IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 23, 2017, accepted October 9, 2017, date of publication October 17, 2017, date of current version November 7, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2764047

A Simple Multi-Objective Optimization Based on the Cross-Entropy Method

RODOLFO E. HABER[1], (Member, IEEE), GERARDO BERUVIDES[1], RAMÓN QUIZA[2], AND ALEJANDRO HERNANDEZ[2]

[1]Centre for Automation and Robotics, CSIC-UPM, 28500 Madrid, Spain
[2]Study Center on Advanced and Sustainable Manufacturing, University of Matanzas, Matanzas 44740, Cuba

Corresponding author: Rodolfo E. Haber (rodolfo.haber@car.upm-csic.es)

This work was supported by ECSEL under Grant 692480.

ABSTRACT A simple multi-objective cross-entropy method is presented in this paper, with only four parameters that facilitate the initial setting and tuning of the proposed strategy. The effects of these parameters on improved performance are analyzed on the basis of well-known test suites. The histogram interval number and the elite fraction had no significant influence on the execution time, so their respective values could be selected to maximize the quality of the Pareto front. On the contrary, the epoch number and the working population size had an impact on both the execution time and the quality of the Pareto front. Studying the rationale behind this behavior, we obtained clear guidelines for setting the most appropriate values, according to the characteristics of the problem under consideration. Moreover, the suitability of this method is analyzed based on a comparative study with other multi-objective optimization strategies. While the behavior of simple test suites was similar to all methods under consideration, the proposed algorithm outperformed the other methods considered in this paper in complex problems, with many decision variables. Finally, the efficiency of the proposed method is corroborated in a real case study represented by a two-objective optimization of the microdrilling process. The proposed strategy performed better than the other methods with a higher hyperarea and a shorter execution time.

INDEX TERMS Cross-entropy method, multi-objective optimization, tuning parameters.

I. INTRODUCTION

Optimization plays a key role in modern science and engineering. Multi-objective optimization has emerged as a suitable choice for solving a wide range of technical problems associated with real-world applications that often involve multiple and conflicting objectives [1], [2]. Nevertheless, no single point in non-trivial problems will minimize all given objective functions at once [3]. Solving these kinds of problems is a difficult task, especially for non-convex problems [4]. Furthermore, these problems usually include highly complex constraints and uncertainties [5].

Two main approaches can be followed for solving multi-objective optimization problems. In the a priori techniques, several objectives are combined into a single one, either by aggregation or by supplying some preferential information. An approach that actually transforms the multi-objective problem into a single-objective one, prior to optimization is suggested in [6].

The second approach generates a set of optimal solutions in the wider sense, i.e., when all the objectives are simultaneously considered in the feasible region, then there are no other superior solutions [7]. These solutions are known as the Pareto set and the set of the corresponding objective vectors are known as the Pareto front [8].

Evolutionary computation is inspired by ideas of natural evolution and adaptation [9]. Nowadays, there are dozens of successful applications of genetic algorithms [10], particle swarm optimization [10], simulated annealing [11] and cross-entropy [12] reported in the literature. Due to their population-based nature, evolutionary algorithms are able to approximate the whole Pareto front in a single run [13]–[15].

The first cross-entropy method was introduced for probability estimation of rare events. The cross-entropy method has since been adapted to multi-objective optimization [16]–[19]. Its effectiveness at solving not only typical test problems, but also real-world challenges has been clearly demonstrated [20], [21].

Nowadays, evolutionary algorithms are usually highly sensitive to variations in their parameters [22] and their behavior and performance are usually studied in experimental contexts that seek to give greater solidity to the existing theoretical foundations [23]. So, selection of the proper values for the algorithm parameters is one of the most demanding issues in the application of evolutionary computation.

This paper introduces some modifications to the multi-objective cross-entropy method previously proposed by Beruvides *et al.* [24], in order to improve the setting of parameters and convergence. Firstly, the sensitivity of the algorithm to the remaining parameters is experimentally studied. A 2-level half-fraction with a center point experimental design was selected to evaluate the influence of the epoch number, the working population size, the histogram intervals number, and the elite fraction, on the quality of the Pareto front. Subsequently, a surface analysis is performed to obtain the relationship between the epoch number and the population size with the execution time and the hyperarea ratio. Finally, the performance of the proposed method is compared with those from other evolutionary approaches, showing better results with similar execution times.

The paper is organized in six sections. Following this introduction, the algorithm description is presented. The third section introduces a sensitivity analysis of the parameter values in relation to Pareto front quality. Subsequently, a comparative study with other techniques to tackle well-known test suites is depicted in section 4. A practical example to demonstrate the efficacy of the proposed method is shown in section 5. A two-objective optimization problem of the micro-drilling process is presented. Finally, the main conclusions are outlined and future works discussed.

II. ALGORITHM DESCRIPTION

A. GENERAL DESCRIPTION

The proposed Simple Multi-Objective Cross-Entropy (SMOCE) method (see Fig. 1) is based on the MOCE+ algorithm [24]. Similar to any multi-objective optimization technique, SMOCE aims to solve the following problem:

$$\min(\mathbf{y}) = \mathbf{F}(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n, \quad \mathbf{y} \in \mathbb{R}^m \tag{1}$$

where:

$$l_i \leq x_i \leq u_i, \quad i = \{1, \ldots, n\} \tag{2}$$

which is constrained by:

$$g_i(\mathbf{x}) \leq 0, \quad i = \{1, \ldots, p\}. \tag{3}$$

At the core of SMOCE is the working population, at epoch t:

$$\mathbf{Q}^{(t)} = \{(\mathbf{x}_1, \mathbf{y}_1)^{(t)}, \ldots, (\mathbf{x}_Z, \mathbf{y}_Z)^{(t)}\}; \tag{4}$$

composed of the Z solutions $\mathbf{x}_k = \{x_{k,1}, \ldots, x_{k,n}\}$ and their respective m-th evaluated objective functions f_m for the k-th vector, $\mathbf{x}_k$,

$$\mathbf{y}_k = \{y_{k,1} = f_1(\mathbf{x}_k), \ldots, y_{k,m} = f_m(\mathbf{x}_k)\}. \tag{5}$$

