Hindawi Publishing Corporation
Modelling and Simulation in Engineering
Volume 2011, Article ID 496732, 21 pages
doi:10.1155/2011/496732

Research Article

Investigation on Evolutionary Computation Techniques of a Nonlinear System

Tran Trong Dao[1, 2]

[1] *Faculty of Information Technology and Applied Mathematics, Ton Duc Thang University, Nguyen Huu Tho Street, Tan Phong Ward, District 7, Ho Chi Minh City, Vietnam*
[2] *Office of Science-Technology, Cooperation and Postgraduate Studies, Ton Duc Thang University, Nguyen Huu Tho Street, Tan Phong Ward, District 7, Ho Chi Minh City, Vietnam*

Correspondence should be addressed to Tran Trong Dao, trantrongdao@tdt.edu.vn

Received 21 March 2011; Revised 23 June 2011; Accepted 4 August 2011

Academic Editor: Luis Carlos Rabelo

The main aim of this work is to show that such a powerful optimizing tool like evolutionary algorithms (EAs) can be in reality used for the simulation and optimization of a nonlinear system. A nonlinear mathematical model is required to describe the dynamic behaviour of batch process; this justifies the use of evolutionary method of the EAs to deal with this process. Four algorithms from the field of artificial intelligent—differential evolution (DE), self-organizing migrating algorithm (SOMA), genetic algorithm (GA), and simulated annealing (SA)—are used in this investigation. The results show that EAs are used successfully in the process optimization.

1. Introduction

Evolutionary computation (EC) techniques, which are based on a powerful principle of evolution: survival of the fittest, constitute an interesting category of heuristic search. Evolutionary computation techniques are stochastic algorithms whose search methods model some natural phenomena: genetic inheritance and Darwinian strive for survival. The best known algorithms in this class include genetic algorithms, evolutionary programming, evolution strategies, and genetic programming. There are also many hybrid systems which incorporate various features of the above paradigms and consequently are hard to classify; anyway, we refer to them just as evolutionary computation methods [1].

In computer science, evolutionary computation is a subfield of artificial intelligence that involves combinatorial optimization problems.

Nowadays, Optimization is one of these words which is used almost every day in different fields of human activities. Everybody wants to maximize profit and minimize cost. This means optimizing in every task of industry, transportation, medicine, everywhere. For these purposes, we need to have suitable tools which are able to solve very difficult and complicated problems. As previous years proved, use of artificial intelligence and soft computing contribute to improvements in a lot of activities. One of such tools of soft computing are evolutionary algorithms [2].

In this paper, the modeling of a dynamic chemical engineering process is presented in a highly understandable way using a unique combination of the simplified fundamental theory and direct hands-on computer simulation. A nonlinear mathematical model is required to describe the dynamic behaviour of batch process; this justifies the use of evolutionary method of the EAs to deal with this process, for static optimization of a chemical batch reactor. Consequently, it is used to design geometry technique equipments for chemical reaction. The method was used to optimize the design of the growth chamber and was found to be in good agreement with the observed growth rate results.

Here, EAs were used to investigative and optimize batch reactor to improve its parameters. Consequently, EAs are used to model the technical requirements for chemical reaction. The optimized reactor was used in a simulation with optimization by evolutionary algorithms, and the results are presented in graphs.

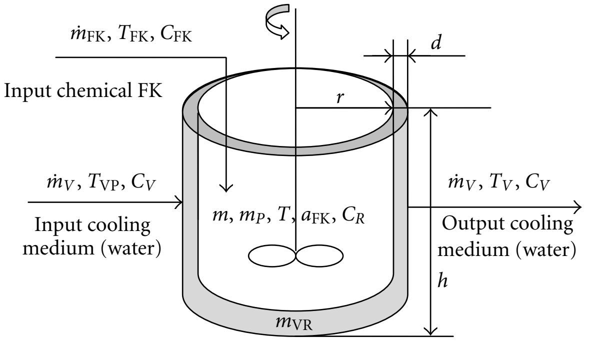

$\dot{m}_{\mathrm{FK}}, T_{\mathrm{FK}}, C_{\mathrm{FK}}$
Input chemical FK
d
r
$\dot{m}_V, T_{\mathrm{VP}}, C_V$
Input cooling medium (water)
$m, m_P, T, a_{\mathrm{FK}}, C_R$
$\dot{m}_V, T_V, C_V$
Output cooling medium (water)
h
m_{VR}

Figure 1: Batch reactor.

Research on this work is concerned with the field of optimization of chemical engineering through EAs. The main purposes and goals of the research can be summarized as thus:

(i) description and analysis of the chosen dynamic system more concretely those in the processes of a batch reactor,

(ii) proposing a set of solving algorithms for the application of stochastic optimization, which enhances confidence in the optimization results, particularly in the chemical reaction,

(iii) selecting and demonstrating EAs and practical method to optimize the chemical process, especially of batch reactor,

(iv) demonstrating the use of designed algorithms for global optimization of the chemical process and comparison between each selected algorithm, and

(v) presenting conclusions and suggesting further research perspective.

2. Description of a Reactor

This work uses a mathematical model of a reactor shown in Figure 1. The vessel with a double side for cooling medium is further equipped with stirrer for mixing reaction mixtures.

Reactor includes two physical inputs. First input denoted "input chemical FK" that substances into reaction with mass flow rate $\dot{m}_{\mathrm{FK}}$, temperature T_{FK}, and specific heat c_{FK}. Second input denoted "input cooling medium" that water drain into the reactor double side with mass flow rate $\dot{m}_V$, temperature T_{VP}, and specific heat c_V. This coolant further traverses among-jacketed through space of reaction, and its total weight in this space is m_{VR}. Coolant after that gets off the exit reaction denoted "output cooling medium" about mass flow rate $\dot{m}_V$, temperature T_V, and specific heat c_V. At the beginning of the process, there is an initial batch inside the reactor with parameter mass m_P. Reaction mixture then has total mass m, temperature T, and specific heat c_R and stirs till the time chemicals FK described by parameter concentration a_{FK}.

2.1. Nonlinear Model of Reactor. Description of the reactor applies a system of four balance equations. The first one expresses a mass balance of reaction mixture inside the reactor, the second a mass balance of the chemical FK, and the last two formulate enthalpic balances, namely, balances of reaction mixture and cooling medium. Equation (1), which for simplified notation of basic equations (2), is represented by term "k" as follows:

$$\dot{m}_{\mathrm{FK}} = m'[t], \tag{1}$$

$$\begin{aligned}
\dot{m}_{\mathrm{FK}} &= m[t]a'_{\mathrm{FK}}[t] + km[t]a_{\mathrm{FK}}[t], \\
\dot{m}_{\mathrm{FK}}c_{\mathrm{FK}}T_{\mathrm{FK}} + \Delta H_r km[t]a_{\mathrm{FK}}[t] &= KS(T[t] - T_V[t]) \\
&\quad + m[t]c_R T'[t], \\
\dot{m}_V c_V T_{\mathrm{VP}} + KS(T[t] - T_V[t]) &= \dot{m}_V c_V T_V[t] \\
&\quad + m_{\mathrm{VR}} c_V T'_V[t], \\
k &= \mathrm{A}e^{-\mathrm{E}/RT[t]}.
\end{aligned} \tag{2}$$

After modification into the standard form, the balance equations are obtained in the form (3):

$$\begin{aligned}
m'[t] &= \dot{m}_{\mathrm{FK}}, \\
a'_{\mathrm{FK}}[t] &= \frac{\dot{m}_{\mathrm{FK}}}{m[t]} - \mathrm{A}e^{-\mathrm{E}/RT[t]}a_{\mathrm{FK}}[t], \\
T'[t] &= \frac{\dot{m}_{\mathrm{FK}}c_{\mathrm{FK}}T_{\mathrm{FK}}}{m[t]c_R} + \frac{\mathrm{A}e^{-\mathrm{E}/RT[t]}\Delta H_r a_A[t]}{c_R} \\
&\quad - \frac{KST[t]}{m[t]c_R} + \frac{KST_V[t]}{m[t]c_R}, \\
T'_V[t] &= \frac{\dot{m}_V T_{\mathrm{VP}}}{m_{\mathrm{VR}}} + \frac{KST[t]}{m_{\mathrm{VR}}c_V} - \frac{KST_V[t]}{m_{\mathrm{VR}}c_V} - \frac{\dot{m}_V T_V[t]}{m_{\mathrm{VR}}}.
\end{aligned} \tag{3}$$

The design of the reactor was based on standard chemical-technological methods and gives a proposal of reactor physical dimensions and parameters of chemical substances. These values are called in this participation expert parameters. The objective of this part of the work is to perform a simulation and optimization of the given reactor.

Therefore, into system equations (3) were instated constants:

$A = 219{,}588\,\mathrm{s}^{-1}$, $E = 29967{,}5087\,\mathrm{J\cdot mol^{-1}}$, $R = 8{,}314\,\mathrm{J\cdot mol^{-1}\cdot K^{-1}}$, $c_{\mathrm{FK}} = 4400\,\mathrm{J\cdot kg\cdot K^{-1}}$, $c_V = 4118\,\mathrm{J\cdot kg\cdot K^{-1}}$, $c_R = 4500\,\mathrm{J\cdot kg\cdot K^{-1}}$, $\Delta H_r = 1392350\,\mathrm{J\cdot kg^{-1}}$, $K = 200\,\mathrm{kg\cdot s^{-3}\cdot K^{-1}}$.

Next parameters, which are important for calculations, are

(i) geometric dimension of the reaction: $r[m]$, $h[m]$,

(ii) density of chemicals: $\rho_P = 1203\,\mathrm{kg\cdot m^{-3}}$, $\rho_{\mathrm{FK}} = 1050\,\mathrm{kg\cdot m^{-3}}$,

(iii) stoichiometric rate chemical: $m_P = 2{,}82236\cdot m_{\mathrm{FK}}$.

2.2. Mathematical Problems. This optimization was found by optimized parameters with one another linked, which includes heat transfer surface, volume, and mass mixtures of reaction. Heat transfer surface S has the relation:

$$S = 2\pi rh + \pi r^2, \tag{4}$$

where r is radius and h is height of the space reactor (see Figure 1).

Volume of vessel of rector applies to the relation:

$$V = \pi r^2 h. \tag{5}$$

Total mass of mixtures in the reaction mass m_P and mass input chemical FK m_{FK}, is:

$$m = m_P + m_{\mathrm{FK}}. \tag{6}$$

The stechiometric ratio is given by

$$m_P = 2,82236 m_{\mathrm{FK}}. \tag{7}$$

Total volume of mixtures in the reaction equals sum of volume initial mixtures in the reaction and volume of FK:

$$V = V_P + V_{\mathrm{FK}} = \frac{m_P}{\rho_P} + \frac{m_{\mathrm{FK}}}{\rho_{\mathrm{FK}}}. \tag{8}$$

The relationship between the optimized volume of reactor and the mass of added chemical FK is given by (8). Then substituting to (7) gives the mass of the initial batch in the reactor:

$$m_{\mathrm{FK}} = \frac{\rho_P \rho_{\mathrm{FK}} V}{2,82236 \rho_{\mathrm{FK}} + \rho_P}. \tag{9}$$

In this example, the optimization was then added parameter thickness d of vessel, which has relation that is

$$m_{\mathrm{VR}} = \rho_V S d. \tag{10}$$

To minimize the area arising as a difference between the required and real-temperature profile of the reaction mixture in a selected time interval, which was the duration of a batch cycle? The required temperature is 97°C (370.15 K). The cost function that is minimized gives:

$$f_{\cos t} = \sum_{t=0}^{t} |w - T[t]|. \tag{11}$$

3. Methods and Evolutionary Algorithms

3.1. Introduction and Using Evolutionary Algorithms. As the history of the field suggests, there are many different variants of evolutionary algorithms. The common underlying idea behind all these techniques is the same: given a population of individuals, the environmental pressure causes natural selection (survival of the fittest) and this causes a rise in the fitness of the population. Given a quality function to be maximized, we can randomly create a set of candidate solutions, that is, elements of the function's domain, and apply the quality function as an abstract fitness measure—the higher the better. Based on this fitness, some of the better candidates are chosen to seed the next generation by applying recombination and/or mutation to them. Recombination is an operator applied to two or more selected candidates (the so-called parents) and results in one or more new candidates (the children). Mutation is applied to one candidate and results in one new candidate. Executing recombination and mutation leads to a set of new candidates (the offspring) that compete—based on their fitness (and possibly age)—with the old ones for a place in the next generation. This process can be iterated until a candidate with sufficient quality (a solution) is found or a previously set computational limit is reached. In this process there are two fundamental forces that form the basis of evolutionary systems.

