionistic Fuzzy Sets, Vol. 19, 2013, No. 3, 1–13.
6. K. Atanassov, "Generalized index matrices", Comptes rendus de l'Academie Bulgare des Sciences, Vol. 40, 1987, No. 11, 15–18.
7. K. Atanassov, "On index matrices, Part 1: Standard cases", Advanced Studies in Contemporary Mathematics, Vol. 20, 2010, No. 2, 291–302.
8. K. Atanassov, "On index matrices, Part 2: Intuitionistic fuzzy case", Proceedings of the Jangjeon Mathematical Society, Vol. 13, 2010, No. 2, 121–126.
9. V. Atanassova, D. Mavrov, L. Doukovska, K. Atanassov, "Discussion on the threshold values in the InterCriteria Decision Making approach", Notes on IFS, Vol. 20, 2014, No. 2, 94–99.
10. V. Atanassova, L. Doukovska, K. Atanassov, D. Mavrov, "Intercriteria Decision Making Approach to EU Member States Competitiveness Analysis", BMSD, 2014, 289–294.
11. V. Atanassova, L. Doukovska, D. Karastoyanov, F. Capkovic, "InterCriteria Decision Making Approach to EU Member States Competitiveness Analysis: Trend Analysis", IEEE 2014, 107–115.
12. V. Atanassova and I. Vardeva, "Sum- and average-based approach to criteria shortlisting in the InterCriteria Analysis", "Notes on IFS", Vol. 20, 2014, No. 4, 41–46.
13. G. Bastin and D. Dochain, "On-line Estimation and Adaptive Control of Bioreactors", Els. Sc. Publ., 1991.
14. M. Battarra , A. A. Pessoa, A. Subramanian and E. Uchoa, "Exact Algorithms for the Traveling Salesman Problem with Draft Limits", European Journal of Operational Research, Volume 235, Issue 1, 2014, pp. 115–128.
15. C. Blum and A. Roli, "Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison", ACM Computing Surveys, Vol. 35(3), 2003, pp. 268-308.
16. E. Bonabeau, M. Dorigo and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, New York,Oxford University Press, 1999.
17. I. Boussaid, J. Lepagnot and P. Siarry, "A Survey on Optimization Metaheuristics", Information Sciences, Vol. 237, 2013, pp. 82-117.
18. A. Csebfalv, "A Hybrid Meta-heuristic Method for Continuous Engineering Optimization", Civil Engineering, Vol. 53/2, 2009, pp. 93-100.
19. M. Dorigo and T. Stutzle, *Ant Colony Optimization*, MIT Press, 2004.
20. I. Dumitrescu and T. Sttzle, "Combinations of Local Search and Exact Algorithms", G.R. Raidl (Ed.) et al., Applications of Evolutionary Computation, Lecture Notes in Computer Science, Vol. 2611, 2003, pp. 211-223.
21. Genetic Algorithms, `http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/hmw/article1.html` (last accessed January 14, 2015)
22. A. Georgieva and I. Jordanov, "Hybrid Metaheuristics for Global Optimization using Low-discrepancy Sequences of Points", Computers and Operation Research, Vol. 37(3), 2010, pp. 456–469.
23. F. Glover and G. Kochenberger (Eds.), "Handbook of Metaheuristics", International Series in Operations Research and Management Science, Kluwer Academic Publishers, Vol. 57, 2003.
24. D. E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison Wesley Longman, London, 2006.
25. H. Guangdong, P. Ling and Q. Wang, "A Hybrid Metaheuristic ACO-GA with an Application in Sports Competition Scheduling", Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, Vol. 3, 2007, pp. 611–616.

26. H. Guangdong and Q. Wang, "A Hybrid ACO-GA on Sports Competition Schedulingby Ant Colony Optimization - Methods and Applications", Edited by Avi Ostfeld, 2011, pp. 89–100.
27. N. Harvey, "Use of Heuristics: Insights from Forecasting Research", Thinking & Reasoning, Vol. 13 Issue 1, 2007, pp. 5–24.
28. J. H. Holland, "Adaptation in Natural and Artificial Systems", 2nd Edn. Cambridge, MIT Press, 1992.
29. M. Lukasiewycz, M. Gla, F. Reimann, and J. Teich, "Opt4J - A Modular Framework for Meta-heuristic Optimization", In Proc. of the Genetic and Evolutionary Computing Conference (GECCO 2011), Dublin, Ireland, 2011, pp. 1723-1730.
30. S. Masrom, S. Z. Z. Abidin, P. N. Hashimah, and A. S. Abd. Rahman, "Towards Rapid Development of User Defined Metaheuristics Hybridisation", International Journal of Software Engineering and Its Applicatons, Vol. 5, 2011.
31. O. Roeva, T. Pencheva, B. Hitzmann, St. Tzonkov, "A Genetic Algorithms Based Approach for Identification of Escherichia coli Fed-batch Fermentation", Int. J. Bioautomation, Vol. 1, 2004, pp. 30–41.
32. O. Roeva and S. Fidanova, "Chapter 13. A Comparison of Genetic Algorithms and Ant Colony Optimization for Modeling of E. coli Cultivation Process", In book "Real-World Application of Genetic Algorithms", O. Roeva (Ed.), InTech, 2012, pp. 261–282.
33. O. Roeva, "Improvement of Genetic Algorithm Performance for Identification of Cultivation Process Models", Advanced Topics on Evolutionary Computing, Book Series: Artificial Intelligence Series – WSEAS, 2008, pp. 34–39.
34. O. Roeva, S. Fidanova, V. Atanassova, "Hybrid ACO-GA for Parameter Identification of an E. coli Cultivation Process Model", Large-Scale Scientific Computing, Lecture Notes in Computer Science 8353, Springer, Germany, ISSN 0302-9743, 2014, 288–295.
35. H. Smith, "Use of the Anchoring and Adjustment Heuristic by Children", Current Psychology: A Journal For Diverse Perspectives On Diverse Psychological Issues, Vol. 18, Issue 3, 1999, pp. 294–300.
36. E. G. Talbi and El-ghazali (Ed.), "Hybrid Metaheuristics", Studies in Computational Intelligence, Vol. 434, 2013, XXVI, 458 p. 109 illus.
37. E. G. Talbi, "A Taxonomy of Hybrid Metaheuristics", Journal of Heuristics, 8, 2002, pp. 541-564.
38. J. Toutouh, "Metaheuristics for Optimal Transfer of P2P Information in VANETs", MSc Thesis, University of Luxembourg, 2010.
39. G. J. Woeginger, "Exact Algorithms for NP-Hard Problems: A Survey", Lecture Notes in Computer Science, Volume 2570, 2003, pp. 185–207.
40. H. Yi, Q. Duan and T. Warren Liao, "Three Improved Hybrid Metaheuristic Algorithms for Engineering Design Optimization", Applied Soft Computing, Vol. 13(5), 2013, pp. 2433-2444.