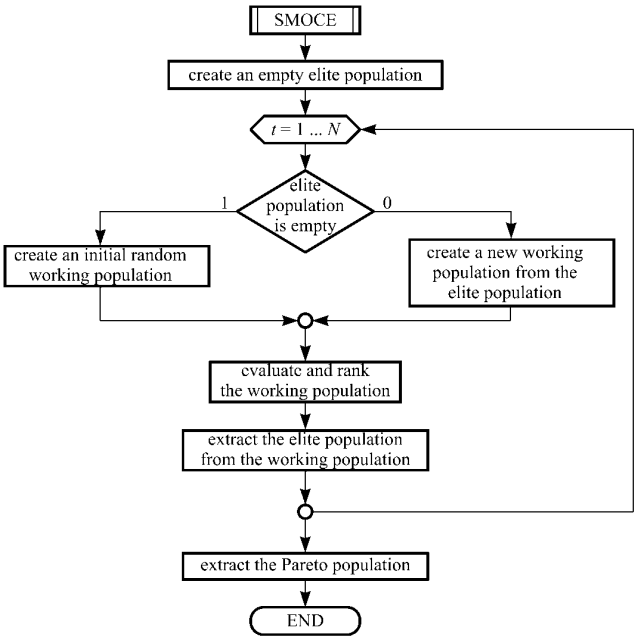


FIGURE 1. Block diagram of the simple multi-objective cross-entropy (SMOCE) algorithm.

The evolutionary process takes place on a loop with a unique ending condition: arrival at the epoch number, N. These are the two main differences between SMOCE, the method addressed in this paper, and MOCE+, where two nested loops and three different ending conditions are considered. These modifications permit, on the one hand, a reduction of objective function evaluations and, on the other hand, the removal of some parameters such as the maximum evaluation number and the convergence limit. Although these parameters were introduced in the first version [24], in order to stop the execution as soon as the Pareto front reached the point of stability near the actual front, under practical conditions, the main parameters, such as the epoch number, were set ad-hoc. Consequently, removing some of them will help to simplify the optimization method without loss of actual effectiveness.

In the first epoch, an initial working population is randomly created; in the following epochs, a new population, $Q(t)$, is created from the previous one, $Q(t-1)$. The corresponding values of the objective function for each solution are evaluated after creating the population.

Considering the elite solutions that are extracted from the current working population:

$$\mathbf{\Theta}^{(t)} = \{(\boldsymbol{\xi}_1, \boldsymbol{\upsilon}_1)^{(t)}, \ldots, (\boldsymbol{\xi}_E, \boldsymbol{\upsilon}_E)^{(t)}\}; \tag{6}$$

where, $(\boldsymbol{\xi}_i, \boldsymbol{\upsilon}_i)^{(t)}$ represents the *i-th* working population for $(\mathbf{x}_i, \mathbf{y}_i)^{(t)}$. $E = \alpha Z$ is the elite solution number and α is the elite fraction, both of which are required parameters of the algorithm. These elite solutions are included in the next epoch population, which introduces elitism in SMOCE (another improvement with respect to the previous MOCE+). In the

following sections, the main steps of the SMOCE algorithms are explained in further detail.

B. GENERATING THE INITIAL POPULATION

Another difference between the (SMOCE) method that is proposed here and the former (MOCE+) method is the generation of the initial population. In the MOCE+ method, the initial population is generated by using a normal random distribution, while a uniform random distribution is used here, i.e.:

$$x_{i,j} = \mathcal{U}(l_i, u_i), \quad i = \{1, \ldots, n\},\ j = \{1, \ldots, Z\}; \tag{7}$$

where, $x_{i,j}$ is the value of the i-th decision variable in the j-th solution; and, $\mathcal{U}(l_i, u_i)$ is the uniform random distribution in the interval $[l_i, u_i]$.

This distribution means that uniformly distributed individuals can be generated throughout the decision variables domain, which is especially convenient for dealing with problems where the Pareto solutions are concentrated in a small section of the whole domain. When using a normal distribution, the higher probability is close to the mean value, which is either arbitrarily chosen or assumed in the center of the variable interval. Evidently, the Pareto solution may be located at a considerable distance from the selected mean value, increasing the algorithm effort that is needed to reach certain positions. On the contrary, uniform (flat) probability distributions guarantee equal probability for creating initial solutions in any region of the decision space.

C. GENERATING A NEW POPULATION

The generation of the new working population, from the current one at each epoch, is another important difference between the two methods. In this new approach, the elite solutions, $\{(\boldsymbol{\xi}_1, \boldsymbol{v}_1), \ldots, (\boldsymbol{\xi}_E, \boldsymbol{v}_E)\}$, are clustered by using the histogram of the objective functions, instead of the histogram of the decision variables. This approach appears to be most appropriate because, a small variation in the decision space may often lead to a large variation in the objective space.

D intervals are created in each dimension of the objective space, to establish the histogram:

$$[\underline{c}_{i,k}, \overline{c}_{i,k}], \quad i = \{1, \ldots, m\},\ k = \{1, \ldots, D\}; \tag{8}$$

where the lower and upper bounds for each interval are:

$$\underline{c}_{i,k} = b_i^{\min} + \frac{(k-1)}{D}(b_i^{\max} - b_i^{\min}); \tag{9}$$

and

$$\overline{c}_{i,k} = b_i^{\min} + \frac{k}{D}(b_i^{\max} - b_i^{\min}); \tag{10}$$

and where:

$$b_i^{\min} = \min(\{v_{i,1}, \ldots, v_{i,E}\}), \quad i = \{1, \ldots, m\}; \tag{11}$$

and

$$b_i^{\max} = \max(\{v_{i,1}, \ldots, v_{i,E}\}), \quad i = \{1, \ldots, m\}; \tag{12}$$

are the minimum and maximum values of the *i-th* objective in the elite population.