Variation operators (recombination and mutation) create the necessary diversity, and thereby, facilitate novelty, while Selection acts as a force pushing quality.

The combined application of variation and selection generally leads to improving fitness values in consecutive populations. It is easy (although some-what misleading) to see such a process as if the evolution is optimising, or at least "approximising," by approaching optimal values closer and closer over its course. Alternatively, evolution is often seen as a process of adaptation. From this perspective, the fitness is not seen as an objective function to be optimised, but as an expression of environmental requirements. Matching these requirements more closely implies an increased viability, reflected in a higher number of offspring. The evolutionary process makes the population adapt to the environment better and better.

Let us note that many components of such an evolutionary process are stochastic. During selection fitter individuals have a higher chance to be selected than less fit ones, but typically even the weak individuals have a chance to become a parent or to survive. For recombination of individuals the choice of which pieces will be recombined is random. Similarly, for mutation, the pieces that will be mutated within a candidate solution, and the new pieces replacing them, are chosen randomly. The general scheme of an evolutionary algorithm is given in Pseudocode 1 in a pseudocode fashion [3, 4].

Evolutionary algorithms are a group of algorithms which use their special operators as mutation, crossover, and others to find an ideal solution. Possible candidates are defined by a cost function in which arguments are values of each solution. The best one is in the global extreme—maximum or minimum [2].

These evolutionary algorithms have been known for decades and live through the advancement from the weaker ones to more robust ones which are used with success in a lot of tasks nowadays. Since their first appearance, there is quite long queue of representatives: genetic algorithms, differential evolution, self-organizing migrating algorithm, particle swarm intelligence, ant Colony optimization, and artificial immune system. In optimization, algorithms belong also to some stochastic and deterministic ones: hill climbing, simulated annealing, Monte Carlo, and a lot of others or their mutations. These techniques promise fast optimization compared to classical mathematical approach. On the other hand, also between these optimization techniques it is possible to find better and worse. Their behaviour was described in a lot of references. And the research in this area is still full of white

```
BEGIN
  INITIALISE population with random candidate solutions;
  EVALUATE each candidate;
  REPEAT UNTIL (TERMINATION CONDITION is satisfied) DO
    1 SELECT parents;
    2 RECOMBINE pairs of parents;
    3 MUTATE the resulting offspring;
    4 EVALUATE new candidates;
    5 SELECT individuals for the next generation;
  OD
END
```

Pseudocode 1: The general scheme of an evolutionary algorithm in pseudocode.

places. There is wide field of possible applications as tuning of parameters, making of comparisons, trying to find new ones somehow [5].

Optimization algorithms—mainly evolutionary algorithms—are a necessary part of the above-described tools and can be used independently. Here, an overview only of algorithms, which were used in further simulations, will be given.

Division of optimization algorithms might be as follows (Figure 2). This is not the only one point of view on that [2].

3.2. A Brief Survey of Scoping and Screening Chemical Reaction Networks Using Stochastic Optimization. Many methods were adapted for the so-called optimal chemical reactor. From the research article of Marcoulaki and Kokossis [6] introduced about a brief survey of scoping and screening chemical reaction networks using stochastic optimization. The new methods focus on a systematic and thorough consideration of the available options and employ technology in the form of superstructures, optimization techniques, and a variety of graphical methods.

The importance of mathematical methods in optimizing reactor has been exemplified early enough with the application of dynamic programming for the estimation of optimal operating conditions in CSTR cascades [7] and the development of graphical techniques for single reversible reactions in PFRs (1961).

Around the same time, a set of brilliant contributions by Horn [8] provided the basis of material that later emerged as attainable-region (AR) approaches. Dyson and Horn [9] developed graphical tools for optimal temperature control schemes, feed distribution profiles along a PFR, and catalyst minimization problems [10]. In these early days, separate groups made attempts to consolidate options and alternatives within comprehensive reactor structures [11–13]. Optimization approaches initially addressed fixed reactor structures. Examples include the work of Paynter and Haskins [14] and Chitra and Govind [15–17]. The first studies of comprehensive structures should be attributed to Achenie and Biegler [18–20], who employed existing representations [11, 12] to launch optimization techniques in the form of NLP methods.

Kokossis and Floudas [21–23] first introduced the idea of a reactor network superstructure modeled and optimized as an MINLP formulation. Though general and inclusive, their representation did not follow previous developments, but made an effort to facilitate the functionalities of the MINLP technology with the synthesis objectives. Mainly to scope, optimize, and analyze the reaction process, Kokossis and Floudas replaced detailed models with simple though generic structures, enough to screen for design options and estimate the limiting performance of the reaction system. In the same vane, dynamic components have been replaced by CSTR cascades. A superstructure of generic elements (ideal CSTRs and PFRs) was postulated to account for all possible interconnections among the units. The representation was modeled and optimized as a MINLP model.

Though fundamental limitations appear evident, persistent efforts to extend the graphical methods have appeared in the literature [24–30].

A more promising direction has been pursued by Biegler and coworkers. The motivation has been to instill better guarantees in the optimization efforts by exploiting ideas and rules established in the construction of the AR. Applications presented in this area include the work by Balakrishna and Biegler [31, 32] and Lakshmanan and Biegler [33–35] and involved mathematical programming applications in the form of NLP and MINLP formulations. Optimal control formulation has been presented by Rojnuckarin et al. [36] and Schweiger and Floudas [37]. Hildebrandt and Biegler [38] presented a review of the attainable region approaches and suggested areas for future development of the concept.

Especially in recent years, the methods of artificial intelligence, namely, the evolutionary algorithms, were used to optimise successfully chemical processes. Evolutionary algorithms have been applied to the solution of NLP in many engineering applications. The best-known algorithms in this class include genetic algorithms (GA), evolutionary programming (EP), evolution strategies (ES), and genetic programming (GP). There are many hybrid systems, which incorporate various features of the above paradigms and consequently are hard to classify, which can be referred just as EC methods, Dasgupta and Michalewicz [39]. They differ from the conventional algorithms since, in general, only the information regarding the objective function is required.

In recent years, EC methods have been applied to a broad range of activities in process system engineering including modeling, optimization, and control. See, for example, real-time control of plasma reactor [40–42], optimization of reactive distillation processes using self-organizing migrating algorithm and differential evolution strategies [43], using method of artificial intelligence to optimise and control chemical reactor [44], investigation on optimization of process parameters and chemical reactor geometry by evolutionary algorithms [44], or an optimum solution for a process control problem (continuous stirred tank reactor) using a hybrid neural network [45].

4. Select Evolutionary Algorithms

For the experiments described here, stochastic optimisation algorithms, such as differential evolution (DE) [46], self-organizing migrating algorithm (SOMA) [47, 48], genetic algorithms (GA) [49], and simulated annealing (SA) [50, 51] were selected. Main reason why DE, SOMA, GA, and SA have been seed comes from contemporary state in chemical engineering and EAs use. Since now has been done some research with attention on use of EAs in chemical engineering optimization, including DE. This participation has to show that applicability of relatively new algorithms is also positive and can lead to applicable results, as was shown, for example, in Zelinka [2], which has been done under the 5th EU project RESTORM (acronym of Radically Environmentally Sustainable Tannery Operation by Resource Management), and main aim was to use EAs in chemical engineering processes. True is also that there is a plenty of other heuristic like particle swarm [52], scatter search [53], memetic algorithms, simulated annealing [50], and according to no free lunch theorem [54] is clear that each heuristic would be less or more applicable on example presented here. SOMA is a stochastic optimization algorithm that is modelled on the social behaviour of cooperating individuals [47, 48]. It was chosen because it has been proved that the algorithm has the ability to converge towards the global optimum [47, 48]. GA is one of the most modern paradigms for general problem solving. Genetic algorithms are more robust than existing directed search methods. Another important property of GA-based search methods is that they maintain population of potential solutions—all other methods process a single point of the search space like hill climbing method. Hill climbing methods provide local optimum values, and these values depend on the selection of the starting point. Also there is no information available on the relative error with respect to global optimum. To increase the success rate in the hill climbing method, it is executed for large number of randomly selected different starting points. On the other hand, GA is a multidirectional search maintaining a population of potential solutions and encourages information formation and exchange between these directions. Furthermore, SA is a generic probabilistic meta-algorithm for the global optimization problem, namely, locating a good approximation to the global optimum of a given function in a large search space. SA has been used in various combinatorial optimization problems and has been particularly successful in circuit design problems (see [50]).

4.1. Differential Evolution (DE). Differential evolution [46] is a population-based optimization method that works on real-number-coded individuals. For each individual $x_{i,G}$ in the current generation G, DE generates a new trial individual $x'_{i,G}$ by adding the weighted difference between two randomly selected individuals $x_{r1,G}$ and $x_{r2,G}$ to a third randomly selected individual $x_{r3,G}$. The resulting individual $x'_{i,G}$ is crossed over with the original individual $x_{i,G}$. The fitness of the resulting individual, referred to as perturbated vector $u_{i,G+1}$, is then compared with the fitness of $x_{i,G}$. If the fitness of $u_{i,G+1}$ is greater than the fitness of $x_{i,G}$, $x_{i,G}$ is replaced with $u_{i,G+1}$; otherwise, $x_{i,G}$ remains in the population as $x_{i,G+1}$. Deferential evolution is robust, fast, and effective with global optimization ability. It does not require that the objective function is differentiable, and it works with noisy, epistatic, and time-dependent objective functions (see Pseudocode 2).

There are some versions for optimization by mean differential evolution, and two standard versions of DE, concretely DERand1Bin, and DERand2Bin were chosen for optimization of chemical reactor.

4.2. Self-Organizing Migrating Algorithm (SOMA). SOMA is a stochastic optimization algorithm that is modelled on the social behaviour of co-operating individuals [47, 48]. It was chosen because it has been proved that the algorithm has the ability to converge towards the global optimum [47, 48]. SOMA works on a population of candidate solutions in loops called migration loops. The population is initialized randomly distributed over the search space at the beginning of the search. In each loop, the population is evaluated, and the solution with the highest fitness becomes the leader L. Apart from the leader, in one migration loop, all individuals will traverse the input space in the direction of the leader. Mutation, the random perturbation of individuals, is an important operation for evolutionary strategies (ES). It ensures the diversity amongst the individuals, and it also provides the means to restore lost information in a population. Mutation is different in SOMA compared with other ES strategies. SOMA uses a parameter called PRT to achieve perturbation. This parameter has the same effect for SOMA as mutation has for GA. The PRT vector defines the final movement of an active individual in search space.

The randomly generated binary perturbation vector controls the allowed dimensions for an individual. If an element of the perturbation vector is set to zero, then the individual is not allowed to change its position in the corresponding dimension. An individual will travel a certain distance (called the path length) towards the leader in n steps of defined

```
1. Input: $D, G_{\max}, NP \geq 4, F \in (0, 1+), CR \in [0,1]$, and initial bounds: $\vec{x}^{(lo)}, \vec{x}^{(hi)}$.
2. Initialize: $\begin{cases} \forall i \leq NP \wedge \forall j \leq D : x_{i,j,G=0} = x_j^{(lo)} + rand_j[0,1] \bullet (x_j^{(hi)} - x_j^{(lo)}) \\ i = \{1, 2, \ldots, NP\},\ j = \{1, 2, \ldots, D\},\ G = 0, rand_j[0,1] \in [0,1] \end{cases}$
  3. While $G < G_{\max}$
    $\forall i \leq NP$
      4. Mutate and recombine :
          4.1  $r_1, r_2, r_3 \in \{1, 2, \ldots., NP\}$, randomly selected, except : $r_1 \neq r_2 \neq r_3 \neq i$
          4.2  $j_{rand} \in \{1, 2, \ldots, D\}$, randomly selected once each $i$
        4.3 $\forall j \leq D, u_{j,i,G+1} = \begin{cases} x_{j,r_3,G} + F \cdot (x_{j,r_1,G} - x_{j,r_2,G}) \\ \qquad \text{if } (rand_j[0,1] < CR \vee j = j_{rand}) \\ x_{j,i,G} \text{ otherwise} \end{cases}$
      5. Select
        $\vec{x}_{i,G+1} = \begin{cases} \vec{u}_{i,G+1} & \text{if } f(\vec{u}_{i,G+1}) \leq f(\vec{x}_{i,G}) \\ \vec{x}_{i,G} & \text{otherwise} \end{cases}$
  $G = G + 1$
```

Pseudocode 2: Pseudocode of DE.

```
Input: $N$, Migrations, $PopSize \geq 2$, $PRT \in [0,1]$, $Step \in (0,1]$, MinDiv $\in (0,1]$,
       Path Length $\in$ (0,5], $Specimen$ with upper and lower bound $x_j^{(hi)}, x_j^{(lo)}$
Inicialization: $\begin{cases} \forall i \leq PopSize \wedge \forall j \leq N : x_{i,j,Migrations=0} = x_j^{(lo)} + rand_j[0,1] \bullet (x_j^{(hi)} - x_j^{(lo)}) \\ i = \{1, 2, \ldots, Migrations\}, j = \{1, 2, \ldots, N\}, Migrations = 0, rand_j[0,1] \in [0,1] \end{cases}$
  While $Migrations < Migrations_{\max}$
    $\forall \mathrm{i} \leq PopSize$
      $While \quad t \leq PathLength$
      $if \quad rnd_j < PRT \quad pak \quad PRTVector_j = 1 \quad else \quad 0, \quad j = 1, \ldots, N$
      $x_{i,j}^{ML+1} = x_{i,j,start}^{ML} + (x_{L,j}^{ML} - x_{i,j,start}^{ML}) \quad t \quad PRTVector_j$
      $f(x_{i,j}^{ML+1}) = \text{if } f(x_{i,j}^{ML}) \leq f(x_{i,j,start}^{ML}) \text{ else } f(x_{i,j,start}^{ML})$
      $t = t + Step$
  $Migrations = Migrations + 1$
```

Pseudocode 3: Pseudocode of SOMA.

length. If the path length is chosen to be greater than one, then the individual will overshoot the leader. This path is perturbed randomly. For an exact description of use of the algorithms see [47, 48] for SOMA. Pseudocode of SOMA is (as shows in Pseudocode 3).

Now a day, there are some version of algorithm SOMA. In this work I have used two strategies of SOMA for optimization and predictive control of a chemical reactor. They are "All to One" (SOMAATO), that is, the worst version and "All to One Random" (SOMAATOR), that is, the best version of SOMA.

There are some version of algorithm SOMA. In this work I have used two strategies of SOMA for optimization and predictive control of a chemical reactor. They are "All to One" (SOMAATO), that is, the worst version, and "All to One Random" (SOMAATOR), that is, the best version of SOMA as follow.

(i) *All to one*—this strategy was described in previous section. "All to one" means that all subjects in population migrate to the leader (except leader itself).

(ii) *All to one random* is strategies, in which all individuals move back to one individual (leader), which is not the deepest position on the hyperplane, but it is on the migration of individuals of each randomly selected from the population. Here emerged possible modification of this strategy, such that the individuals do not select randomly, but as appropriate, as is the case of genetic algorithms.

Table 1: SOMA parameter setting.

	A
PathLength	3
Step	0.41
PRT	0.1
PopSize	20
Migrations	50
MinDiv	−1
Individual Length	6
CF Evaluations	6951

4.3. Genetic Algorithm (GA).

Genetic algorithms (GA) imitate the evolutionary processes with emphasis on genotype-based operators (genotype/phenotype dualism). The GA works on a population of artificial chromosomes, referred to as individuals. Each individual is represented by a string of L bits. Each segment of this string corresponds to a variable of the optimizing problem in a binary encoded form.

Table 2: DE parameter setting.

	A
NP	20
F	0.9
CR	0.2
Generations	200
Individual Length	6
CF Evaluations	4000

Table 3: GA parameter setting.

	A
PopSize	20
MutationCostant	0.2
Generations	200
Individual Length	6
CF Evaluations	4000

Table 4: SA parameter setting.

	A
PocetCastic	2
Diameter	0.5
kmax	66
Tmin	0.0001
Tmax	1000
alfa (cooling factor)	0.8

Table 5: Optimized reactor parameters and their range inside which optimization has been done.

Parameter	Range
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0–500
r [m]	0.3–3.0
h [m]	0.5–3.5
T_{VP} [K]	273.15–323.15
$\dot{m}_V$ [kg·s^{-1}]	0–10
d [m]	0.03–0.1

Table 6: The best values of optimized parameters by SOMA, DE.

Parameter	SOMAATO	SOMAATA	DERan1Bin	DERan2Bin
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.0226579	0.042049	0.225063	0.135087
r [m]	0.302427	0.543594	0.195776	2.50725
h [m]	3.12646	1.86628	0.83969	0.715263
T_{VP} [K]	319.286	294.718	296.179	318.115
$\dot{m}_V$ [kg·s^{-1}]	5.58697	9.94168	9.35465	9.431
d [m]	0.0474563	0.0563635	0.030377	0.0583956

Table 7: The best values of optimized parameters by GA, SA.

Parameter	GA	SA_Elitism	SA_NoElitism
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.00417218	0.187008	0.519387
r [m]	2.58293	2.51294	2.84268
h [m]	3.40438	0.858557	2.76458
T_{VP} [K]	310.944	314.657	319.494
$\dot{m}_V$ [kg·s^{-1}]	5.09912	9.02945	6.56782
d [m]	0.0315134	0.0597223	0.0918064

Table 8: Estimated parameters for DERand1Bin.

Parameter	Min.	Avg.	Max.
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.00732026	0.157858	0.467508
r [m]	0.360715	1.80667	2.97025
h [m]	0.506418	1.79237	3.47477
T_{VP} [K]	293.29	306.98	322.669
$\dot{m}_V$ [kg·s^{-1}]	5.32214	9.31932	9.99522
d [m]	0.0300719	0.0391487	0.078392

Table 9: Estimated parameters for DERand2Bin.

Parameter	Min.	Avg.	Max.
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.00985924	0.133546	0.551649
r [m]	0.328569	1.48296	2.99652
h [m]	0.527274	1.96553	3.48392
T_{VP} [K]	293.185	306.886	323.001
$\dot{m}_V$ [kg·s^{-1}]	0.382172	8.31582	9.99879
d [m]	0.0300549	0.0440286	0.0941844

Table 10: Estimated parameters for SOMAATO.

Parameter	Min.	Avg.	Max.
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.0064630	0.0290028	0.126721
r [m]	0.3	0.719769	2.38758
h [m]	0.500022	1.22914	3.29516
T_{VP} [K]	293.52	308.434	322.923
$\dot{m}_V$ [kg·s^{-1}]	2.91889	8.84481	9.99993
d [m]	0.0304519	0.0558279	0.0987916

Table 11: Estimated parameters for SOMAATOR.

Parameter	Min.	Avg.	Max.
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.0159922	0.0927306	0.378352
r [m]	0.321836	1.28136	2.88948
h [m]	0.501199	1.76971	3.47737
T_{VP} [K]	293.444	303.046	322.487
$\dot{m}_V$ [kg·s^{-1}]	1.66669	9.53494	9.99987
d [m]	0.0301466	0.0386274	0.093981

The population is evolved in the optimization process mainly by cross-over operations. This operation recombines the bit strings of individuals in the population with a certain probability Pc. Mutation is secondarily in most applications of a GA. It is responsible to ensure that some bits are changed, thus, allowing the GA to explore the complete search space even if necessary alleles are temporarily lost due to convergence.

Pseudocode 4 describes the general principle of a genetic algorithm.

```
t = 0;
initialize(p(t=0));
evaluatae(P(t=0));
While is NotTerminated() do
$P_c$(t) = reproduction($P_p$);
mutace($P_c$(t));
evaluate($P_c$(t));
P(t+1) = buildNextGenerationForm($P_c$(t),
P(t));
t=t+1;
end
```

Pseudocode 4: Pseudocode of GA.

Table 12: Estimated parameters for GA.

Parameter	Min.	Avg.	Max.
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.0041721	0.362735	1.4908
r [m]	0.308429	2.20758	2.98134
h [m]	0.673105	2.2323	3.48275
T_{VP} [K]	293.253	308.797	323.114
$\dot{m}_V$ [kg·s^{-1}]	0.234471	5.14055	9.96397
d [m]	0.0303933	0.0652728	0.0995752

Table 13: Estimated parameters for SA_Elitism.

Parameter	Min.	Avg.	Max.
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.0234063	0.25735	0.978476
r [m]	0.382136	2.12841	2.9857
h [m]	0.570715	2.25282	3.47805
T_{VP} [K]	293.557	309.318	322.894
$\dot{m}_V$ [kg·s^{-1}]	0.0981996	5.77572	9.99886
d [m]	0.0303382	0.0661099	0.0994636

Table 14: Estimated parameters for SA_NoElitism.

Parameter	Min.	Avg.	Max.
$\dot{m}_{\mathrm{FK}}$ [kg·s^{-1}]	0.0176978	0.315186	1.19071
r [m]	0.506012	2.1492	2.99712
h [m]	0.513602	2.30953	3.49055
T_{VP} [K]	293.282	306.615	322.332
$\dot{m}_V$ [kg·s^{-1}]	0.0015723	5.27791	9.90747
d [m]	0.0300534	0.0630213	0.0994389

4.4. Simulated Annealing (SA). Simulated annealing (SA) is based on the similarity between the solid annealing process and solving combinatorial optimization problems [50]. SA consists of several decreasing temperatures. Each temperature has a few iterations. First, the beginning temperature is selected, and an initial solution is randomly chosen. The value of the cost function based on the current solution (i.e., the initial solution in this case) will then be calculated. The goal is to minimize the cost function. Afterwards, a new solution from the neighborhood of the current solution will be generated. The new value of the cost function based on the new solution will be calculated and compared to the current cost function value. If the new cost function value is less than the current value, it will be accepted. Otherwise, the new value would be accepted only when the Metropolis's criterion [55], which is based on Boltzmann's probability, is met. According to Metropolis's criterion, if the difference between the cost function values of the current and the newly generated solutions (ΔE) is equal to or larger than zero, a random number δ in $[0, 1]$ is generated from a uniform distribution. If (12) is met, the newly generated solution is accepted as the current solution:

$$\delta \leq e\left(-\frac{\Delta E}{T}\right). \qquad (12)$$

The number of new solutions generated at each temperature is the same as the iteration number at the temperature which is constrained by the termination condition. The termination condition could be as simple as a certain number of iterations. After all the iterations at a temperature complete, the temperature would be lowered based on the temperature-updating rule. At the updated (and lowered) temperature, all required iterations will have to be completed before moving to the next temperature. This process would repeat until the halting criterion is met. The halting criterion could be "reaching the pre-set minimum temperature." The result of simulated annealing (SA) is related to the number of iterations at each temperature and the speed of reducing temperature. The temperature updating rule proposed in this paper is shown:

$$\text{Temperature} = Te^{(-rt)}, \qquad (13)$$

where T is the initial temperature, r the cooling ratio, and t the number of times the temperature has been lowered. The cooling ratio controls the speed of cooling. The higher the cooling ratio, the faster the temperature cools down.

In this work I have chosen two versions of SA algorithms: SA elitism (SA_Elitism) and SA without elitism (SA_NoElitism) for investigation on optimization of a chemical reactor.

4.5. Usage of Elitism. It uses synchronization at the end of temperature phase; otherwise, the communication proceeds asynchronous after each iteration.

(i) *Disadvantage*: of this approach lies in excessive communication, which results in computation time increase.

(ii) *Advantage*: elitism removes problem with the acceptance of worse solutions at low temperature phase.

5. Static Optimization of Reactor

The above-described reactor, in the original setup, gives unsatisfactory results. To improve reactor behavior, static optimization was performed using the algorithms SOMA, DE, GA, SA. In this work the optimization was performed by the following optimization of batching-value reactor's parameter geometry.