The D intervals obtained in this way are then combined with the m variables, to obtain D^m classes, and then all the αZ elitist solutions are arranged into these classes. Thereafter, the mean value and the standard deviations are computed from the solutions of each class for each objective function:

$$\mu_{i,k} = \frac{1}{\#\{\boldsymbol{\xi}^{(k)}\}} \sum_{j=1}^{\#\{\xi^{(k)}\}} \xi_{i,j}^{(k)}, \quad i = \{1, \ldots, m\}, \quad k = \{1, \ldots, D^m\}; \tag{13}$$

$$\sigma_{i,k} = \sqrt{\frac{\sum_{j=1}^{\#\{v^{(k)}\}} (\xi_{i,j}^{(k)} - \mu_{i,k})^2}{\#\{\boldsymbol{\xi}^{(k)}\} - 1}}, \quad i = \{1, \ldots, m\}, \quad k = \{1, \ldots, D^m\}; \tag{14}$$

where, #{A} denotes the cardinality (i.e., the number of elements) of the set {A}, and $\{(\boldsymbol{\xi}, \boldsymbol{v})^{(k)}\}$ is the subset of the elite population belonging to the k-$^{\text{th}}$ class, i.e.:

$$\{(\boldsymbol{\xi}, \boldsymbol{v})\}^{(k)} \stackrel{\text{def}}{=} \{(\boldsymbol{\xi}, \boldsymbol{v}) \in \boldsymbol{\Theta}^{(t)} : \underline{c}_{i,k} \le \boldsymbol{v}_i \le \overline{c}_{i,k}, \ \forall \boldsymbol{i} = \{1, \ldots, m\}\}. \tag{15}$$

Lastly, the new working population is composed of the E elite solutions and the $z^{(k)}$, $k = \{1, \ldots, D^m\}$, solutions created from the D^m classes, giving:

$$z^{(k)} = \frac{\#\{(\boldsymbol{\xi}, \boldsymbol{v})^{(k)}\}(Z - E)}{E}, \quad k = \{1, \ldots, D^m\}. \tag{16}$$

These solutions from each class, are created by using normal random distributions with a mean, $\mu_{i,k}$, and a standard deviation, $\sigma_{i,k}$, and truncated to the interval $[l_i, u_i]$, for $i = \{1, \ldots, m\}$:

$$x_{i,j}^{(t)} = \mathcal{N}(\mu_{i,k}, \sigma_{i,k}, l_i, u_i), \quad i = \{1, \ldots, n\}. \tag{17}$$

D. EVALUATING THE POPULATION

The evaluation of the solutions not only implies computing the values of the objective functions, but also the constraints, which are considered by the penalty method. Therefore, the constrained objective functions take the following form:

$$y_{i,k}^{(t)} = f_i(\mathbf{x}_k^{(t)}) + \sum_{j=1}^{p} \gamma_j g_j(\mathbf{x}_k^{(t)}), \quad i = \{1, \ldots, n\}, \quad k = \{1, \ldots, Z\}; \tag{18}$$

where, $\gamma_j \ge 0$ are the penalty coefficients assigned to each constraint.

E. EXTRACTING THE ELITIST POPULATION

Elitism is considered in this method by including the elite solutions in the working population of the next epoch.

The selection of the elitist population is also different in the new approach. In the former (MOCE+) method, the elite population includes all individuals with a lower rank than some threshold value (which decreases from some initial value to

TABLE 1. Levels of the algorithm parameters for the screening.

Algorithm parameter	Level values		
	Low	Medium	High
Epoch number, N	10	2505	5000
Working population size, Z	50	525	1000
Histogram intervals number, D	5	15	25
Elite fraction, α	0.10	0.35	0.60

zero, through the algorithm execution). On the contrary, in the new (SMOCE) method, a prescribed fraction of the working population, including the individuals with the lowest ranks, is selected as the elite population in each epoch. This selection is carried out through the Pareto ranking criterion, based on vector dominance concepts. The Pareto rank of a vector, in a vector set, is the number of the other vectors which dominate it. The elite population is composed of those E solutions with the lowest Pareto rank.

III. SENSITIVITY ANALYSIS OF THE ALGORITHM PARAMETER VALUES

A. SCREENING

The first step in the selection of the most appropriate values for the SMOCE parameters is to determine their influence both on the Pareto front quality and on the execution time. All the simulations were performed on a personal computer with an Intel Core i3-2120 CPU (3.30 GHz) and 4.0 GB RAM. During the experiments, all the computer resources were taken up by the proposed optimization method and no other applications nor communication tasks were running simultaneously. The execution time is more inclusive rather than the computing time for evaluation objective functions, because it comprises other tasks such as sorting solutions and the generation of new populations.

The four parameters of the SMOCE (epoch number, N; working population size, Z; histogram intervals number, D; and elite fraction, α); were considered in the intervals shown in Table 1. A 2-level half-fraction with center point design for the experiments was selected for the simulations. Five replicates were executed for each experimental point.

Thirteen commonly used test problems, from three suites conventionally known as MOP [25], ZDT [26] and WFG [27], were considered in the simulations. All of them have two objectives, but are very different with regard to their other features (see Table 2).

Four metrics were used to evaluate the quality of the Pareto fronts: the hyperarea ratio, *HR*; generational distance, *GD*; convergence, *CV*; and spacing, *SP*. While generational distance and convergence measure the front convergence (i.e., how close the front that is estimated is to the true Pareto front), the spacing reflects its diversity (i.e., how uniformly the solutions are distributed throughout the Pareto front), and the hyperarea ratio combines both criteria.

TABLE 2. Features of the selected test problems.

Problem	Variables	Pareto front geometry	Modality
MOP1	1	Convex	Unimodal/Unimodal
MOP2	3	Concave	Unimodal/Unimodal
MOP3	2	Disconnected	Multimodal/Unimodal
MOP4	3	Disconnected	Multimodal/Unimodal
MOP6	2	Disconnected	Unimodal/Multimodal
ZDT1	30	Convex	Multimodal/Unimodal
ZDT2	30	Concave	Multimodal/Unimodal
ZDT3	30	Disconnected	Unimodal/Multimodal
WFG2	32	Disconnected	Unimodal/Multimodal
WFG3	32	Degenerated	Unimodal/Unimodal
WFG4	32	Concave	Multimodal
WFG5	32	Concave	Deceptive
WFG6	32	Concave	Unimodal

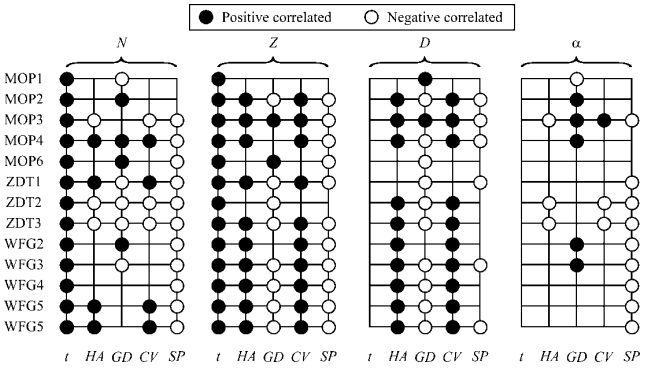


FIGURE 2. Relationships resulting from the screening analysis.