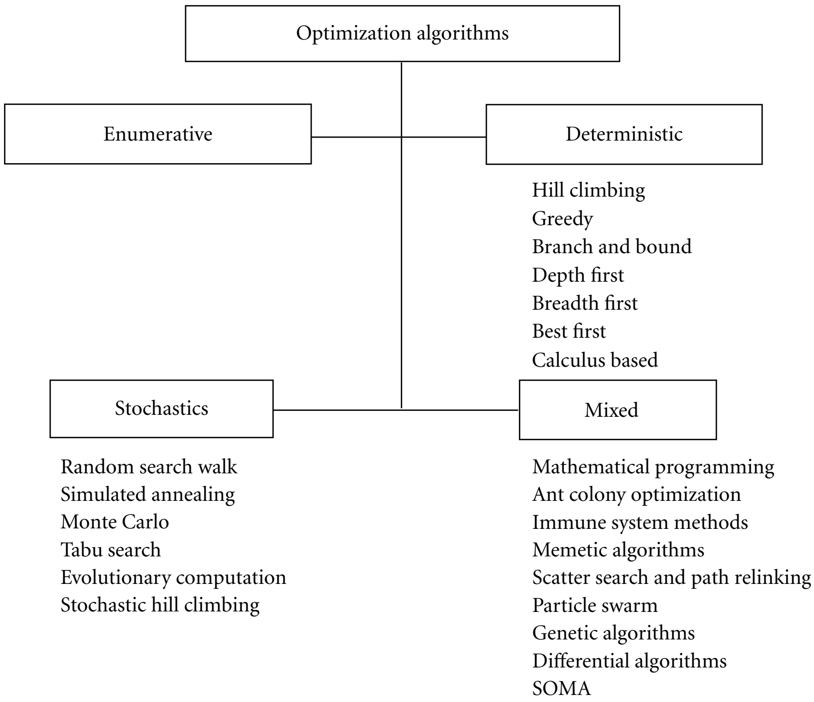


Figure 2: Division of evolutionary algorithms, taken from [2].

5.1. The Cost Function (CF). In this optimization the point was to minimize the area arising as a difference between the required and real temperature profile of the reaction mixture in a selected time interval, which was the duration of a batch cycle. The required temperature was 97°C (370.15 K). The cost function that was minimized is given in:

$$f_{\cos t} = \sum_{t=0}^{t} |w - T[t]|, \tag{14}$$

where: w: control point, T: temperature.

The CF has been calculated in general from the distance between desired state and actual system output.

5.2. Parameter Settings. The parameter settings have been found empirically and are given in Table 1 (SOMA) and Table 2 (DE). In Tables 3 and 4 are parameters' setting for GA and SA. The main criterion for this setting was to keep the same setting of parameters as much as possible and of course the same number of cost function evaluations as well as population size (parameter PopSize for SOMA, GA, and NP for DE). Number of optimized reactor parameters and their range inside represents in Table 5.

6. Experimental Results

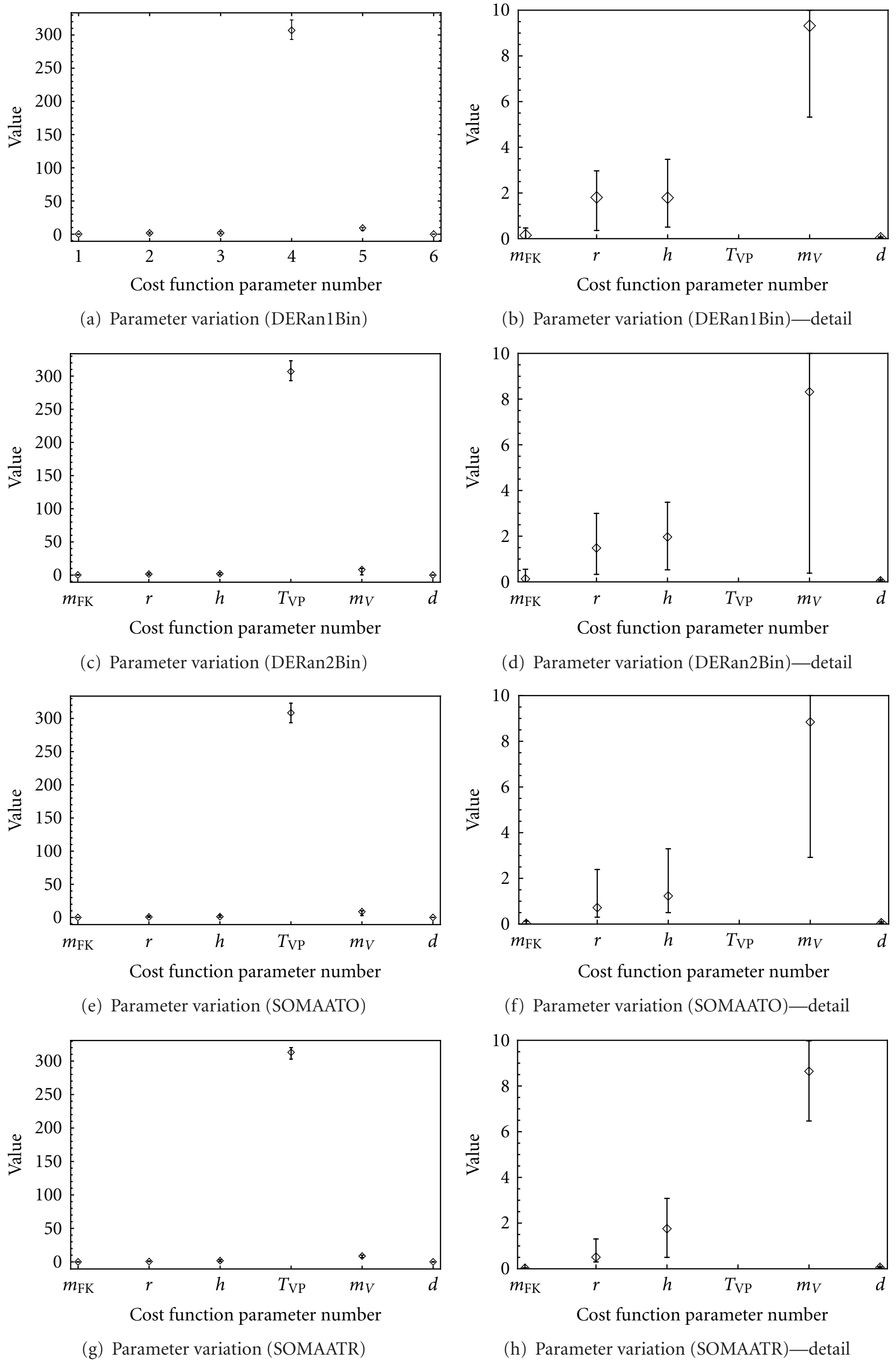
Due to the fact that EAs are partly of stochastic nature, a large set of simulations has to be done in order to get data for statistical data processing. Four algorithms (SOMA, DE, GA, and SA) have been applied 100 times in order to find the optimum of process parameters and the reactor geometry. All important data has been visualized directly and/or processed for graphs demonstrating performance of four algorithms. Estimated parameters and their diversity (minimum, maximum, and average) are depicted in Figures 3 and 4. From those pictures it is visible that results from four algorithms are comparable. For the demonstration are graphically the best solutions shown in the subfigures (b), (d), (f), and (h) of Figures 5–11. There is shown time dependence of process parameters from four algorithms. The best values of parameter setting are recorded in Tables 6 and 7. All one hundred simulations diversity (minimum, maximum and average) were described from Tables 8–14 for each version of four algorithms. On Figures 5–11 are, for example, shown records of all 100 simulations and the best solutions of all 100 simulations (Figures 5 and 6 for SOMA, Figures 6 and 7 for DE, Figure 8 for GA, and Figures 9 and 10 for SA).

6.1. Parameter Diversity for Repeated 100 Times Simulations. See Tables 8–14.

6.2. Graphics Results. See Figures 3–11.

7. Discussion to the Result Optimization

This work has presented a systematic procedure to derive a solution model for operation of a dynamic chemical reactor process. The results produced by the optimizations depend not only on the problem being solved but also on the way of how to define a given function. All simulations were repeated 100 times for each EA with the same initial conditions for each simulation.



(a) Parameter variation (DERan1Bin)

(b) Parameter variation (DERan1Bin)—detail

(c) Parameter variation (DERan2Bin)

(d) Parameter variation (DERan2Bin)—detail

(e) Parameter variation (SOMAATO)

(f) Parameter variation (SOMAATO)—detail

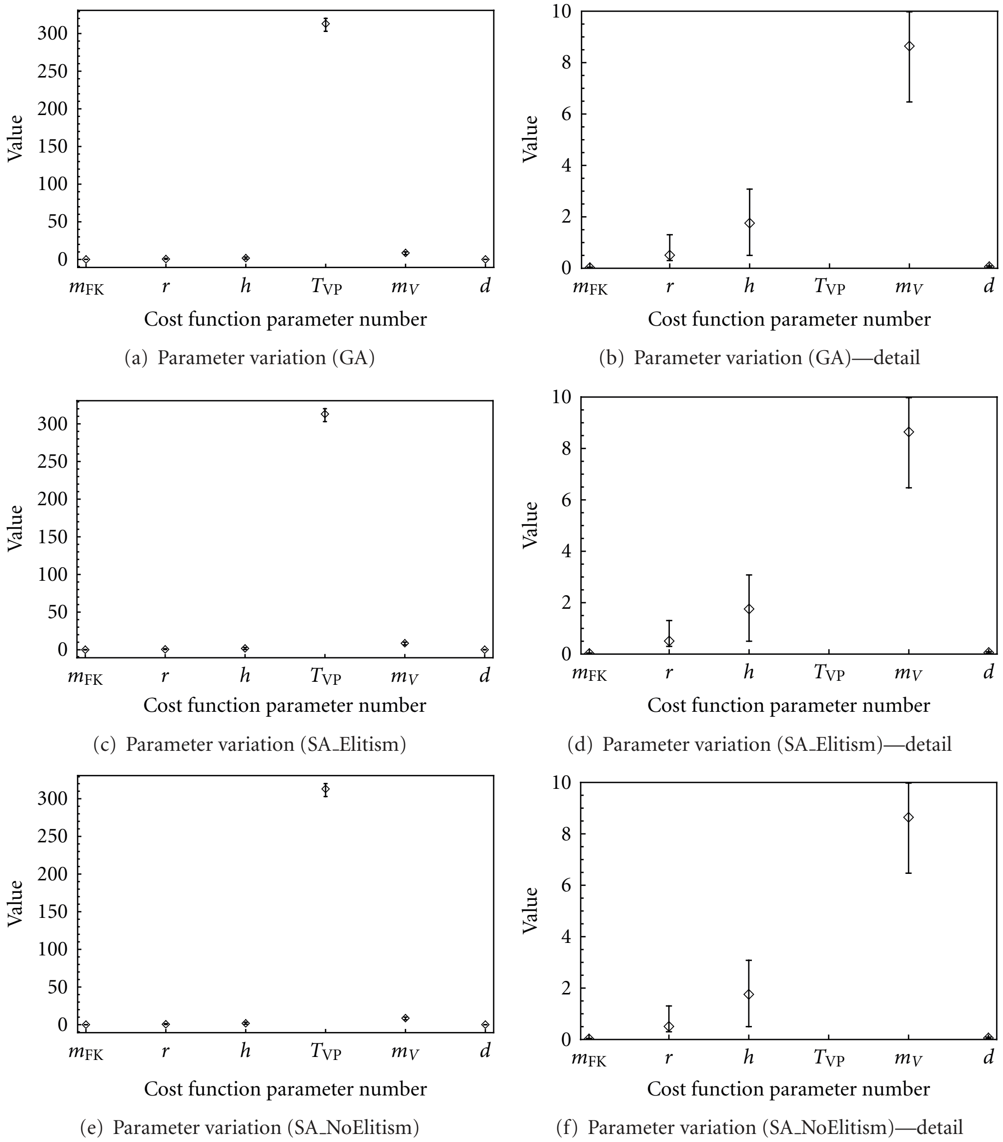
(g) Parameter variation (SOMAATR)

(h) Parameter variation (SOMAATR)—detail

Figure 3: Parameter variation of SOMA and DE.

The differences between four methods SOMA, DE, GA, and SA are best seen in Tables 5, 6, and 7. The first part shows the parameters of batch reactor designed by an expert, and the second part shows the parameters obtained through static optimization.

Calculation was 100 times repeated and the best, worst, and average results (individual) were recorded from the last population in each simulation. All one hundred triplets (best, worst, average) were used to create Figures 3 and 4.



(a) Parameter variation (GA)

(b) Parameter variation (GA)—detail

(c) Parameter variation (SA_Elitism)

(d) Parameter variation (SA_Elitism)—detail

(e) Parameter variation (SA_NoElitism)

(f) Parameter variation (SA_NoElitism)—detail

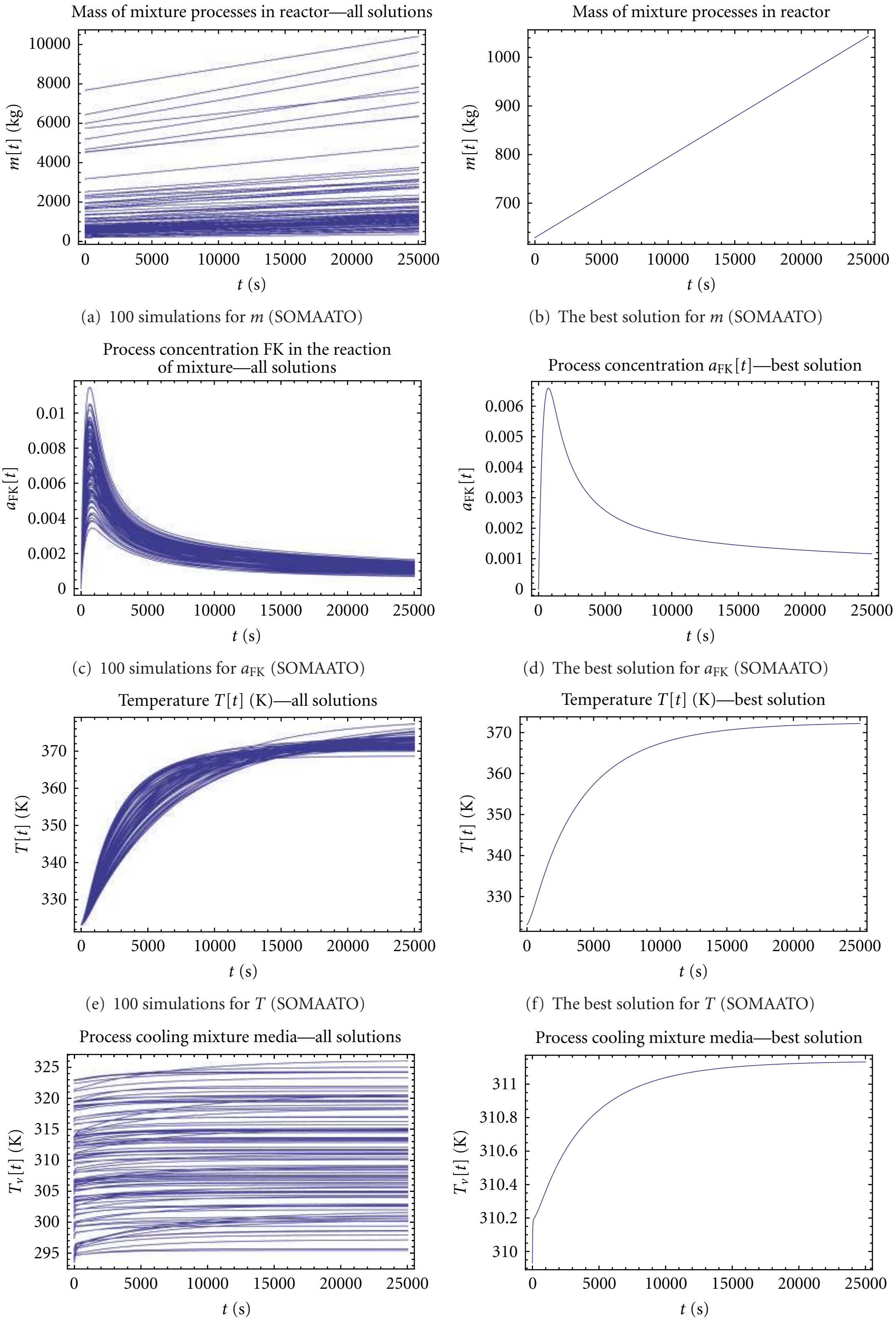
Figure 4: Parameter variation of GA and SA.

Four algorithms (SOMA, DE, GA, SA) have been applied 100 times in order to find the optimum of process parameters and the reactor geometry. *The primary aim of this comparative study is not to show which algorithm is better or worse,* but to show from the outputs of all simulations depicted in Figures 5–11 that evolution SOMA represents the best solution from actual simulation more than DE, GA, and SA. Based on data from all simulations, four comparisons can be done. From parameter variation of view, the estimated parameters depicted in Figures 3 and 4 show that four algorithms are comparable in performance (with small deviations).

From the graphs, it is evident that the courses of SOMA algorithm are more densities in thin spectra and not far from the start of mass axis (see Figure 5(a)). Only few values drift out of the spectra. The results of the DE algorithm are faster in the weight spectra. From these results we may conclude that SOMA has much better convergence than DE, GA, SA algorithms (see Figures 5–11). For better overview of comparison between four algorithms, I have chosen process temperature of reaction mixture T, shown in Figure 12.

In Figure 12 we can see the process parameter temperature T simulation by SOMA were of more stability than other algorithms (concretely, in this experimental problem of batch reactor).

From the obtained results, it is possible to say that all simulations give satisfactory results, and; thus, *evolutionary algorithms are capable of solving this class of difficult problems and the quality of results does not depend only on the problem being solved, but they are extremely sensitive for the proper definition of the cost function, selection of parameter setting of evolutionary algorithms.*


Mass of mixture processes in reactor—all solutions

(a) 100 simulations for m (SOMAATO)

Mass of mixture processes in reactor

(b) The best solution for m (SOMAATO)

Process concentration FK in the reaction of mixture—all solutions

(c) 100 simulations for a_{FK} (SOMAATO)

Process concentration $a_{\mathrm{FK}}[t]$—best solution

(d) The best solution for a_{FK} (SOMAATO)

Temperature $T[t]$ (K)—all solutions

(e) 100 simulations for T (SOMAATO)

Temperature $T[t]$ (K)—best solution

(f) The best solution for T (SOMAATO)

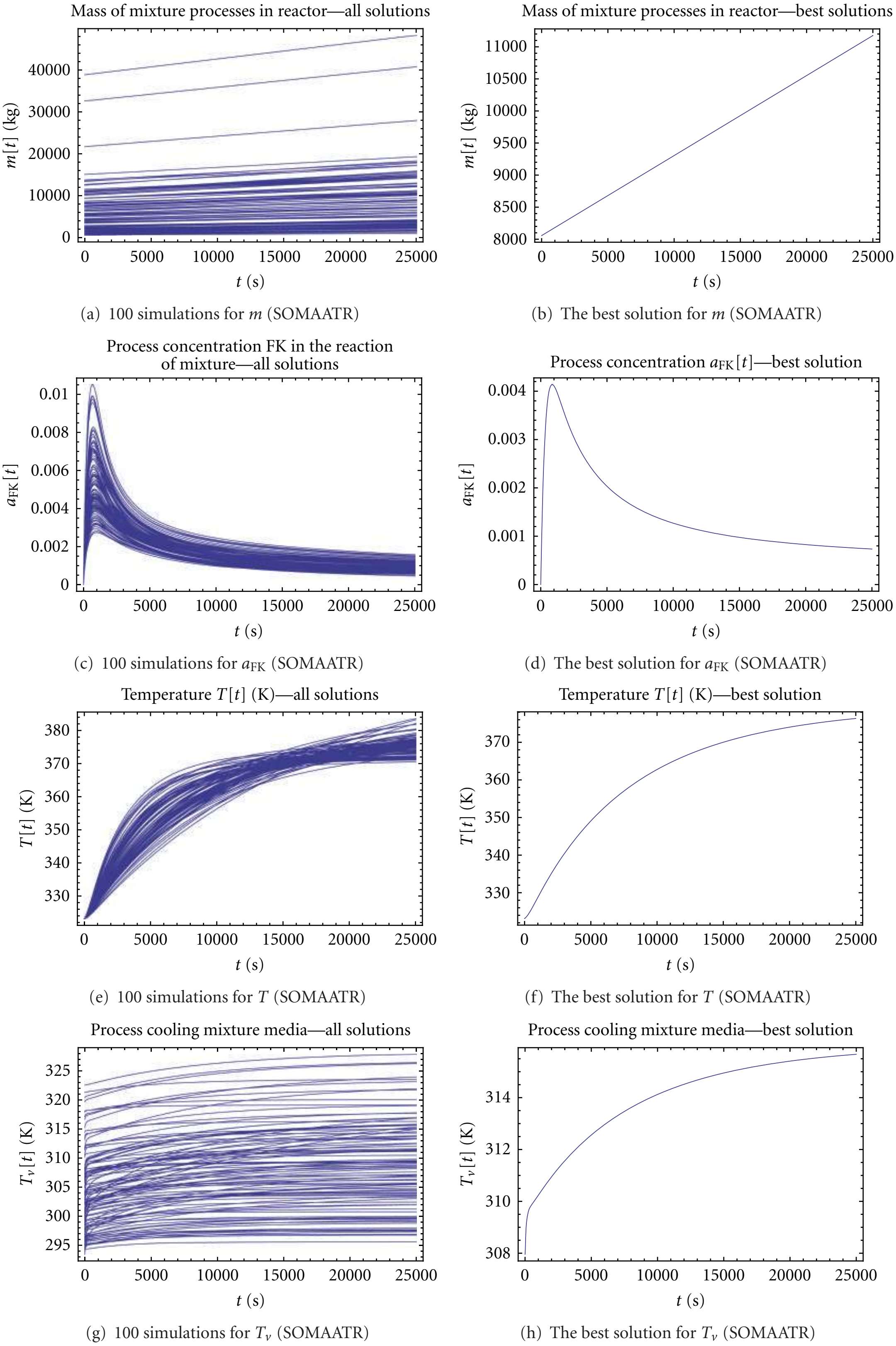
Process cooling mixture media—all solutions

(g) 100 simulations for T_v (SOMAATO)

Process cooling mixture media—best solution

(h) The best solution for T_v (SOMAATO)

Figure 5: Processes parameters for 100 simulations of SOMA—version SOMAATO.



Mass of mixture processes in reactor—all solutions

(a) 100 simulations for m (SOMAATR)

Mass of mixture processes in reactor—best solutions

(b) The best solution for m (SOMAATR)

Process concentration FK in the reaction of mixture—all solutions

(c) 100 simulations for a_{FK} (SOMAATR)

Process concentration $a_{\mathrm{FK}}[t]$—best solution

(d) The best solution for a_{FK} (SOMAATR)

Temperature $T[t]$ (K)—all solutions

(e) 100 simulations for T (SOMAATR)

Temperature $T[t]$ (K)—best solution

(f) The best solution for T (SOMAATR)

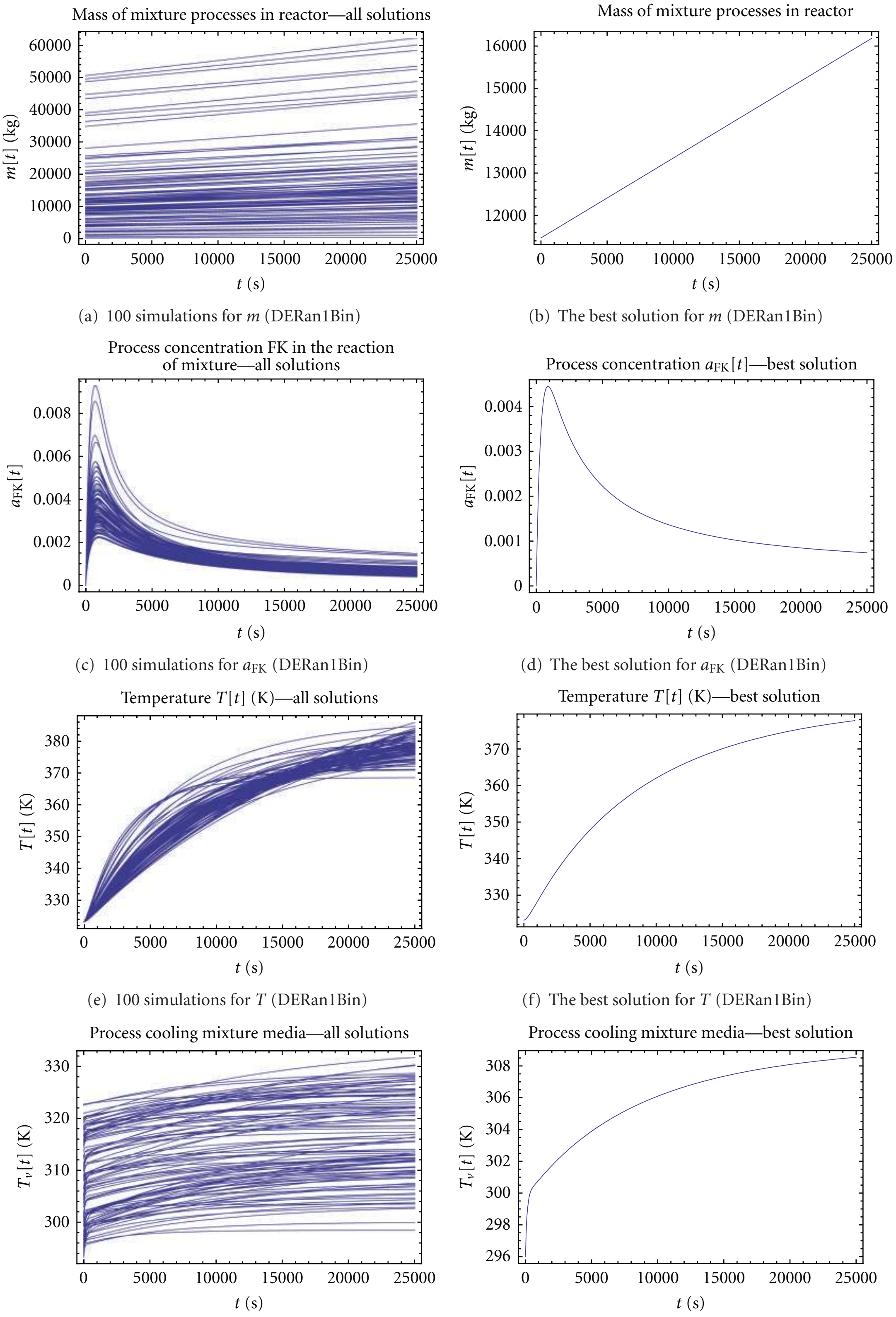
Process cooling mixture media—all solutions

(g) 100 simulations for T_v (SOMAATR)

Process cooling mixture media—best solution

(h) The best solution for T_v (SOMAATR)

Figure 6: Processes parameters for 100 simulations of SOMA—version SOMAATR.