The effect of each parameter was evaluated through a multiple regression, by considering the Student t-test of the corresponding coefficients. The relationship is considered significant at a 95% confidence level. Fig. 2 represents the relationships that were obtained. The epoch number has a direct relationship with the execution time in all the test problems under consideration. This fact was expected, in view of the structure of the SMOCE algorithm, because this parameter represents the only ending condition that was included. An increase in the epoch number, also improves the front diversity (expressed by an inverse relationship with the spacing) in most of the test problems (except in MOP1 and MOP2). However, there is no significant relationship with other metrics.

A rise in the population size increases the execution time for all of the test problems. It also improves the (convergence) and the (diversity) quality of the Pareto front that is obtained in most of the test problems.

The execution time is not affected by the histogram interval number. A rise in this parameter causes an improvement in the front convergence (shown in the direct relationship with the convergence metric and the inverse relationship with the generational distance for most of the test problems), and in front diversity (indicated by the inverse relationship with the spacing). There is also a direct relationship with the hyperarea

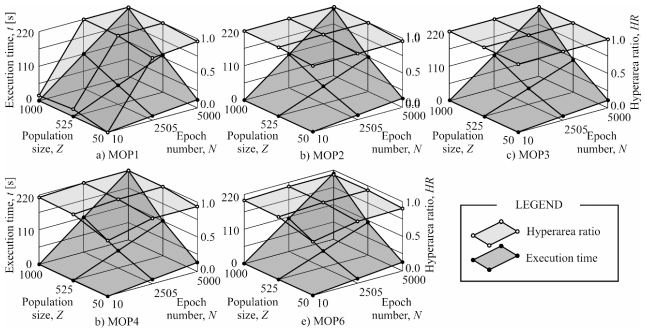


FIGURE 3. Execution time and hyperarea ratio vs. population size and epoch number for MOP test-problem suite.

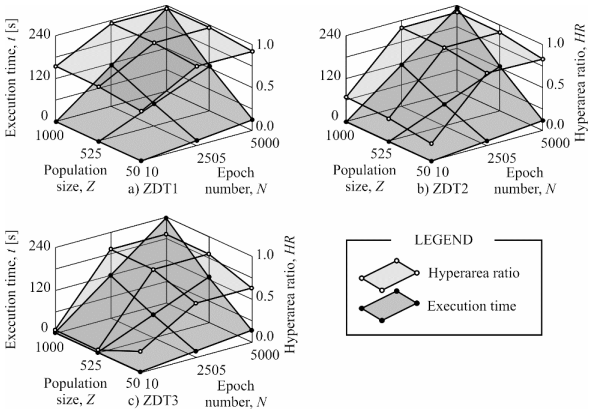


FIGURE 4. Execution time and hyperarea ratio vs. population size and epoch number behavior for the ZDT test-problem suite.

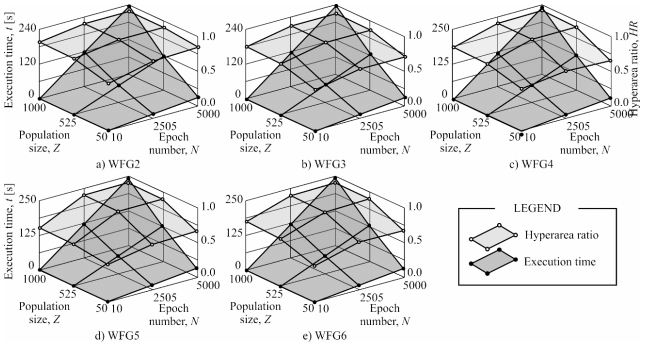


FIGURE 5. Execution time and hyperarea ratio vs. population size and epoch number for WFG test-problem suite.

for most of the problems. Finally, the elite ratio has no influence on the execution time and the quality metrics, except in the spacing for the problems in sets ZDT and WFG, where an improvement was noted.

Considering the previous analyses, the values of the histogram interval number and the elite fraction were chosen at their respective higher levels (i.e., $D = 25$ and $\alpha = 0.65$), because they yielded a better quality Pareto front without shortening the execution time.

B. RESPONSE SURFACE

The following response surface analysis was performed to obtain the relationship between both the epoch number and population size with the execution time and Pareto front quality.

A full 3-level experimental design was selected, in order to complete the study. Both factors (population size, Z, and epoch number, N) were held at their respective levels and used as screening (see Table 1). Twenty replications were performed at each experimental point.

The hyperarea ratio was selected for the analysis of Pareto front quality, because this parameter characterizes both convergence and diversity. Fig. 3 shows the graphical representation of the execution time and hyperarea ratio obtained for each experimental level for the MOP suite.

Both, the population size and the epoch number, have a direct impact on the execution time as shown in Fig. 3. On the contrary, they only increase the front quality up to medium levels. After this point, there is no significant change in the front quality. Therefore, the most convenient values for the experimental factors are those corresponding (or near) to the middle levels, i.e., $Z = 525$, $N = 2505$.

It is important to point out that the hyperarea ratio values obtained for this suite are very high. Except for the MOP6 problem, the higher values are near to one. This result means that the Pareto front that is modeled is close to the theoretical solution.

Figs. 4 and 5 show the results for suites ZDT and WFG, respectively. Similar behavior is depicted in both the execution time and the hyperarea ratio. In every case, the combination of medium levels (Z = 525, N = 2505) means that we can obtain near-optimal front quality within a reasonable execution time.

The most noticeable difference is shown by the lower values of the hyperarea ratio in the two last test suites compared with MOP. This difference may be due to the higher complexity of the problems in these suites. Nevertheless, the differences in execution times are negligible. Accordingly, the execution time depends on the number of evaluations, but not on the complexity of the optimization problem (number of variables, characteristics of the target function, etc.).

It should also be noted that the quality of the Pareto front cannot be improved beyond some region independently of any increase in the algorithm parameters (population size and epoch number), at least, over the intervals under consideration.

IV. COMPARISON WITH OTHER HEURISTIC METHODS

The WFG test suite was selected for comparing the performance of the SMOCE with some of the most popular multi-objective optimization heuristics due to the complexity of this benchmark. The comparison was carried out not only with the base (MOCE+) algorithm [24], but also taking into account the following strategies:

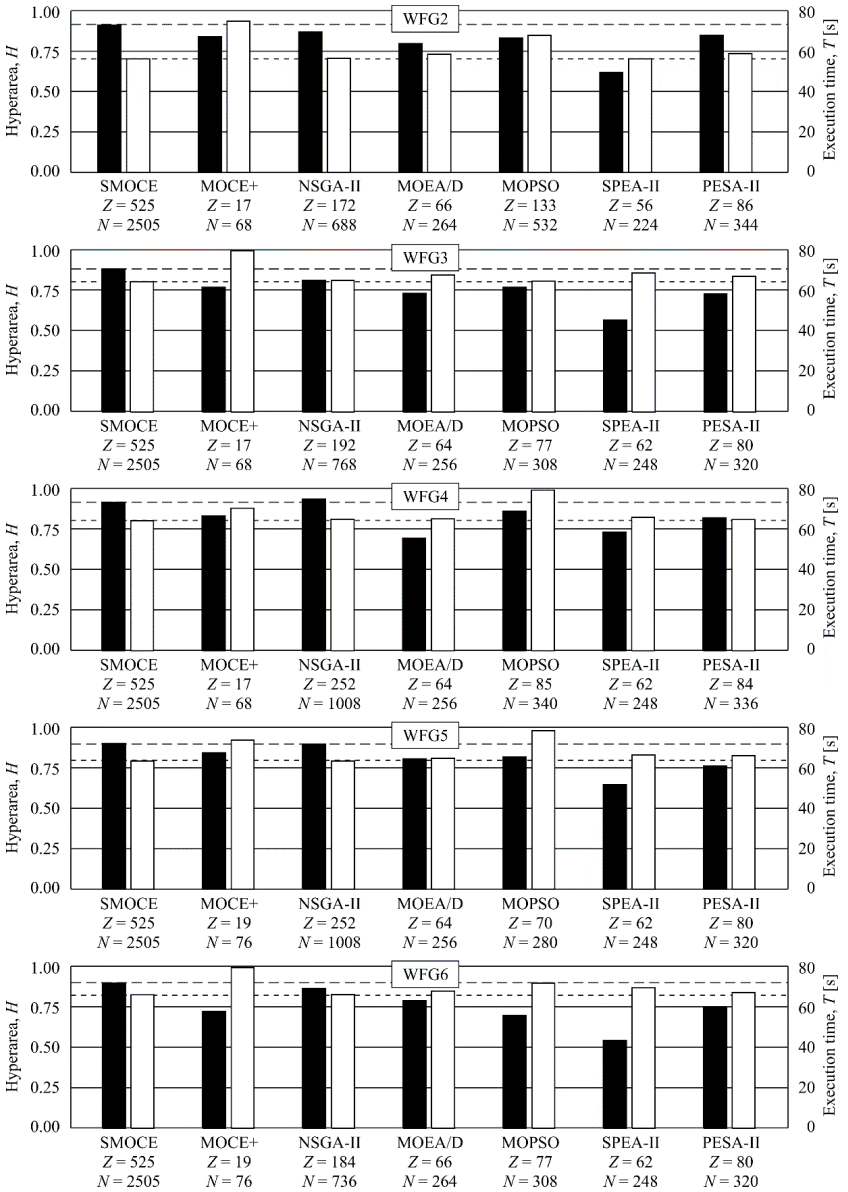


FIGURE 6. **Performance comparison of SMOCE and other optimization heuristics.**

- Non-Sorting Genetic Algorithm II (NSGA-II) [28].
- Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) [29].
- Multi-Objective Particle Swarm Optimization (MOPSO) [30].
- Strength Pareto Evolutionary Algorithm (SPEA-II) [31].
- Pareto Archived Evolutionary Strategy (PESA-II) [32].

The SMOCE parameters were selected on the basis of the results shown in section 3, corresponding to the medium level values of population size and epoch number, because after these values, there is no remarkable increment in the hyperarea ratio.

The population size and epoch number of the other heuristics were selected in the ratio 1:4, i.e., the epoch number that is four times higher than the population sizes. Both values were chosen so that the execution time would be approximately equal (actually, slightly higher) than that of the SMOCE. In all, 25 replications were performed for each heuristic, in order to ensure algorithm convergence.

Fig. 6 depicts the main results. The effectiveness of the SMOCE method (i.e., the average hyperarea ratio), for a similar computational effort (given by the execution time), is higher than the other strategies considered in the comparison, except for NSGA-II in relation to the specific benchmark WFG4. It is important to remark that NSGA II has less Z and N, but the execution time is longer than the proposed method. The main rationale for this behavior is linked to the computing time, not only used in evaluating the objective functions, but also in other tasks such as sorting solutions and generating new populations.

FIGURE 7. Experimental setup.

Although the tests are not enough to consider that SMOCE absolutely outperforms the other approaches under any circumstance, these outcomes clearly support the statement on the effectiveness and efficiency of the proposed algorithms and its potential application to real world problems.

V. TWO-OBJECTIVE OPTIMIZATION OF A MICRODRILLING PROCESS. A PRACTICAL EXAMPLE.

Micro-scale manufacturing processes are non-linear, time-variant processes that are difficult to represent with precise mathematical equations. Among them, micro-mechanical drilling has been widely applied and reported in various applications. For instance, drilling of micro-holes is a key process in laminated printed circuit boards. Nevertheless, this widely applied process brings additional complexity to the cutting mechanism as the edge radius is comparable to chip thickness at low feeds [33], [34].

The selection of the optimal cutting conditions is a key issue in micro-manufacturing processes. However, the development of accurate models that relate objectives and constraints with decision variables is not straightforward. Likewise, computationally efficient strategies have to be prepared before the optimization problem is in itself addressed.

A two-objective optimization problem of a microdrilling process is selected, in order to demonstrate the efficiency of the proposed algorithm in a real scenario. A two-objective optimization problem focused on both production and quality maximization is the most common challenge from the industrial viewpoint. Indeed, this optimization can be considered relatively simple from the academic viewpoint, but it is truly challenging from the micro-manufacturing perspective. The microdrilling process of a titanium-aluminum-vanadium alloy (Ti6Al4V) was considered. A Kern-Evo high-precision machining center (see Fig. 7) was used in the experimental setup, equipped with a Kistler Minidyn 9256 piezoelectric force dynamometer. Drills with a diameter of 0.2 mm were

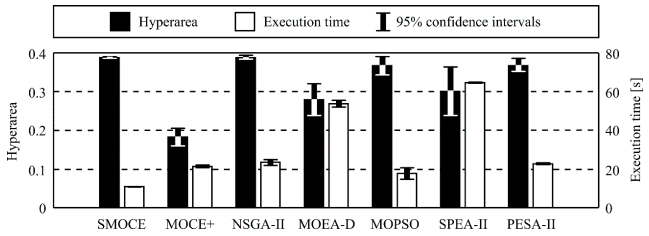


FIGURE 8. Results of the optimization methods applied to the microdrilling process.

used for drilling 0.8 mm-deep holes. The following cutting-parameter intervals (cutting speed, v; feed rate, f; and peck drilling step, s), which are also the decision variables of the optimization problem, were considered:

$$\begin{aligned} 9.4\ \text{m/min} &\leq v \leq 27.6\ \text{m/min} \\ 10\ \text{mm/min} &\leq f \leq 200\ \text{mm/min} \\ 0.02\ \text{mm} &\leq s \leq 0.04\ \text{mm} \end{aligned} \tag{19}$$

The first objective was the drilling time, τ, which can be computed by:

$$\tau = \frac{s}{f_0}\left[\left(\frac{h_\text{d}}{s}\right)^2 + \left(\frac{h_\text{d}}{s}\right) - 2\right] + \frac{h_\text{d}}{f_z} \tag{20}$$

where, h_d is the hole depth and f_0 is the fast-feed used for the backward motion. This objective represents the productivity process, as the lower the drilling time the higher the number of manufactured holes in a given period.

The second optimization objective is the amplitude of the vibrations, A, on the plane perpendicular to the drilling axis. This goal characterizes the quality of the manufactured holes and it is modeled using a two-layer feed-forward neural network (i.e., a multi-layer perceptron):

$$A = \Phi_\text{NN}(v, f, s) \tag{21}$$

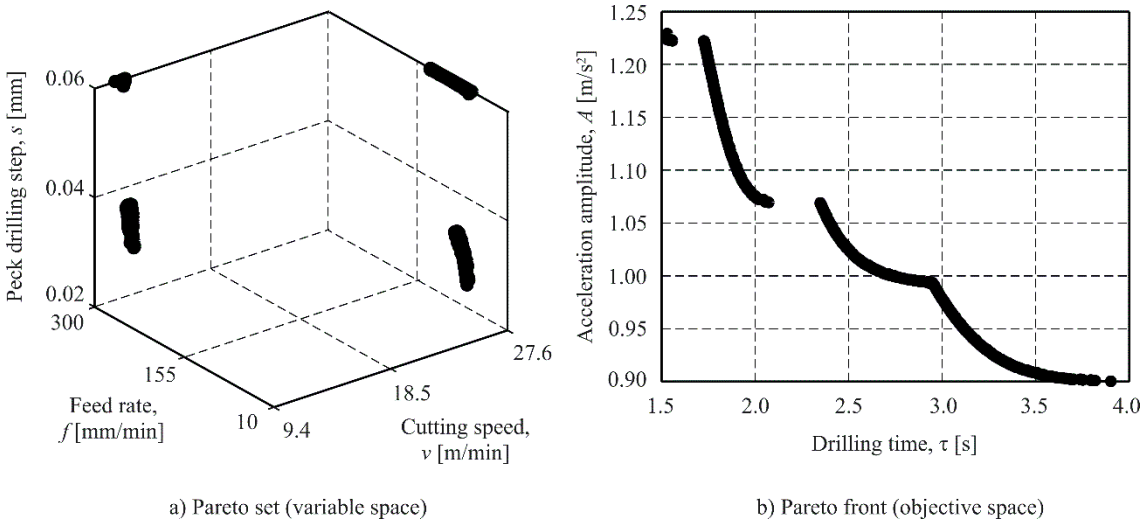
where, Φ_NN is a function described by:

$$\Phi_\text{NN}(\mathbf{x}) = c + \sum_{j=1}^{N}\left\{\frac{2w_j}{1+\exp\left[-2\left(b_j + \sum_{i=1}^{M} u_{ij}x_i\right)\right]} - 1\right\} \tag{22}$$

where $\mathbf{x} = \{x_i\} = \{v, f, s\}$ is the input vector; $\mathbf{U} = \{u_{ij}\}$, $i = \{1 \ldots M\}$, $j = \{1 \ldots N\}$, is the weights matrix of the hidden layer; $\mathbf{B} = \{b_j\}, j = \{1 \ldots N\}$, is the biases vector of the hidden layer; $\mathbf{W} = \{w_j\}, j = \{1 \ldots N\}$, is the weights vector of the output layer; and $\mathbf{C} = \{c\}$ is the bias of the unique neuron of the output layer. Moreover, all implemented functions for the case study are available in a repository [35].

It is obvious that both objectives should be minimized by means of a high-productivity drilling process that guarantees high-hole quality (closely related with the amplitude of vibrations).

Furthermore, an important constraint should be considered regarding the thrust force, F_z, which should be lower than the permissible thrust force, F_al, that is pre-established to avoid



a) Pareto set (variable space)

b) Pareto front (objective space)

FIGURE 9. Pareto set (a) and Pareto front (b) calculated by SMOCE.

buckling-related breakage of the tool. This constraint can be expressed by:

$$F_z \leq F_{\text{al}}; \tag{23}$$

which, for a more convenient homogeneous representation, can be rearranged as:

$$\frac{F_z}{F_{\text{al}}} - 1 \leq 0. \tag{24}$$

The thrust force, F_z, is also modeled using a multilayer perceptron similar to (22):

$$F_z = \Phi_{\text{NN}}(v, f, s) \tag{25}$$

fitted from the experimental data. The corresponding weights and biases are given in [35]. The permissible force can be determined with Euler's equation:

$$F_{\text{al}} = \eta \frac{\pi^2 E I_{\min}}{\mu L} \tag{26}$$

where, $E = 650$ MPa is the Young's modulus of the tool material, $I_{\min} = 9.52 \times 10^{-6}$ mm^4 is the minimum area moment of inertia of the drill cross-section, $\mu = 2$ is the coefficient that takes into account the boundary conditions, $L = 2.5$ mm is the length of the drill flute, and $\eta = 0.5$ is a security factor.

The optimization process was not only performed with the proposed SMOCE algorithms but also with other techniques used in the previous section. SMOCE and NSGA-II had the same values for population size and epochs number (i.e., 500), whereas for MOCE+, MOEAD, MOPSO and PESA-II, the value, for both population size and epoch number, was 200. Finally, the population size and the epoch number were set to 200 and 50, respectively for the SPEA-II method. These parameters were chosen by simulations, in order to obtain the best Pareto front quality on the basis of the hyperarea and the execution time.

In all, 25 replicates were run for each algorithm, in order to analyze the convergence of the obtained Pareto front. Fig. 8 depicts the hyperarea and the execution time from each method considered in the comparative study. SMOCE showed both a higher hyperarea and a lower execution time than the other methods. Only the NSGA-II method yielded a similar hyperarea but with a lengthier execution time. Another important point relates to the convergence of the Pareto fronts. The results of SMOCE show a lower spread than the other techniques, meaning better convergence and, therefore, higher reliability.

The Pareto set and the corresponding Pareto front, computed with SMOCE, are shown in Fig. 9 as well as the distribution of the non-dominated solutions. Solutions are grouped into four well-differentiated sets: two corresponding to higher values of cutting speeds and, the others, to the higher values of the peck-drilling steps. Selecting the most convenient solution relies on the technologist or the machine tool operator, depending on the specific workshop conditions. If the part or component quality is an important aspect, solutions with lower vibrations will be selected, but higher drilling time, should be selected. On the contrary, if productivity plays a key role, the lower drilling time, corresponding to the higher vibration levels, should be selected. Finally, other trade-off solutions can be selected in other non-extreme conditions.

VI. CONCLUSION

This paper has presented a simple multi-objective cross-entropy method. Only four parameters (epoch number, working population size, histogram interval number, and elite fraction) are required, in order to perform the tuning process.

The analysis on the basis of well-known test suites has demonstrated that the histogram interval number and the elite fraction had no impact on improving the performance of the proposed method, and therefore both were removed from the

study. On the contrary, the epoch number and the working population size had a remarkable influence on the execution time. Likewise, the epoch number and the working population size clearly influenced the Pareto front quality, up to a level, where the quality was no longer improved. It should be noted that this behavior was similar for all test suites considered in this study.

The proposed method has also been compared with other multi-objective optimization strategies to address well-known test suites. The proposed strategy showed better results in terms of hyperarea ratio for similar execution times. Finally, the efficacy of the proposed method has been demonstrated in a real case study represented by a two-objective optimization problem of the microdrilling process. The proposed strategy outperformed the other methods with higher hyperarea and shorter execution time.

Two important issues will be addressed in future works. The first one is the way the algorithm deals with constraints, which is based on the penalty method. This approach has usually been highly dependent on the penalty values, which are often selected by trial and error. Secondly, the self-adaptiveness of the parameters has the potential to improve both the effectiveness and the efficiency of the algorithm, so the introduction of this capability will be considered in further studies.

REFERENCES

[1] D. Alanis, P. Botsinis, S. X. Ng, and L. Hanzo, "Quantum-assisted routing optimization for self-organizing networks," *IEEE Access*, vol. 2, pp. 614–632, 2014.
[2] T.-K. Liu, Y.-P. Chen, and J.-H. Chou, "Developing a multiobjective optimization scheduling system for a screw manufacturer: A refined genetic algorithm approach," *IEEE Access*, vol. 2, pp. 356–364, 2014.
[3] J. X. Da Cruz Neto, G. J. P. Da Silva, O. P. Ferreira, and J. O. Lopes, "A subgradient method for multiobjective optimization," *Comput. Optim. Appl.*, vol. 54, no. 3, pp. 461–472, 2013.
[4] S. Deshpande, L. T. Watson, and R. A. Canfield, "Pareto front approximation using a hybrid approach," *Procedia Comput. Sci.*, vol. 18, pp. 521–530, 2013.
[5] X.-S. Yang, S. Koziel, and L. Leifsson, "Computational optimization, modelling and simulation: Past, present and future," *Procedia Comput. Sci.*, vol. 29, pp. 754–758, 2014.
[6] D. A. Van Veldhuizen and G. B. Lamont, "Multiobjective evolutionary algorithms: Analyzing the state-of-the-art," *Evol. Comput.*, vol. 8, no. 2, pp. 125–147, Mar. 2000.
[7] H. A. Abbass, R. Sarker, and C. Newton, "PDE: A Pareto-frontier differential evolution approach for multi-objective optimization problems," in *Proc. Congr. Evol. Comput.*, vol. 2. Piscataway, NJ, USA, May 2001, pp. 971–978.
[8] M. Li, L. Liu, and D. Lin, "A fast steady-state ε-dominance multi-objective evolutionary algorithm," *Comput. Optim. Appl.*, vol. 48, no. 1, pp. 109–138, 2011.
[9] R. Sarker, M. Mohammadian, and X. Yao, *Evolutionary Optimization* (International Series in Operations Research & Management Science). New York, NY, USA: Kluwer, 2003.
[10] R.-I. Chang, H.-M. Hsu, S.-Y. Lin, C.-C. Chang, and J.-M. Ho, "Query-based learning for dynamic particle swarm optimization," *IEEE Access*, vol. 5, pp. 7648–7658, 2017.
[11] S. Deng, C. Yuan, J. Yang, and A. Zhou, "Distributed mining for content filtering function based on simulated annealing and gene expression programming in active distribution network," *IEEE Access*, vol. 5, pp. 2319–2328, 2017.
[12] R. E. Haber, C. Juanes, R. del Toro, and G. Beruvides, "Artificial cognitive control with self-X capabilities: A case study of a micro-manufacturing process," *Comput. Ind.*, vol. 74, pp. 135–150, Dec. 2015.
[13] R. E. Precup, R. C. David, E. M. Petriu, M. B. Radac, and S. Preitl, "Adaptive GSA-based optimal tuning of PI controlled servo systems with reduced process parametric sensitivity, robust stability and controller robustness," *IEEE Trans. Cybern.*, vol. 44, no. 11, pp. 1997–2009, Nov. 2014.
[14] M. B. Rădac, R.-E. Precup, E. M. Petriu, and S. Preitl, "Iterative data-driven tuning of controllers for nonlinear systems with constraints," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6360–6368, Nov. 2014.
[15] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 32–49, 2011.
[16] D. P. Kroese, S. Porotsky, and R. Y. Rubinstein, "The cross-entropy method for continuous multi-extremal optimization," *Methodol. Comput. Appl. Probab.*, vol. 8, no. 3, pp. 383–407, 2006.
[17] J. Bekker and C. Aldrich, "The cross-entropy method in multi-objective optimisation: An assessment," *Eur. J. Oper. Res.*, vol. 211, no. 1, pp. 112–121, 2011.
[18] C. Hauman, "The application of the cross-entropy method for multi-objective optimisation to combinatorial problems," M.S. thesis, Fac. Eng., Stellenbosch Univ., Stellenbosch, South Africa, 2012.
[19] K. Sebaa, A. Tlemçani, M. Bouhedda, and N. Henini, "Multiobjective optimization using cross-entropy approach," *J. Optim.*, vol. 2013, Art. no. 270623, Sep. 2013.
[20] K. R. Chernyshov and E. P. Jharko, "Entropy criteria in the econometric model identification," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 827–832, 2016.
[21] M. A. Olivares-Mendez, L. Mejias, P. Campoy, and I. Mellado-Bataller, "Cross-entropy optimization for scaling factors of a fuzzy controller: A see-and-avoid approach for unmanned aerial systems," *J. Intell. Robot. Syst., Theory Appl.*, vol. 69, nos. 1–4, pp. 189–205, 2013.
[22] R. Wang, M. M. Mansor, R. C. Purshouse, and P. J. Fleming, "An analysis of parameter sensitivities of preference-inspired co-evolutionary algorithms," *Int. J. Syst. Sci.*, vol. 46, no. 13, pp. 2407–2420, 2015.
[23] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.
[24] G. Beruvides, R. Quiza, and R. E. Haber, "Multi-objective optimization based on an improved cross-entropy method. A case study of a micro-scale manufacturing process," *Inf. Sci.*, vols. 334–335, pp. 161–173, Mar. 2016.
[25] D. A. Van Veldhuizen, "Multiobjective evolutionary algorithms: Classifications, analyses, and new innovations," Ph.D. dissertation, Air Force Inst. Technol., Wright-Patterson AFB, OH, USA, 1999.
[26] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, 2000.
[27] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 477–506, Oct. 2006.
[28] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[29] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
[30] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 256–279, Jun. 2004.
[31] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," Swiss Federal Inst. Technol., Zürich, Switzerland, Tech. Rep. TIK-103, 2001.
[32] D. W. Corne, N. R. Jerram, J. D. Knowles, and M. J. Oates, "PESA-II: Region-based selection in evolutionary multiobjective optimization," presented at the 3rd Annu. Conf. Genet. Evol. Comput., 2002.
[33] G. Beruvides, R. Quiza, R. del Toro, and R. E. Haber, "Sensoring systems and signal analysis to monitor tool wear in microdrilling operations on a sintered tungsten–copper composite material," *Sens. Actuators, A, Phys.*, vol. 199, pp. 165–175, Sep. 2013.
[34] K. Sambhav, P. Tandon, S. G. Kapoor, and S. G. Dhande, "Mathematical modeling of cutting forces in microdrilling," *J. Manuf. Sci. Eng.*, vol. 135, no. 1, 2013, Art. no. 014501.
[35] GAMHE, CSIC Internal Repository. (2016). *Multi-Objective Optimization Cross Entropy (MOCE+) Function Repository*. [Online]. Available: http://gamhe.eu/downloads/?route=.%2FCEMOO%2FMOCE%2B

RODOLFO E. HABER (M'17) received the Ph.D. degree in industrial engineering from the Universidad Politécnica de Madrid, Spain, in 1999. He is Vice-Director of the Centre for Automation and Robotics, CSIC-UPM. In particular, he is now involved in two European projects related with cyber-physical systems: EMC2 and IoSENSE. He has authored 3 books, 20 book chapters, and over 60 articles in indexed journals and dozens of conference papers. His main research activities are focused on intelligent systems, control and supervisory systems, and artificial cognitive systems. Since 2002, he has belonged to the IFAC's Technical Committee 3.1 Computers for Control. Since 2005, he has been a member of the ASME Technical Committee on Model Identification and Intelligent Systems. Since 2007, he has been an Associate Editor of the *Transactions on Computational Science*, the *International Journal of Mechatronics and Manufacturing Systems*, the *World Scientific Journal*, the *International Journal of Computational Intelligence and Neurosciences*, and the *Journal of Control Engineering and Applied Informatics*.

GERARDO BERUVIDES received the Ph.D. degree (*cum laude*) in informatics and telecommunications from the Universidad Autónoma de Madrid, Spain, in 2017. He was an Assistant Professor with the University of Matanzas from 2010 to 2014. He has made pre-doctoral stays in several universities and research centers (in Germany, U.K., and Japan) carrying out research or academic tasks. He is an Assistant Researcher with the Centre for Automation and Robotics, CSIC-UPM, Madrid. His research interests are modeling artificial intelligence, metaheuristic optimization, and machine learning techniques applied to manufacturing processes. Furthermore, he has published several papers on these topics. He is a member of the reviewer board of several international journals.

RAMÓN QUIZA received the Ph.D. degree in manufacturing engineering from the Universidad de Matanzas, Cuba, in 2005. He has participated in the development of several software products. He has made post-doctoral stays in several universities and research centers (in Germany, Spain, Suriname, and Venezuela) carrying out research or academic tasks. He is currently the Director of the Study Centre on Advanced and Sustainable Manufacturing, University of Matanzas. He has published several papers and books contributions in the fields of applied artificial intelligence and modeling and optimization of manufacturing processes. His main research interests are optimization, artificial neural networks, artificial intelligence, and manufacturing processes. He is a member of the editorial board of three scientific journals.

ALEJANDRO HERNANDEZ received the M.Sc. degree in mechanical engineering from the University of Matanzas (UMCC), Cuba, in 2016. He is currently the Head of the Department of Mechanical Engineering, UMCC. His research interests are optimization of manufacturing processes and applied artificial intelligence. He has published several papers on these topics.