Mass of mixture processes in reactor—all solutions

(a) 100 simulations for m (DERan1Bin)

Mass of mixture processes in reactor

(b) The best solution for m (DERan1Bin)

Process concentration FK in the reaction of mixture—all solutions

(c) 100 simulations for a_{FK} (DERan1Bin)

Process concentration $a_{\mathrm{FK}}[t]$—best solution

(d) The best solution for a_{FK} (DERan1Bin)

Temperature $T[t]$ (K)—all solutions

(e) 100 simulations for T (DERan1Bin)

Temperature $T[t]$ (K)—best solution

(f) The best solution for T (DERan1Bin)

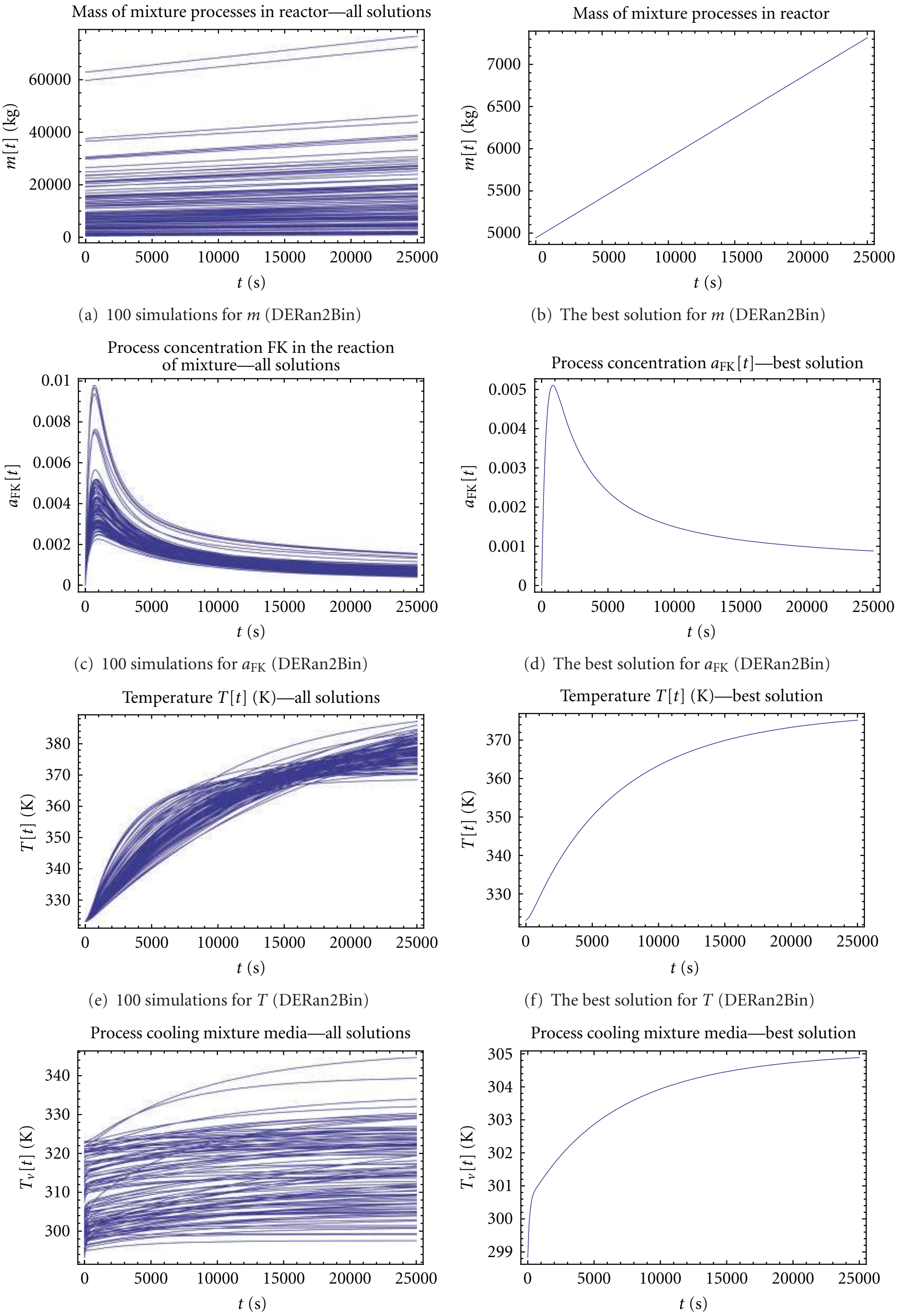
Process cooling mixture media—all solutions

(g) 100 simulations for T_v (DERan1Bin)

Process cooling mixture media—best solution

(h) The best solution for T_v (DERan1Bin)

Figure 7: Processes parameters for 100 simulations of DE—version DERan1Bin.



(a) 100 simulations for m (DERan2Bin)

(b) The best solution for m (DERan2Bin)

(c) 100 simulations for a_{FK} (DERan2Bin)

(d) The best solution for a_{FK} (DERan2Bin)

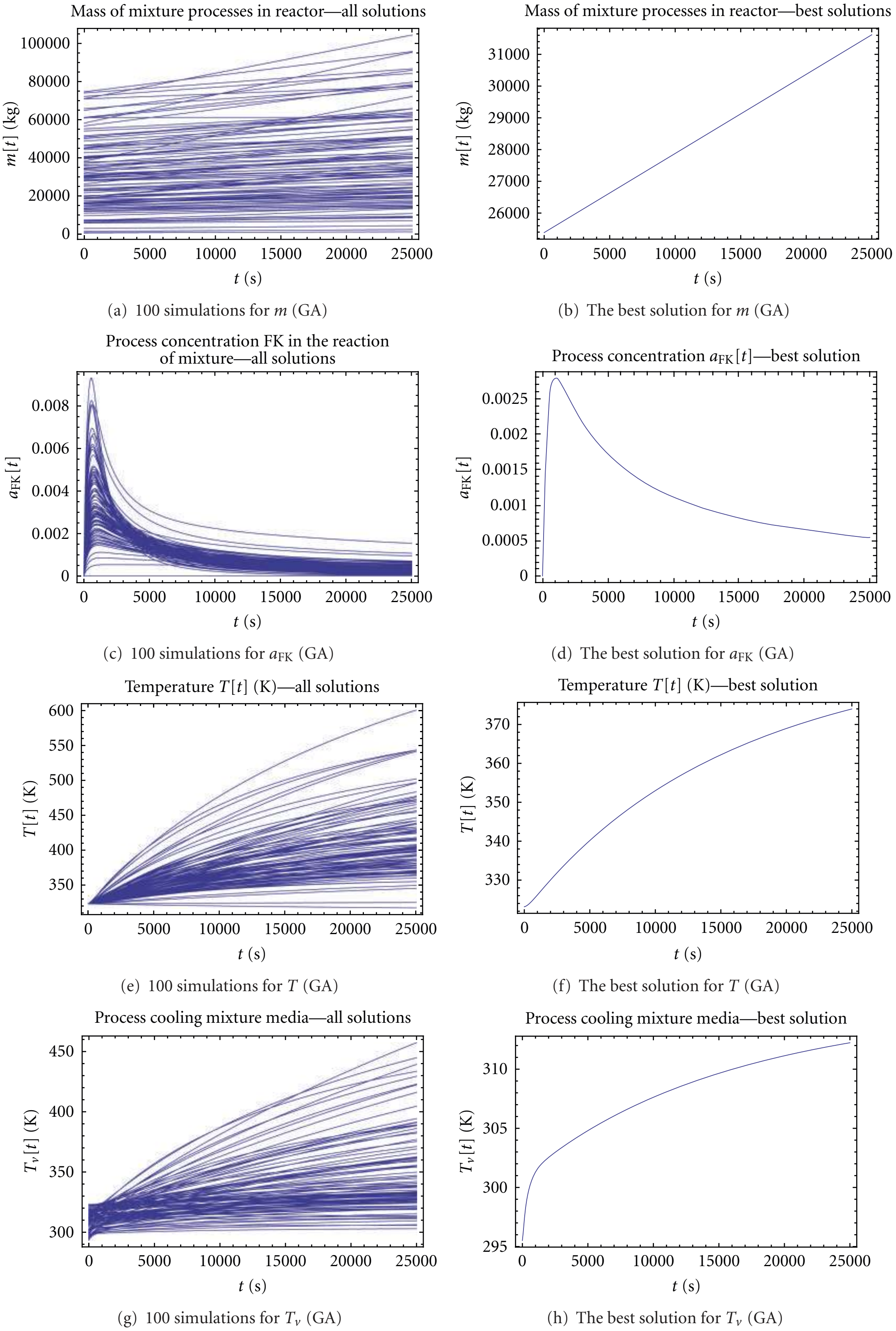
(e) 100 simulations for T (DERan2Bin)

(f) The best solution for T (DERan2Bin)

(g) 100 simulations for T_v (DERan2Bin)

(h) The best solution for T_v (DERan2Bin)

FIGURE 8: Processes parameters for 100 simulations of DE—version DERan2Bin.



(a) 100 simulations for m (GA)

(b) The best solution for m (GA)

(c) 100 simulations for a_{FK} (GA)

(d) The best solution for a_{FK} (GA)

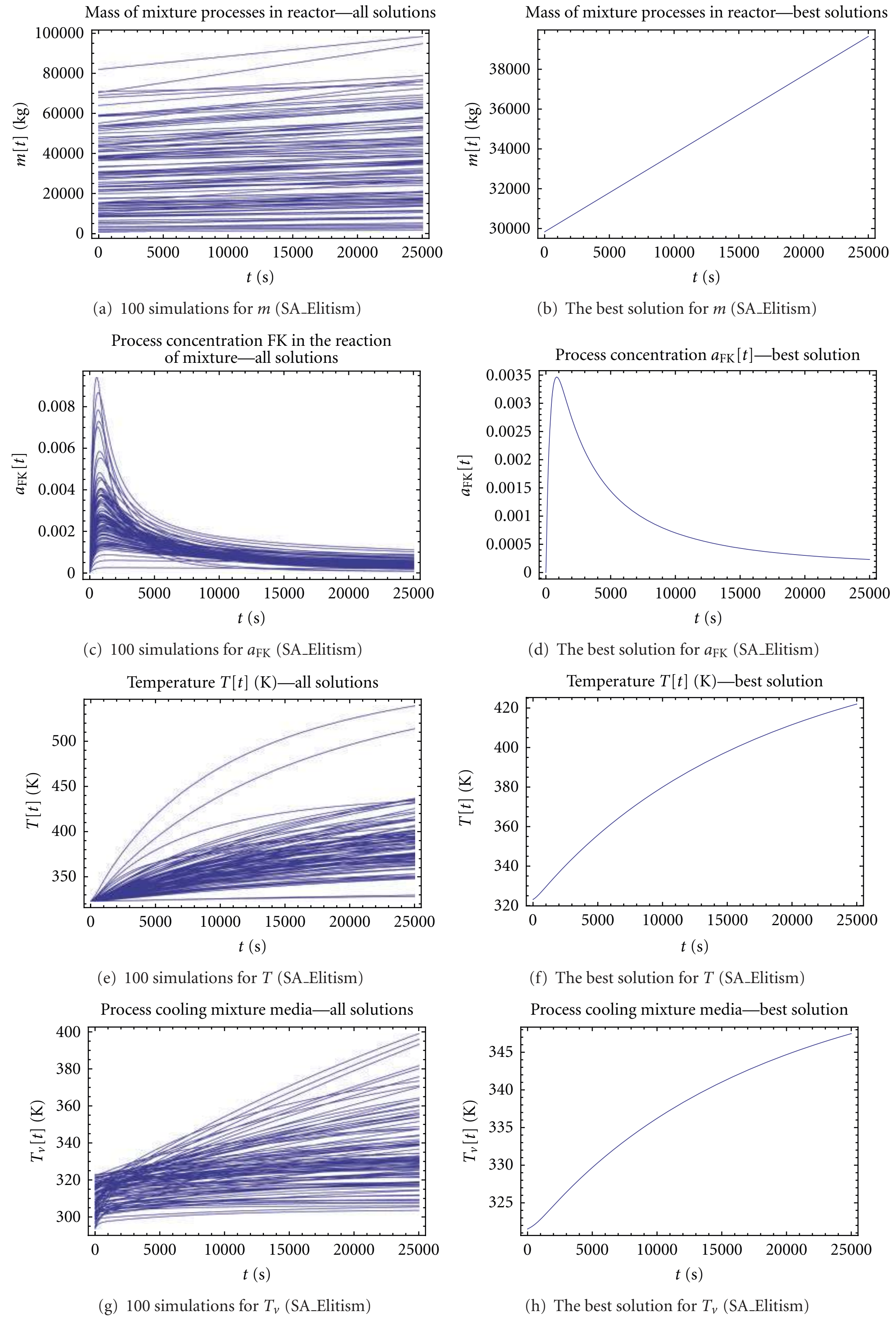
(e) 100 simulations for T (GA)

(f) The best solution for T (GA)

(g) 100 simulations for T_v (GA)

(h) The best solution for T_v (GA)

Figure 9: Processes parameters for 100 simulations of GA.



(a) 100 simulations for m (SA_Elitism)

(b) The best solution for m (SA_Elitism)

(c) 100 simulations for a_{FK} (SA_Elitism)

(d) The best solution for a_{FK} (SA_Elitism)

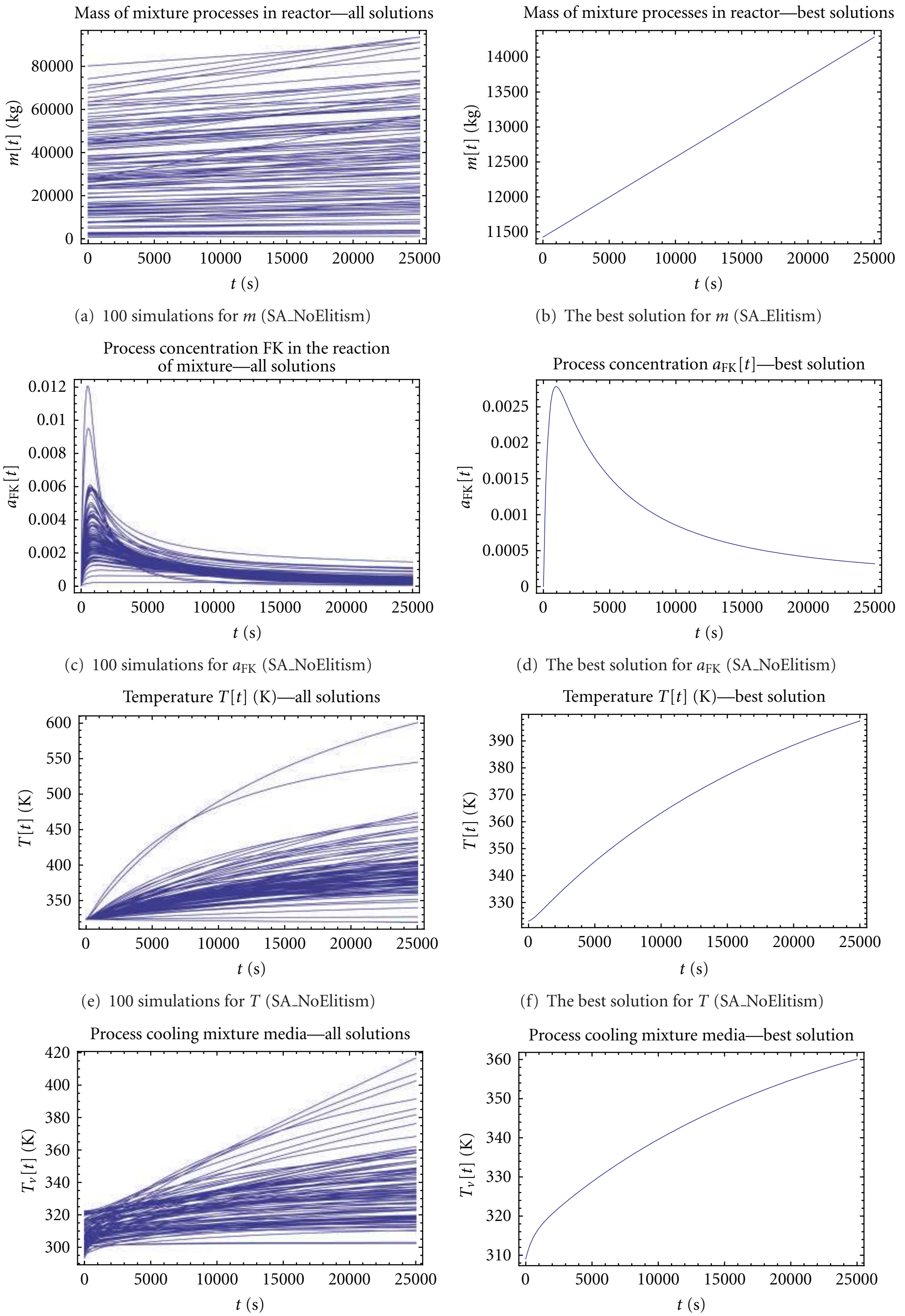
(e) 100 simulations for T (SA_Elitism)

(f) The best solution for T (SA_Elitism)

(g) 100 simulations for T_v (SA_Elitism)

(h) The best solution for T_v (SA_Elitism)

Figure 10: Processes parameters for 100 simulations of SA—version SA_Elitism.



(a) 100 simulations for m (SA_NoElitism)

(b) The best solution for m (SA_Elitism)

(c) 100 simulations for a_{FK} (SA_NoElitism)

(d) The best solution for a_{FK} (SA_NoElitism)

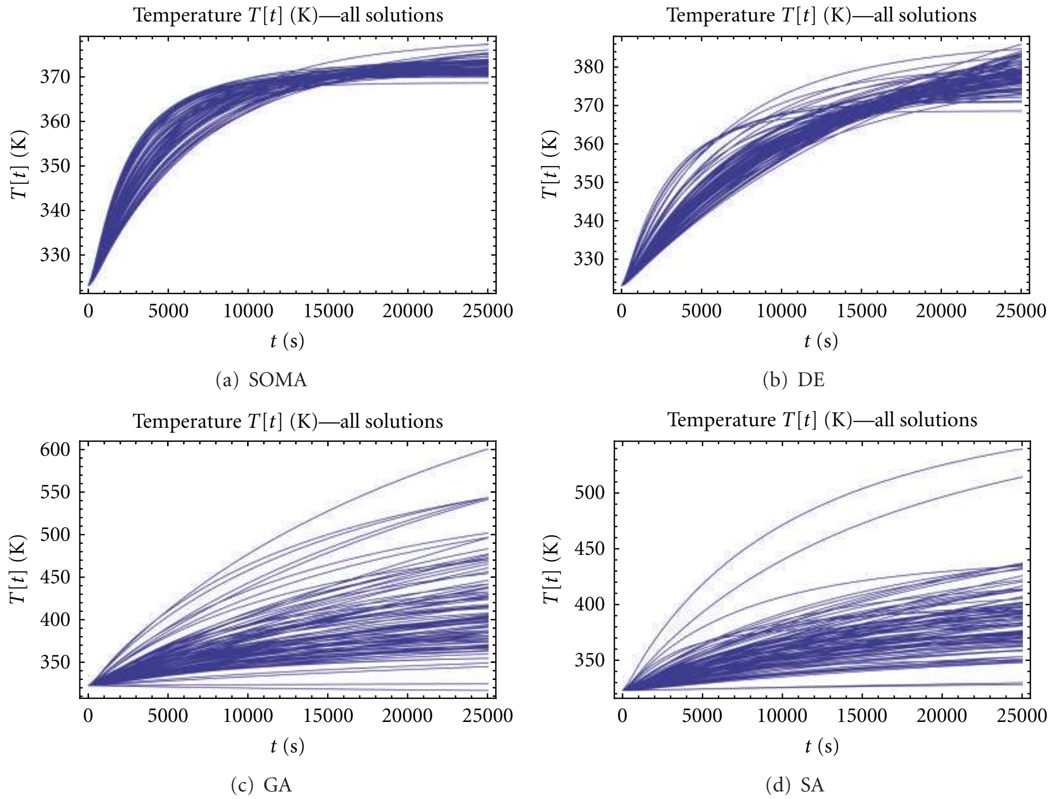
(e) 100 simulations for T (SA_NoElitism)

(f) The best solution for T (SA_NoElitism)

(g) 100 simulations for T_v (SA_NoElitism)

(h) The best solution for T_v (SA_NoElitism)

Figure 11: Processes parameters for 100 simulations of SA—version SA_NoElitism.



(a) SOMA

(b) DE

(c) GA

(d) SA

FIGURE 12: 100 simulations for T.

8. General Conclusion and Further Research Perspective

In this paper, evolutionary algorithms were used for static optimization of chemical reactor in order to improve the quality of this behaviour in the uncontrolled state. The optimization tool has been described and selected four EAs (SOMA, DE, SA, GA), especially for a certified high robustness and ability to successfully solve complex optimization problems, especially of chemical reactor.

In fact, using the methods of article intelligence, mainly, the evolutionary computation techniques, can be used in such a difficult task which is analyzed and optimized of a nonlinear system, especially of given chemical reactor. The main aim of paper was focused on the examples of EAs implementation to methods for chemical reaction for the purpose of obtaining better results, which means faster reaching of desired stable state and superior stabilization and could be robust and effective to optimize difficult problems in the fields of chemical engineering.

Basic optimization process presented here was based on a relatively simple function. Unless the experiment is limited by technical issues when searching for optimal parameters, there is no problem in defining more complex functional including as subcriteria, for example, stability, costs, time-optimal criteria, controllability, or their arbitrary combinations.

Future research of evolutionary algorithms SOMA, DE, GA, SA is still open. According to all results obtained during this research, it is planned that the main activities would be focused on the expanding of this study for other chemical dynamic systems.

From the results of paper, we can conclude that EAs have shown great potential and ability to solve complex problems of optimization, not only at the field of chemical engineering process but also in diverse industrial fields.

Acknowledgments

This work is part of the science activities at the Ton Duc Thang University, Ho Chi Minh City, Vietnam. The author would like to thank his professor I. Zelinka from the Department of Informatics and Artificial Intelligence, the Tomas Bata university in Zlín, Czech Republic, who inspired his writing skill and supported him with a very professional way in research.

References

[1] Z. Michalewicz and M. Michalewicz, "Evolutionary computation techniques and their applications," in *Proceedings of the IEEE International Conference on Intelligent Processing Systems, (ICIPS '97)*, pp. 14–25, October 1997.

[2] I. Zelinka, *Umělá Inteligence v Problémech Globální Optimalizace*, BEN, Praha, 2002.

[3] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*, Natural Computing Series, Springer, New York, NY, USA, 1st edition, 2003.

[4] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*, Natural Computing Series, Springer, New York, NY, USA, 2nd edition, 2007.

[5] Z. Oplatkova, *Metaevolution—synthesis of evolutionary algorithms by means of symbolic regression*, Ph.D. thesis, Faculty of Applied Informatics, Tomas Bata University in Zlin, 2009.

[6] E. C. Marcoulaki and A. C. Kokossis, "Scoping and screening complex reaction networks using stochastic optimization," *AIChE Journal*, vol. 45, no. 9, pp. 1977–1991, 1999.

[7] R. Aris, "Studies in optimization-III. The optimum operating conditions in sequences of stirred tank reactors," *Chemical Engineering Science*, vol. 13, no. 2, pp. 75–81, 1960.

[8] F. Horn, "Attainable and non-attainable regions in chemical reaction technique," in *Proceedings of the European Symposium on Chemical Reaction Engineering*, Pergamon Press, London, UK, 1964.

[9] D. C. Dyson and J. M. Horn, "Optimum distributed feed reactors for exothermic reversible reactions," *Journal of Optimization Theory and Applications*, vol. 1, no. 1, pp. 40–52, 1967.

[10] D. C. Dyson and J. M. Horn, "Optimum adiabatic cascade reactor with direct intercooling," *Industrial and Engineering Chemistry Fundamentals*, vol. 8, no. 1, pp. 49–53, 1969.

[11] D. Y. C. Ng and D. W. T. Rippin, "The effect of incomplete mixing on conversion in homogeneous reactions," in *Proceedings of the European Symposium on Chemical Reaction Engineering*, Pergamon Press, Oxford, UK, 1965.

[12] R. Jackson, "Optimization of chemical reactors with respect to flow configuration," *Journal of Optimization Theory and Applications*, vol. 2, no. 4, pp. 240–259, 1968.

[13] A. L. Ravimohan, "Optimization of chemical reactor networks with respect to flow configuration," *Journal of Optimization Theory and Applications*, vol. 8, no. 3, pp. 204–211, 1971.

[14] J. D. Paynter and D. E. Haskins, "Determination of optimal reactor type," *Chemical Engineering Science*, vol. 25, no. 9, pp. 1415–1422, 1970.

[15] S. P. Chitra and R. Govind, "Yield optimization for complex reactor systems," *Chemical Engineering Science*, vol. 36, no. 7, pp. 1219–1225, 1981.

[16] S. P. Chitra and R. Govind, "Synthesis of optimal serial reactor structures for homogeneous reactions. Part I: isothermal reactors," *AIChE Journal*, vol. 31, no. 2, pp. 177–184, 1985.

[17] S. P. Chitra and R. Govind, "Synthesis of optimal serial reactor structures for homogeneous reactions. Part II: nonisothermal reactors," *AIChE Journal*, vol. 31, no. 2, pp. 185–194, 1985.

[18] L. E. K. Achenie and L. T. Biegler, "Algorithmic synthesis of chemical reactor networks using mathematical programming," *Industrial and Engineering Chemistry Fundamentals*, vol. 25, no. 4, pp. 621–627, 1986.

[19] L. E. K. Achenie and L. T. Biegler, "Developing targets for the performance index of a chemical reactor network: isothermal systems," *Industrial & Engineering Chemistry Research*, vol. 27, no. 10, pp. 1811–1821, 1988.

[20] L. E. K. Achenie and L. T. Biegler, "A superstructure based approach to chemical reactor network synthesis," *Computers and Chemical Engineering*, vol. 14, no. 1, pp. 23–40, 1990.

[21] A. C. Kokossis and C. A. Floudas, "Optimization of complex reactor networks: I. Isothermal operation," *Chemical Engineering Science*, vol. 45, no. 3, pp. 595–614, 1990.

[22] A. C. Kokossis and C. A. Floudas, "Synthesis of isothermal reactor-separator-recycle systems," *Chemical Engineering Science*, vol. 46, no. 5-6, pp. 1361–1383, 1991.

[23] A. C. Kokossis and C. A. Floudas, "Optimization of complex reactor networks: II. Nonisothermal operation," *Chemical Engineering Science*, vol. 49, no. 7, pp. 1037–1051, 1994.

[24] D. Hildebrandt, D. Glasser, and C. M. Crowe, "Geometry of the attainable region generated by reaction and mixing: with and without constraints," *Industrial & Engineering Chemistry Research*, vol. 29, no. 1, pp. 49–58, 1990.

[25] D. Hildebrandt and D. Glasser, "The attainable region and optimal reactor structures," *Chemical Engineering Science*, vol. 45, no. 8, pp. 2161–2168, 1990.

[26] B. Glasser, D. Hildebrandt, and D. Glasser, "Optimal mixing for exothermic reversible reactions," *Industrial & Engineering Chemistry Research*, vol. 31, no. 6, pp. 1541–1549, 1992.

[27] D. Glasser, D. Hildebrandt, and S. Godorr, "The attainable region for segregated, maximum mixed, and other reactor models," *Industrial & Engineering Chemistry Research*, vol. 33, no. 5, pp. 1136–1144, 1994.

[28] D. Glasser and D. Hildebrandt, "Reactor and process synthesis," *Computers and Chemical Engineering*, vol. 21, no. 1, pp. S775–S783, 1997.

[29] F. Hopley, D. Glasser, and D. Hildebrandt, "Optimal reactor structures for exothermic reversible reactions with complex kinetics," *Chemical Engineering Science*, vol. 51, no. 10, pp. 2399–2407, 1996.

[30] A. Nisoli, M. F. Malone, and M. F. Doherty, "Attainable Regions for Reaction with Separation," *AIChE Journal*, vol. 43, no. 2, pp. 374–386, 1997.

[31] S. Balakrishna and L. T. Biegler, "A constructive targeting approaches for the synthesis of chemical reactor networks," *Industrial & Engineering Chemistry Research*, vol. 31, no. 1, pp. 300–312, 1992.

[32] S. Balakrishna and L. T. Biegler, "Targeting strategies for the synthesis and energy integration of nonisothermal reactor networks," *Industrial and Engineering Chemistry Research*, vol. 31, no. 9, pp. 2152–2164, 1992.

[33] A. Lakshmanan and L. T. Biegler, "Reactor network targeting for waste minimization," in *Pollution Prevention via Process and Product Modifications*, M. El-Halwagi and D. Petrides, Eds., vol. 90 of *AIChE Symposium Series*, p. 128, AIChE, New York, NY, USA, 1994.

[34] A. Lakshmanan and L. T. Biegler, "Synthesis of optimal chemical reactor networks," *Industrial and Engineering Chemistry Research*, vol. 35, no. 4, pp. 1344–1353, 1996.

[35] A. Lakshmanan and L. T. Biegler, "A case study for reactor network synthesis: the vinyl chloride process," *Computers and Chemical Engineering*, vol. 21, no. 1, pp. S785–S790, 1997.

[36] A. Rojnuckarin, C. A. Floudas, H. Rabitz, and R. A. Yetter, "Methane conversion to ethylene and acetylene: optimal control with chlorine, oxygen, and heat flux," *Industrial and Engineering Chemistry Research*, vol. 35, no. 3, pp. 683–696, 1996.

[37] C. A. Schweiger and C. A. Floudas, "Optimization framework for the synthesis of chemical reactor networks," *Industrial and Engineering Chemistry Research*, vol. 38, no. 3, pp. 744–766, 1999.

[38] D. Hildebrandt and L. T. Biegler, *Synthesis of Chemical Reactor Networks*, FOCAPD, Snowmass, Colo, USA, 1994.

[39] D. Dasgupta and Z. Michalewicz, *Evolutionary Algorithms in Engineering Applications*, Springer, Berlin, Germany, 1997.

[40] L. Nolle, A. Goodyear, A. A. Hopgood, P. D. Picton, and N. S. T. J. Braithwaite, "On step width adaptation in simulated annealing for continuous parameter optimisation," in *Computational Intelligence—Theory and Applications*, B. Reusch, Ed.,

vol. 2206 of *Lecture Notes in Computer Science*, pp. 589–598, Springer, Berlin, Germany, 2001.

[41] L. Nolle, I. Zelinka, A. A. Hopgood, and A. Goodyear, "Comparison of an self-organizing migration algorithm with simulated annealing and differential evolution for automated waveform tuning," *Advances in Engineering Software*, vol. 36, no. 10, pp. 645–653, 2005.

[42] I. Zelinka and L. Nolle, "Plasma reactor optimizing using differential evolution," in *Differential Evolution: A Practical Approach to Global Optimization*, K. V. Price, J. Lampinen, and R. Storn, Eds., pp. 499–512, Springer, New York, NY, USA, 2006.

[43] T. D. Tran and I. Zelinka, "Optimization of reactive distillation processes using self-organizing migrating algorithm and differential evolution strategies," in *Proceedings of the 14th International Conference on Soft Computing (MENDEL '08)*, pp. 186–191, Brno, Czech Republic, June 2008.

[44] T. D. Tran and I. Zelinka, "Using method of artificial intelligence to optimise and control chemical reactor," in *Proceedings of the 15th International Conference on Soft Computing (MENDEL '09)*, Brno, Czech Republic, June 2009.

[45] O. J. Emuoyibofarhe and R. A. Sunday, "An optimum solution for a process control problem (continuous stirred tank reactor) using a hybrid neural network," *Journal of Theoretical and Applied Information Technology*, vol. 4, no. 10, pp. 906–915, 2008.

[46] K. Price, "An introduction to differential evolution," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds., pp. 79–108, McGraw-Hill, London, UK, 1999.

[47] I. Zelinka, "SOMA—self organizing migrating algorithm," in *New Optimization Techniques in Engineering*, B. V. Babu and G. Onwubolu, Eds., pp. 167–218, Springer, New York, NY, USA, 2004.

[48] I. Zelinka, "SOMA," in *New Optimization Techniques in Engineering*, B. V. Babu and G. Onwubolu, Eds., Springer, New York, NY, USA, 2004.

[49] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, no. 1, pp. 44–50, 1992.

[50] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[51] V. Černý, "Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm," *Journal of Optimization Theory and Applications*, vol. 45, no. 1, pp. 41–51, 1985.

[52] L. Liu, W. Liu, and D. A. Cartes, "Particle swarm optimization-based parameter identification applied to permanent magnet synchronous motors," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 7, pp. 1092–1100, 2008.

[53] F. Glover, M. Laguna, and R. Martí, "Scatter search," in *Advances in Evolutionary Computation: Theory and Applications*, A. Ghosh and S. Tsutsui, Eds., pp. 519–537, Springer, New York, NY, USA, 2003.

[54] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[55] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

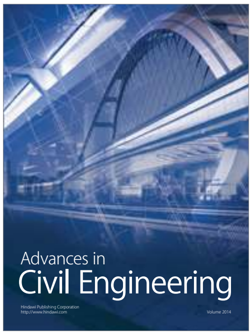
Journal of
Engineering

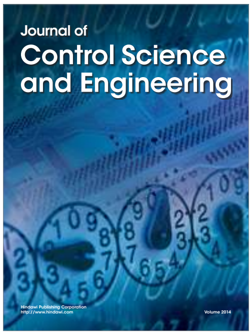
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

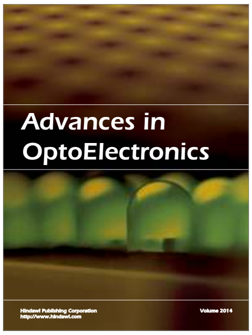
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

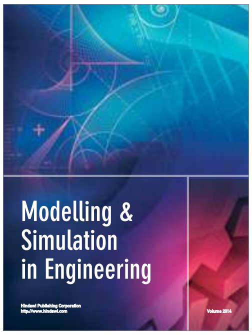
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

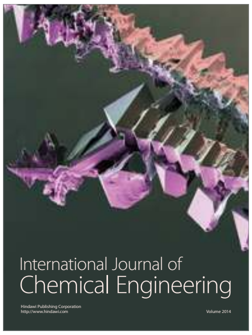

Advances in
OptoElectronics

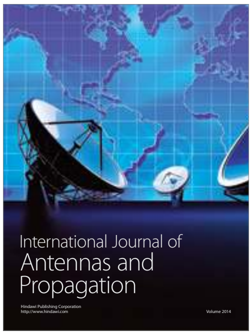
VLSI Design

International Journal of
Navigation and
Observation

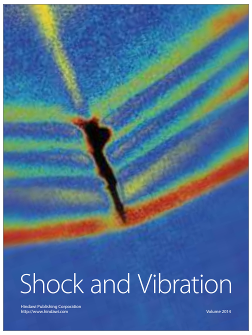

Modelling &
Simulation
in Engineering

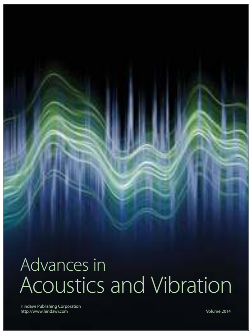
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration