

# Preference Incorporation into Many-Objective Optimization:

## An Ant Colony Algorithm based on Interval Outranking

Gilberto RIVERA<sup>1</sup>, Carlos A. COELLO COELLO<sup>2,3</sup>, Laura CRUZ-REYES<sup>4</sup>, Eduardo R. FERNANDEZ<sup>5</sup>, Claudia GOMEZ-SANTILLAN<sup>4\*</sup>, and Nelson RANGEL-VALDEZ<sup>4</sup>

<sup>1</sup> División Multidisciplinaria de Ciudad Universitaria, Universidad Autónoma de Ciudad Juárez, 32579, Cd. Juárez, Chihuahua, Mexico

<sup>2</sup> CINEVESTAV-IPN, Evolutionary Computation Group, 07300, CDMX, Mexico

<sup>3</sup> Basque Center for Applied Mathematics (BCAM) & Ikerbasque, 48009 Bilbo, Bizkaia, Spain

<sup>4</sup> Postgraduate and Research Division, National Mexican Institute of Technology/Madero Institute of Technology, 89440 Ciudad Madero, Tamaulipas, Mexico

<sup>5</sup> Facultad de Contaduría y Administración, Universidad Autónoma de Coahuila, 27000, Torreón, Coahuila, México.

\* Correspondence: claudia.gomez@itcm.edu.mx

**Abstract.** In this paper, we enriched Ant Colony Optimization (ACO) with interval outranking to develop a novel multi-objective ACO optimizer to approach problems with many objective functions. This proposal is suitable if the preferences of the Decision Maker (DM) can be modeled through outranking relations. The introduced algorithm (Interval Outranking-based ACO, IO-ACO) is the first ant-colony optimizer that embeds an outranking model to bear vagueness and ill-definition of the DM's preferences. This capacity is the most differentiating feature of IO-ACO because this issue is highly relevant in practice. IO-ACO biases the search towards the Region of Interest (RoI), the privileged zone of the Pareto frontier containing the solutions that better match the DM's preferences. Two widely studied benchmarks were utilized to measure the efficiency of IO-ACO, i.e., the DTLZ and WFG test suites. Accordingly, IO-ACO was compared with four competitive multi-objective optimizers: The Indicator-based Many-Objective ACO, the Multi-objective Evolutionary Algorithm Based on Decomposition, the Reference Vector-Guided Evolutionary Algorithm using Improved Growing Neural Gas, and the Indicator-based Multi-objective Evolutionary Algorithm with Reference Point Adaptation. The numerical results show that IO-ACO approximates the RoI better than leading metaheuristics based on approximating the Pareto frontier alone.

**Keywords.** Swarm Intelligence; Many-Objective Optimization; Interval Outranking; Vagueness in the DM's preferences

---

PREPRINT OF A PAPER ACCEPTED IN *SWARM AND EVOLUTIONARY COMPUTATION*

---

## 1 Introduction

Many engineering, science, and industry problems require considering the simultaneous optimization of several conflicting objective functions. Multi-Objective Evolutionary Algorithms (MOEAs) have been applied successfully in solving problems with 2-3 objective functions; still, they often involve more objectives in real life. Most MOEAs have limitations in treating problems with four or more objectives (the so-called Many-objective Optimization Problems, MaOPs). Bechikh et al. (2017) summarized the challenges faced by the state-of-the-art MOEAs in solving MaOPs:

- 1) Many solutions in the current MOEA's population become non-dominated, weakening the selective pressure towards the true Pareto frontier.
- 2) The increasing number of dominance-resistant solutions makes it difficult to discriminate well enough among solutions; these are solutions with poor values in some objectives but with near-optimal values in

others (Ishibuchi et al., 2020).

- 3) The effectiveness of the genetic operators (crossover, mutation, and selection) is reduced.
- 4) There are difficulties in representing the known Pareto frontier because the number of points grows exponentially given a resolution level.
- 5) Visualization of solutions in a high-dimensional space is complex.
- 6) There are high computational costs derived from the estimation of diversity measures.

The above difficulties are severe in Pareto dominance-based MOEAs; yet some challenges (particularly Points 1, 3, and 6) are less demanding for decomposition-based algorithms, like MOEA/D (Zhang & Li, 2007) or non-evolutionary metaheuristics that build independent solutions. However, identifying an approximation to the Pareto frontier is not enough; the Decision Maker (DM) requires finding the best compromise, the solution from the Pareto frontier that better matches their preferences. Completing this process requires articulating the DM's preferences, which can be carried on at different stages of the decision-making process: *a priori*, *a posteriori*, or interactively.

In the *a posteriori* incorporation of preferences, the DM must choose the final solution once the metaheuristic algorithm has generated the approximated Pareto frontier. An assumption behind this strategy is that the solution set must contain the most satisfactory solutions for the DM. This privileged set of solutions is called the Region of Interest (RoI). Another implicit assumption is that the DM can identify their best solution, even in problems with many objective functions. The options to perform this selection process are the following:

- To make a heuristic selection. The difficulty of this task increases with the space dimensionality due to the cognitive limitations of the human mind (cf. Miller, 1956).
- To use a formal multi-criteria decision method, which involves an articulation of the DM's preferences.

Contrarily, the *a priori* incorporation of preferences usually requires that the DM defines the relative importance of the objectives before the search process. This importance is reflected in the weights associated with the objective functions. Then, the DM's preferences are modeled to evaluate and compare solutions through some multi-criteria decision-making techniques.

Lastly, in the interactive incorporation of preferences, the DM must express their preferences about the solutions to skew the search. This interaction is done throughout the search process; this task usually becomes time-consuming and cognitively demanding for the DM. But even so, this approach is perhaps the best option when it is appropriate to restrict the search within a particular and well-defined RoI. Methods with interactive incorporation are trendy because these algorithms exhibit the ability to 'learn' the DM's preferences, thus suggesting more preferred solutions (Tomczyk & Kadzinski, 2020). Besides, the DM learns about their problem and updates their multi-criteria preferences. The DM often feels confident about the found solutions because interactivity allows them to specify new information to redefine their preferences throughout the search process.

The *a priori* and interactive incorporations of preferences give relevant advantages over the *a posteriori* one. First, there is an increment of selective pressure towards solutions closer to the RoI, thus narrowing the search space (Branke et al., 2016) and helping to find better solutions. Second, there is also a decrement in the number of candidate solutions to be the best compromise, which reduces the DM's cognitive effort to choose the final prescription.

As a consequence of the above advantages, the interest in combining MOEAs and multi-criteria decision-making

techniques has increased in recent years. There are many proposals to incorporate preferences into the metaheuristic search processes that exist in the scientific literature. Regardless of the stage of the articulation of preferences, the following general strategies group a vast majority of the approaches:

- Expectation-based methods
- Comparison of objective functions
- Comparison of solutions
- Preference relations that replace Pareto dominance

The expectation strategy refers to goals that the DM wants to achieve for the objectives (Xin et al., 2018); it contains those works that use reference points (e.g., Qi et al., 2019; Liu et al., 2020b; Li et al., 2020; Wang et al., 2021; Abouhawwash & Deb, 2021), and desirability thresholds (e.g., Wagner & Trautmann, 2010; He et al., 2021). The comparison of objective functions includes those algorithms that use weights for the objective functions (e.g., Branke & Deb, 2005; Brockhoff et al., 2013; Wang et al., 2015), priority ranking of the objective functions (e.g., Cvetkovic & Parmee, 2002; Taboada et al., 2007; Kulturel-Konak et al., 2008), and trade-offs between the objective functions (e.g., Branke et al., 2001; Miettinen et al., 2008). The comparison of solutions has different methods, such as the ranking of solutions (e.g., Deb et al., 2010; Yuan et al., 2021), pairwise comparisons (e.g., Branke et al., 2016; Tomczyk & Kadzinski, 2020), classification of solutions (e.g., Cruz-Reyes et al., 2017; 2020), scoring (e.g., Li, 2019; Saldanha et al., 2019), and characterization of the preferred region (e.g., Dunwei et al., 2017). The last group (preference relations that replace Pareto dominance) refers to those methods that use preference relations instead of Pareto dominance (e.g., Parreiras & Vasconcelos, 2007; Fernandez et al., 2010; 2011; Jakubovski Filho et al., 2018; Balderas et al. 2019; Yi et al., 2019).

Many of the above methods should be used interactively (e.g., ranking of solutions, pairwise comparison of solutions, and classification of solutions). Other methods can be used in an *a priori* and interactive way (e.g., preference relations that replace Pareto dominance). However, interactive methods are criticized because they require transitive preferences and full comparability (French, 1993). These requirements may be fulfilled in problems with a few objectives. Still, these properties become unlikely as the number of objectives increases.

In contrast, the *a priori* incorporation of preferences does not demand a rational behavior from the DM. Still, it requires a direct preference parameter elicitation that is only viable with significant imprecision. To alleviate this drawback, Fernandez et al. (2019) proposed an interval-based outranking method, which can handle non-transitive preferences, incomparability, and veto situations, such as the multi-criteria decision methods called ELECTRE (Roy, 1990a). These properties are relevant for solving real-world problems because the preferences of many DMs are non-compensatory and non-transitive. The interval outranking method of Fernandez et al. (2019) allows incorporating imprecisions in preference parameter values when setting *a priori*. The DM feels more comfortable eliciting the preference parameter values as interval numbers than as precise values. This feature is even more critical when the DM's preferences are ill-defined (e.g., the DM is a collective entity) or when the DM is an inaccessible person (e.g., the CEO of a very important enterprise). If the DM cannot directly provide a value for a parameter, the indirect elicitation method of Fernandez et al. (2020) allows inferring all the required parameters. A variant of the interval outranking by Fernandez et al. (2019) was used by Balderas et al. (2019), incorporating *a priori* preferences in an evolutionary algorithm in the context of portfolio optimization.

As one can see in the long list of the papers above, extensive research on incorporating preferences in MOEAs has been conducted. Comparatively, few studies combine preferences with other metaheuristics. One of the most popular non-evolutionary metaheuristics is Ant Colony Optimization (ACO) (Dorigo & Stützle, 2004), which is the subject of this paper. The ACO algorithms are inspired by the collective search for food of ants, where the

ants communicate indirectly with other members of the ant colony. Communication is based on modifying the local environment by depositing a chemical called pheromone. In the foraging for food, some species use the behavior called ‘trace-trace’ and ‘follow-trace’ to find the shortest route between the nest and the food source (Grassé, 1959). The ACO algorithms take some important characteristics from actual ant colonies: there is indirect communication through traces of pheromones, shorter paths have higher pheromones, and ants prefer paths with higher amounts of pheromone. Initially, ACO’s design was for solving combinatorial optimization problems. Optimization through multi-objective ant colonies has been applied as a powerful search technique for solving complex multi-objective NP-hard problems. Consequently, ACO approaches attracted the attention of an increasing number of researchers, and many successful applications are now available (Mandal & Dehuri, 2020; Chandra Mohan & Baskaran, 2012; Dorigo & Blumb, 2005). Multi-Objective ACO (MOACO) algorithms have also been used to treat discrete optimization problems in various domains (Alaya et al., 2007; Zhou et al., 2020).

ACO and MOACO algorithms generally concern specific problems, typically discrete problems. This particularity lies in its characteristics of using specific information of the problem faced; these algorithms do not usually address general problems. The study of Falcón-Cardona and Coello Coello (2016) has the merit of proposing a way to exploit ant colonies without the need for specific knowledge of the problem but from the evaluation of the objective functions, which makes this approach independent of the problem and gives it greater generality. Falcón-Cardona and Coello Coello (2017) presented an extended version of a preliminary study (Falcón-Cardona & Coello Coello, 2016) in which the Indicator-based Multi-Objective Ant Colony Optimization algorithm for continuous search spaces (iMOACO<sub>R</sub>) was introduced. To the best of the authors’ knowledge, this was the first MOACO algorithm explicitly designed to deal with continuous MaOPs; it exhibits a competitive performance compared with MOEA/D and NSGA III (Deb & Jain, 2014). Besides, Zhao et al. (2021) introduced one of the most recent studies to address continuous multi-objective problems through MOACO.

In our view, there is nothing to prevent the *a priori* incorporation of preferences in MOACO algorithms; however, there are very few works within such an avenue of research. This paper is intended to close that research gap by addressing the incorporation of the DM’s preferences in a MOACO algorithm through the interval outranking approach (Balderas et al., 2019). Then, the proposed MOACO uses a relational system of preferences built on interval outranking combined with the algorithm of Falcón-Cardona and Coello Coello (2017) to treat continuous MaOPs. To our knowledge, no previous papers have incorporated preferences using this approach. Besides, outranking is advantageous because it confers desirable properties to handle non-compensatory preferences and veto effects. The use of intervals allows handling imprecision and uncertainty in model parameters specified directly, also dealing with poorly defined preferences.

The remainder of this paper is organized as follows. Section 2 presents a review of the literature on MOACO algorithms incorporating the DM’s preferences. Section 3 provides the background information, including a short description of the interval outranking method of Balderas et al. (2019), which slightly simplifies the proposal by Fernandez et al. (2019). Section 4 details the proposed Interval Outranking-based Ant Colony Optimization (IO-ACO) algorithm and how it incorporates the DM’s preferences. Section 5 describes the experimental design, including results, analyses, and insights. Lastly, Section 6 discusses the conclusions and provides directions for future research.

## **2 A Brief Review on Preference Incorporation in Multi-objective Ant Colony Algorithms**

ACO is a constructive method that forms solutions by adding one solution component at a time to the current partial solution. The DM’s preferences can be incorporated into the construction of solutions since the transition

rule, also called transition probabilities, can be easily redefined with preferences in mind. However, to our knowledge, only a few studies have incorporated preferences in MOACO algorithms.

Chica et al. (2011) introduced some procedures for incorporating preference information into a MOACO algorithm called Multiple Ant Colony System (MACS). These procedures use an *a priori* approach to integrate the Nissan managers' expertise by eliciting their preferences about the decision variables and the objective functions. In the decision space, the procedure concerns only solutions having the same values in the objectives. A discrimination procedure uses new relations formulated through preference measures, which consider expert-relevant requirements concerning decision variables. In the objective space, the preferences are incorporated through two alternative approaches: (a) by units of importance and (b) by setting a set of goals. In (a), experts set units of importance to the achievement of the objectives; then, the definition of Pareto dominance is modified to specify acceptable trade-offs for each pair of objectives regarding the units of importance. New objective functions replace the original ones in an aggregation of them using the units of importance. In (b), experts define goals and incorporate them into the objective functions.

Du et al. (2011) proposed a multi-objective scheduling problem in a hybrid flow shop. The two minimizing objectives considered in the proposed model are makespan and energy consumption, which are often in conflict. The Preference Vector Ant Colony System (PVACS) allowed focusing the search on the RoI instead of the entire Pareto frontier. This aim is achieved by maintaining a separate pheromone matrix for each objective and assigning a preference vector to the ants. This vector, provided by the users, represents the relative importance attached to the objectives.

Cruz et al. (2014) optimized interdependent projects portfolios. They introduced an ACO metaheuristic to incorporate preferences based on the outranking model by Fernandez et al. (2011). This model reduces problems with many objective functions to a surrogate three-objective optimization problem, which is addressed through a lexicographic approach. The proposed Non-Outranked ACO (NO-ACO) algorithm searches for optimal portfolios in synergetic conditions and can handle interactions impacting both objectives and costs. Redundancy is also considered during the construction of portfolios. Since the selective pressure towards a privileged zone of the Pareto frontier increases by incorporating preferences, a region that better matches the DM's preferences can be identified. NO-ACO achieves greater closeness to the RoI with less computational effort than other metaheuristic approaches that do not incorporate preferences.

Fernandez et al. (2015) proposed NO-ACO II, which combines NO-ACO with integer linear programming for optimizing project portfolios with interacting projects and decisions on partial support. The advantages of this approach were evidenced by a comprehensive set of computer experiments using input instances with realistic size.

Do Nascimento Ferreira et al. (2016) proposed an interactive approach to solving the Next Release Problem (NRP), employing an ACO algorithm named Interactive ACO for the Interactive NRP (iACO4iNRP). The purpose is to reach solutions that incorporate subjective aspects while optimizing other relevant metrics related to the engineering requirements. The algorithm interacts with the user, showing all their possible software requirements. They select one expectation about the next software release for each requirement: whether it should be present, or should not, or there is no preference. This preference information is aggregated in a single score, aiming to maximize customer satisfaction. The last measure is calculated as the number of expectations met by a solution divided by the total number of expectations declared by the user. After that, the preference information is used to build solutions as part of the heuristic information of iACO4iNRP.

Bastiani et al. (2015) addressed the project portfolio problem through an ACO algorithm that approximates the Pareto frontier of some proxy variables derived from the list of candidate projects ranked by experts. These proxy variables are the objective functions. This algorithm is a suitable model for project portfolio optimization with projects characterized by a priority ranking. The input information is just the ranking and the costs of the projects.

Fernandez et al. (2017) presented an extension of the study of Bastiani et al. (2015) to address project portfolio problems with scarce information about the candidate projects; unfortunately, the model had many objective functions. Thus, Fernandez et al. (2017) reduced the original many-objective problem to a surrogate model with three objective functions, which consider the DM's preferences articulated in a fuzzy outranking system. The solution of the surrogate model is approximated via an ACO algorithm named Ant-Colony Optimization for Solving portfolio problems with Ordinal information about Projects (ACO-SOP).

Khelifa and Laouar (2020) proposed an intelligent decision support system to find the best urban planning project that fits an established area. The experts recommend the weights of the urban projects, which are interactively updated according to the specifications of each urban area. The system uses a holonic approach to model large and complex urban systems through an algorithm called H-MACO (Holonic Multi-Objective Ant Colony Optimization). The algorithm follows a bottom-up process, calling an ant colony system recursively in each sub-area of holons.

By way of summary, Table 1 presents a comparative analysis of the features of IO-ACO in contrast with other ACO algorithms that incorporate the DM's preferences. As far as we know, IO-ACO is the first MOACO algorithm that incorporates preferences, modeled in a relational system based on interval outranking, to address continuous MaOPs.

Table 1. Features of IO-ACO and other MOACO algorithms that incorporate the DM's preferences

Algorithm	Problem	Type of problem	Strategy to incorporate the DM's preferences	Model of preferences
MACS (Chica et al., 2011)	Time and space assembly line balancing MACS	- Discrete - Bi-objective	(a) Comparison of objective functions (b) Expectation-based methods	(a) Weighted sum (b) Reference points
PVACS (Du et al., 2011)	Hybrid flow shop	- Discrete - Bi-objective	Comparison of objective functions	Weighted sum
NO-ACO (Cruz et al., 2014) and NO-ACO II (Fernandez et al., 2015)	Project portfolio optimization	- Discrete - Many-objective (9 objectives)	Preference relations that replace Pareto dominance	Fuzzy outranking
iACO4iNRP (Do Nascimento Ferreira et al., 2016)	Next release problem	- Discrete - Bi-objective	Comparison of solutions	Weighted sum
ACO-SOP (Fernandez et al., 2017)	Portfolio selection on a set of ordered projects	- Discrete - Many-objective (10 objectives)	Preference relations that replace Pareto dominance	Fuzzy outranking
H-MACO (Khelifa & Laouar, 2020)	Urban project planning	- Discrete - Multi-objective (4 objectives)	Comparison of objective functions	Weighted sum
IO-ACO (this paper)	Synthetic (DTLZ and WFG)	- Continuous - Many-objective (3, 5, 7 and 10 objectives)	Preference relations that replace Pareto dominance	Interval outranking

### 3. Background

This section presents the foundations of IO-ACO, our optimizer. Subsection 3.1 briefly describes  $ACO_{\mathbb{R}}$ , the baseline ACO algorithm for continuous domains. Subsection 3.2 defines the general concepts related to multi-objective optimization. Subsection 3.3 describes the basis of outranking for supporting decision making, following the perspective of the European School of Multicriteria Decision Analysis. Subsection 3.4 presents some preliminaries of interval numbers. Lastly, Subsection 3.5 describes the interval outranking model used for identifying the best-compromise solution.

### 3.1 Ant Colony Optimization for Continuous Domains

Ant Colony-based algorithms were initially designed to approach combinatorial optimization with a focus on graph problems. In general, ACO attempts to solve an optimization problem by iteratively:

- Building solutions in a probabilistic way; here, the pheromone trail represents the probability distribution used.
- Updating the pheromone trail according to the patterns in the best-evaluated solutions; this strategy is used to bias the solutions of the next-iteration towards high-quality regions in the search space.

A pivotal element for ACO algorithms is the pheromone representation, typically given in a numerical matrix denoted by  $\tau$ . The pheromone values act as a reinforcement learning model of the experience of the ants in searching.

$ACO_{\mathbb{R}}$  (Socha & Dorigo, 2008) is probably the most remarkable ACO version available to optimize single-objective problems with continuous decision variables. Socha and Dorigo (2008) represented  $\tau$  by an archive storing the best-so-far solutions. Given a problem with  $n$  decision variables, a vector  $x_l = \langle x_{l,1}, x_{l,2}, x_{l,3}, \dots, x_{l,n} \rangle$  represents a solution, and  $f(x_l)$  represents the objective function to minimize. Then,  $\tau$  stores the  $\kappa$  best-evaluated solutions, which are ascending sorted according to  $f(x_l)$ ; the structure of  $\tau$  is presented in Figure 1.

$x_1$	$x_{1,1}$	$x_{1,2}$	...	$x_{1,j}$	...	$x_{1,n}$	$f(x_1)$	$\omega_1$
$x_2$	$x_{2,1}$	$x_{2,2}$	...	$x_{2,j}$	...	$x_{2,n}$	$f(x_2)$	$\omega_2$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
$x_l$	$x_{l,1}$	$x_{l,2}$	...	$x_{l,j}$	...	$x_{l,n}$	$f(x_l)$	$\omega_l$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
$x_\kappa$	$x_{\kappa,1}$	$x_{\kappa,2}$	...	$x_{\kappa,j}$	...	$x_{\kappa,n}$	$f(x_\kappa)$	$\omega_\kappa$
	$G_1$	$G_2$	...	$G_j$	...	$G_n$		

Figure 1. Pheromone representation in  $ACO_{\mathbb{R}}$

According to Figure 1, each solution  $x_l$  has an associated weight  $\omega_l$ , which measures the quality of the solution in terms of its position in  $\tau$ . The weight of the  $l$ th solution is defined as:

$$\omega_l = \frac{1}{\zeta \cdot \kappa \sqrt{2\pi}} e^{-\varphi(l)}, \quad \text{where } \varphi(l) = \frac{[l - 1]^2}{2\zeta^2 \kappa^2}, \quad (1)$$

which essentially defines the weight to be a value of the Gaussian function with argument  $l$ , mean 1.0, and standard deviation  $\zeta \cdot \kappa$ , where  $\zeta$  is a parameter of the algorithm and  $\kappa$  is the number of the best evaluated solutions. When  $\zeta$  is small, the top-ranked solutions are strongly preferred, and when it is large, the probability becomes more uniform. The effect of this parameter on  $ACO_{\mathbb{R}}$  is to adjust the balance between the influences of the iteration-best and the best-so-far solutions.

According to Figure 1, there are  $n$  Gaussian kernels  $G_j$  ( $1 \leq j \leq n$ ), one for each decision variable. These kernels

are used to infer a probability density function, expressed as

$$G_j(x) = \sum_{l=1}^{\kappa} \omega_l g_l^j(x). \quad (2)$$

A Gaussian kernel is a weighted sum of several one-dimensional Gaussian functions  $g_l^j(x)$ , defined as:

$$g_l^j(x) = \frac{1}{s_l^j \sqrt{2\pi}} e^{-\phi_j(l)}, \quad \text{where } \phi_j(l) = \frac{(x - x_{l,j})^2}{2(s_l^j)^2}, \quad (3)$$

where  $g_l^j(x)$  defines a normal distribution with mean  $x_{l,j}$  and standard deviation  $s_l^j$ . The latter is dynamically calculated as ants construct solutions at each iteration.

An ant constructs a solution by performing  $n$  steps. At the  $j$ th step, the  $i$ th ant infers a value for the variable  $x_{i,j}$ . As mentioned earlier, there are  $n$  Gaussian kernels, each of them is composed of  $\kappa$  regular Gaussian functions. Nevertheless, at the  $j$ th step, only the resulting Gaussian kernel  $G_j$  is required. For this purpose, the weights  $\omega_l$  are computed following Equation 1. Then, the sampling is performed in two phases. Phase one consists of choosing one of the Gaussian functions that compose the Gaussian kernel. The probability  $p_l$  of choosing the  $l$ th Gaussian function is given by

$$p_l = \frac{\omega_l}{\sum_{r=1}^{\kappa} \omega_r}. \quad (4)$$

The choice of the  $l$ th Gaussian function is made once per ant and per iteration. This fact means that, during a complete iteration, the ants only use the Gaussian functions  $g_l^j(x)$  associated with the chosen solution  $x_l$  for constructing the solution incrementally in each step. This strategy allows exploiting the synergy among the decision variables.

Phase two consists of sampling the chosen Gaussian function ( $l$ ). At the  $j$ th step, the standard deviation needs to be known for the single Gaussian function  $g_l^j(x)$  chosen in phase one; consequently, only the standard deviation  $s_l^j$  is needed. Indeed, the sampled Gaussian function differs at each  $j$ th construction step, and  $s_l^j$  is dynamically calculated as follows:

$$s_l^j = \xi \sum_{r=1}^{\kappa} \frac{|x_{r,j} - x_{l,j}|}{\kappa - 1}, \quad (5)$$

which is the average distance from the chosen solution  $x_l$  to other solutions  $x_r$  in the archive and multiplied by the parameter  $\xi$  ( $1 \geq \xi > 0$ ), which is the same for all the dimensions, having an effect like the pheromone evaporation rate in ACO.  $\xi$  influences how the long-term memory is used; that is, with high values, the search is less biased towards the points of the search space that have been already explored. The  $i$ th ant randomly assigns the value of the  $j$ th decision variable following  $x_{i,j} \sim g_l^j(x)$ . The kernels  $G_j(x)$  are probability density functions used to infer values for the decision variables from the top-ranked solutions. As a result, these kernels represent the reinforcement-learning model acquired by the colony.

Several researchers have recently extended  $ACO_R$  to approach multi-objective optimization problems. For instance, Zhang et al. (2019) hybridized  $ACO_R$  with support vector regression and the chaos theory to address the high-speed train nose problem; experiments with up to four objective functions were conducted in this case. Nevertheless, the most remarkable direction to extend  $ACO_R$  is to change the structure of  $\tau$ , such that MHACO (Acciarini et al., 2020) that sorts  $\tau$  by hypervolume scores to address space trajectory optimization with two objectives. Another such example is iMOACO<sub>R</sub>, which performs this sorting considering the R2 metric (Brockhoff et al., 2012) to address the DTLZ and WFG test suites with 3–10 objective functions. Our contribution is in line with this latter strategy: IO-ACO sorts the solutions in  $\tau$  following a system of preferences based on interval outranking.

### 3.2 Foundations of Multi-Objective Optimization

According to Deb (2001), the term optimization refers to finding one or more feasible solutions that correspond to extreme values of one or more objectives (optimal solutions). It is called mono-objective optimization when problems involving a single objective function are addressed. Contrarily, if the problem has few objective functions (typically, 1–4), then it is called multi-objective optimization. Farina and Amto (2002) recognized that the performance of most MOEAs is severely deteriorated when they attempt to solve problems with more than four objectives, termed them ‘many-objective problems.’

Many real-world optimization problems involve the simultaneous treatment of multiple objectives (Coello Coello et al., 2007). The objectives, also called criteria, are often in conflict; that is, by improving one of the objectives, the others may be affected. Determining the best solution is not a trivial task; it implies that, generally, no solution is simultaneously optimal in all the objectives. Also, there may be incomparability among solutions. Contrary to mono-objective optimization —where a single solution is sought, which presents the best performance (global optimum)— in multi-objective optimization, the solution of a Multi-Objective Problem (MOP) gives rise to a set of compromise solutions. Therefore, the solution to a MOP is not unique but a set of possible solutions from which the DM must select the best compromise, the solution to implement.

Deb (2001) defines a solution of a MOP as a vector of decision variables  $x = \langle x_1, x_2, \dots, x_n \rangle$  which optimizes (maximizes or minimizes) a vector function  $f(x)$  whose elements represent the objective functions of the problem, defined as

$$f(x) = \langle f_1(x), f_2(x), f_3(x), \dots, f_m(x) \rangle \quad f_k: \mathbb{R}^n \rightarrow \mathbb{R}, \quad (6)$$

where  $n$  is the number of decision variables, and  $m$  is the number of objective functions. In real-world problems, optimization models often add constraints to Equation 6 depending on the specific domain. All solutions that satisfy those constraints make up the feasible region,  $R_F$  (Coello Coello et al., 2007).

Pareto dominance is frequently used in the context of MOPs to compare —without additional information about the DM’s preferences— a pair of solutions and determine whether one of them is better. Pareto dominance allows discriminating in two solutions ( $x$  and  $y$ ) by only comparing their vector objective functions. Emmerich and Deutz (2018) defines the Pareto dominance relation ( $\preceq$ ) as

$$x \preceq y = \{(x, y) : f_k(x) \leq f_k(y) \forall k \in \{1, 2, 3, \dots, m\} \wedge f_k(x) < f_k(y) \exists k \in \{1, 2, 3, \dots, m\}\}. \quad (7)$$

In other words,  $x \preceq y$  ( $x$  dominates  $y$ ) is held if  $x$  has values non-inferior to  $y$  in all objective functions simultaneously, and  $x$  is better than  $y$  in one objective at least. A solution  $x \in R_F$  is non dominated (Pareto optimal) if  $y \not\preceq x \forall y \in R_F$ . The Pareto set consists of the non-dominated solutions, expressed as  $PS = \{x \in R_F : y \not\preceq x \forall y \in R_F\}$ . The Pareto frontier is the image of the Pareto set, expressed as

$$PF = \{(f_1(x), f_2(x), f_3(x), \dots, f_m(x)) : x \in PS\}. \quad (8)$$

### 3.3 The Relational System of Fuzzy Preferences based on Outranking

The idea of incorporating the fuzzy outranking relations of ELECTRE into metaheuristics for many-objective optimization has been previously studied. In the domain of portfolio optimization, the pioneers of this strategy were Fernandez et al. (2010), who subsequently encouraged a wide range of studies in the last decade that exploits the properties of the outranking relations (e.g., Fernandez et al., 2011, 2017; Rivera et al., 2012; 2021; Cruz et al., 2014; Balderas et al., 2016; Gomez et al., 2018; Rangel-Valdez et al., 2020). These studies provide empirical evidence that the metaheuristic algorithms increase their selective pressure by incorporating the DM's preferences articulated through ELECTRE III. Consequently, they perform better than Pareto-based metaheuristics when MaOPs are addressed.

The basis of the original idea is the relational system of preferences described by Roy and Vanderpooten (1996). A crucial model is  $\sigma(x, y)$ , which is the fuzzy value of the proposition 'x is at least as good as y' and calculated by classical methods from the literature (e.g., Roy 1990b; Brans & Mareschal, 2005). The notion behind the fuzzy relational system of preferences proposed by Fernandez et al. (2011) is that a solution x is preferred to another y if 'x is at least as good as y' and 'y is not at least as good as x'—strictly speaking,  $\sigma(x, y)$  has a high value as  $\sigma(y, x)$  has a low value.

ELECTRE defines  $\sigma(x, y)$  considering

- the concordance index,  $c(x, y)$ , which measures the strength of the criteria coalition in favor of 'x is at least as good as y;' and
- the discordance index,  $d(x, y)$ , which measures the strength of the criteria invalidating the statement 'x is at least as good as y.'

On the one hand, in order to calculate the concordance index, it is necessary to know how the DM perceives the criteria and their values. This calculation requires the following parameters

- *Weight vector*: This represents how important each of the objectives is to the DM and is denoted by the vector  $\vec{w} = \langle w_1, w_2, w_3, \dots, w_m \rangle$ , where  $w_k > 0 \forall k \in \{1, 2, 3, \dots, m\}$ ,  $m$  is the number of objectives and  $\sum_{k=1}^m w_k = 1$ . Usually, the DM could hardly establish the value of each  $w_k$ , yet they can utilize methods such as the revised Simos' procedure (Figueira & Roy, 2002) for this task.
- *Indifference thresholds*: This indicates how small the differences—in terms of objective values—should be for the DM to consider them as marginal or not significant on a practical level. Here, the vector  $\vec{q} = \langle q_1, q_2, q_3, \dots, q_m \rangle$  represents the indifference thresholds, where  $q_k$  is the threshold for the  $k$ th criterion.

On the other hand, the discordance index is calculated based on the set of parameters known as the veto threshold. It is represented by the vector  $\vec{v} = \langle v_1, v_2, v_3, \dots, v_m \rangle$  and indicates the magnitude of the differences (in the objectives) between two alternatives to trigger a veto condition, being  $v_k > q_k \forall k \in \{1, 2, 3, \dots, m\}$ .

$c(x, y)$  is defined as the cumulative sum of the weights of the objectives for which x is non-inferior to y considering the indifference; and the discordance index  $d(x, y)$  introduces the following effect of rejection: if there is a difference against x (according to the  $k$ th criterion) that exceeds  $v_k$ , then the predicate 'x is at least as good as y' is denied, regardless of the concordance index.  $\sigma(x, y)$  combines both measures as  $\sigma(x, y) = c(x, y) \cdot d(x, y)$ .

Perhaps the strongest criticism of outranking models is the difficulty to find the precise values of the parameters ( $\lambda$ ,  $\vec{w}$ ,  $\vec{q}$  and  $\vec{v}$ ) which are unfamiliar for typical DMs, especially when the DM is a mythical entity (e.g., the public opinion), an inaccessible person, even an entity with ill-defined preferences and beliefs (e.g., a heterogeneous group). To mitigate this drawback, Fernandez et al. (2019) proposed an outranking model parametrized through interval numbers. Thus, this interval outranking simultaneously handles multi-criteria non-compensatory preferences and imperfect information in model parameters and criterion scores. In this sense, the interval outranking is more robust than the outranking models that require precise values for the parameters.

### 3.4 Preliminaries on Interval Numbers

Interval numbers are an extension of real numbers and a subset of the real line  $\mathbb{R}$  (cf. Moore, 1979). The representation of interval numbers throughout this document will be boldface italic letters, like  $\mathbf{E} = [\underline{E}, \overline{E}]$ , where  $\underline{E}$  and  $\overline{E}$  correspond to the lower and upper limits. Among several operations of interval numbers, we are interested in the addition and the order relations defined below.

Let  $\mathbf{D}$  and  $\mathbf{E}$  be two interval numbers. The addition is defined as  $\mathbf{D} + \mathbf{E} = [\underline{D} + \underline{E}, \overline{D} + \overline{E}]$ . The relations  $\geq$  and  $>$  on interval numbers are defined by using the possibility function  $P(\mathbf{E} \geq \mathbf{D})$ . This function is defined as:

$$P(\mathbf{E} \geq \mathbf{D}) = \begin{cases} 1 & \text{if } p_{ED} > 1, \\ p_{ED} & \text{if } 0 \leq p_{ED} \leq 1, \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

$$\text{where } p_{ED} = \frac{\overline{E} - \underline{D}}{(\overline{E} - \underline{E}) + (\overline{D} - \underline{D})}.$$

For the case when  $\mathbf{D}$  and  $\mathbf{E}$  are real numbers  $D$  and  $E$  (degenerate intervals),  $P(\mathbf{E} \geq \mathbf{D}) = 1$  iff  $E \geq D$ ; otherwise,  $P(\mathbf{E} \geq \mathbf{D}) = 0$ .

A realization is a real number  $e$  that lies within an interval  $\mathbf{E}$  (Fliedner & Liesio, 2016). Fernandez et al. (2019) interpret  $P(\mathbf{E} \geq \mathbf{D}) = \alpha$  as the degree of credibility that once two realizations are given from  $\mathbf{E}$  and  $\mathbf{D}$ , the realization  $d$  will be smaller than or equal to the realization  $e$ . The relations  $\mathbf{E} \geq \mathbf{D}$  and  $\mathbf{E} > \mathbf{D}$  are defined by  $P(\mathbf{E} \geq \mathbf{D}) \geq 0.5$  and  $P(\mathbf{E} \geq \mathbf{D}) > 0.5$ , respectively. These relations can also compare real numbers. The possibility functions meet the transitivity property because  $P(\mathbf{E} \geq \mathbf{D}) = \alpha_1 \geq 0.5$  and  $P(\mathbf{D} \geq \mathbf{C}) = \alpha_2 \geq 0.5 \Rightarrow P(\mathbf{E} \geq \mathbf{C}) \geq \min\{\alpha_1, \alpha_2\}$ . Consequently,  $\geq$  and  $>$  are also transitive relations on interval numbers.

### 3.5 The Best Compromise Solution in Terms of Interval Outranking

Below, we describe the interval outranking approach by Balderas et al. (2019). First, the parameters of the outranking model are extended to become interval numbers or interval vectors. Thus, the weight vector is  $\vec{w} = \langle \mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3, \dots, \mathbf{w}_m \rangle$ , where  $\mathbf{w}_k = [\underline{w}_k, \overline{w}_k] \forall k \in \{1, 2, 3, \dots, m\}$ , and  $\sum_{k=1}^m \underline{w}_k \leq 1$  and  $\sum_{k=1}^m \overline{w}_k \geq 1$ . Similarly, the indifference-threshold vector is  $\vec{q} = \langle \mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \dots, \mathbf{q}_m \rangle$ , where  $\mathbf{q}_k = [\underline{q}_k, \overline{q}_k] \forall k \in \{1, 2, 3, \dots, m\}$  and the veto-threshold vector is  $\vec{v} = \langle \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \dots, \mathbf{v}_m \rangle$ , where  $\mathbf{v}_k = [\underline{v}_k, \overline{v}_k]$  and  $\mathbf{v}_k > \mathbf{q}_k \forall k \in \{1, 2, 3, \dots, m\}$ . Lastly,

the majority threshold  $\lambda = [\underline{\lambda}, \overline{\lambda}]$ , where  $\underline{\lambda} \geq 0.5$  and  $\overline{\lambda} \leq 1$ .

Let  $x$  and  $y$  two feasible solutions, and  $f_k(x)$  and  $f_k(y)$  their  $k$ th objective functions. Let us consider the concordance coalition as the set  $C_{x,y} = \{k \in \{1,2,3, \dots, m\} : P(-q_k \leq f_k(y) - f_k(x)) \geq 0.5\}$ . The concordance coalition indicates the objectives favoring the statement ‘ $x$  is at least as good as  $y$ ’. The criteria that are not in the concordance coalition belong to the discordance coalition,  $D_{x,y} = \{k \in \{1,2,3, \dots, m\} : k \notin C_{x,y}\}$ .

Then, the concordance index  $c(x, y) = [c(x, y), \overline{c(x, y)}]$  is a cumulative sum of the weights of the objectives belonging to  $C_{x,y}$ , whose limits are defined as

$$\underline{c(x, y)} = \begin{cases} \sum_{k \in C_{x,y}} w_k & \text{if } \left( \sum_{k \in C_{x,y}} w_k + \sum_{k \in D_{x,y}} \overline{w}_k \right) \geq 1, \\ 1 - \sum_{k \in D_{x,y}} \overline{w}_k & \text{otherwise,} \end{cases} \quad (10)$$

and

$$\overline{c(x, y)} = \begin{cases} \sum_{k \in C_{x,y}} \overline{w}_k & \text{if } \left( \sum_{k \in C_{x,y}} \overline{w}_k + \sum_{k \in D_{x,y}} w_k \right) \leq 1, \\ 1 - \sum_{k \in D_{x,y}} w_k & \text{otherwise.} \end{cases} \quad (11)$$

The discordance index is calculated as

$$d(x, y) = 1 - \max_{k \in D_{x,y}} \{P(f_k(x) - f_k(y) \geq v_k)\}. \quad (12)$$

Then, the outranking function is

$$\sigma(x, y) = \min\{P(c(x, y) \geq \lambda), d(x, y)\}. \quad (13)$$

Let  $\beta$  be a threshold on the credibility of the statement ‘ $x$  is at least as good as  $y$ ’ ( $\beta \geq 0.5$ ). The binary relation  $S$  (outranking) is represented as

$$xSy = \{(x, y) : \sigma(x, y) \geq \beta\}, \quad (14)$$

and the crisp relation ‘ $x$  is preferred to  $y$ ’ is expressed as

$$xPr y = \{(x, y) : x \preceq y \vee (xSy \wedge \neg ySx)\}, \quad (15)$$

where the symbol ‘ $\preceq$ ’ stands for dominance in the Pareto sense.

From a set of feasible solutions  $\mathcal{O}$ , the following sets can be defined:  $S(\mathcal{O}, x) = \{y \in \mathcal{O} : xSy\}$ , and  $P(\mathcal{O}, x) = \{y \in \mathcal{O} : yPr x\}$ .  $S(\mathcal{O}, x)$  allows measuring the strength of solution  $x$ , and  $P(\mathcal{O}, x)$  allows measuring its weakness. The best compromise  $x^*$  from a set of solutions  $\mathcal{O}$  has the best values of strength and weakness and can be expressed as the bi-objective problem

$$x^* = \arg \min_{x \in \mathcal{O}} \{ |P(\mathcal{O}, x)|, -|S(\mathcal{O}, x)| \}, \quad (16)$$

with lexicographic priority in favor of  $|P(\mathcal{O}, x)|$ .

Pertinent remarks on this relational system are the following:

- (1) This definition of the best compromise is in compliance with the concept of Pareto dominance: the best

compromise is always a non-dominated point in the Pareto frontier, which is a relevant property derived from Equation 15. Accordingly, the RoI —the set of solutions satisfying Problem 16— is a subset of the Pareto frontier.

- (2) Problem 16 maps the original  $m$ -dimensional objective space to a bi-dimensional space. So, the preference system is scalable because it is valid regardless of the number of objectives in the problem domain ( $m$ ).

#### 4. Our Proposal: Interval Outranking-based Ant Colony Optimization

IO-ACO is a many-objective optimizer that incorporates the DM's preferences to iMOACO<sub>R</sub>, a remarkable version of ACO<sub>R</sub> for continuous MaOPs. According to Falcón-Cardona and Coello Coello (2017), iMOACO<sub>R</sub> has shown competitive results compared to algorithms that are standard to address MaOPs (namely, MOEA/D and NSGA III). This reason was the primary motivation to propose an optimization algorithm extending iMOACO<sub>R</sub>.

We changed iMOACO<sub>R</sub> to incorporate the interval outranking model. Initially, iMOACO<sub>R</sub> sorts the solutions in the pheromone structure  $\tau$  according to their R2 scores (Brockhoff et al. 2012), an indicator that measures uniformity on the distribution of solutions to establish different quality levels among Pareto-efficient solutions.

In contrast, IO-ACO ranks the Pareto-efficient solutions by domination fronts which are obtained by considering the minimization of the two objectives  $|P(\mathcal{O}, x)|$  and  $-|S(\mathcal{O}, x)|$ ; the former has lexicographic priority according to the best-compromise definition given in Problem 16. The set composed by these fronts is denoted by  $\mathcal{F} = \{\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \dots, \mathcal{F}_f\}$ , where  $\mathcal{F}_1$  contains the non-dominated solutions,  $\mathcal{F}_2$  has the solutions that are dominated by only one solution,  $\mathcal{F}_3$  those dominated by two solutions, and so forth. In general,  $\mathcal{F}_r$  contains the solutions dominated by  $r - 1$ , where  $1 \leq r \leq f$ , and  $f$  is the total number of levels. Having the solutions ranked, they are stored in  $\tau$  together with their assigned rank. The solutions in  $\mathcal{F}_1$  are the best-so-far vectors. Once  $\tau$  contains the  $\kappa$  top-ranked solutions, the mechanisms to generate new solutions can be applied as defined in ACO<sub>R</sub> (See Section 3.1).

Figure 2 represents the pheromone trail in IO-ACO. Here, there are  $\kappa$  solutions with  $n$  decision variables memorized in  $\tau$ . The solutions are sorted following the ranking given by  $\mathcal{F}$ , where,  $\mathfrak{I}(x_l)$  is the front assigned to  $x_l$ ; therefore,  $\mathfrak{I}(x_1) \leq \mathfrak{I}(x_2) \leq \dots \leq \mathfrak{I}(x_l) \leq \dots \leq \mathfrak{I}(x_\kappa)$ . Each weight  $\omega_l$  is redefined as

$$\omega_l = \frac{1}{\zeta \cdot \mathfrak{I}(x_\kappa) \sqrt{2\pi}} e^{-\varphi(l)}, \quad \text{where } \varphi(l) = \frac{[\mathfrak{I}(x_l) - 1]^2}{2\zeta^2 \mathfrak{I}(x_\kappa)^2}. \quad (17)$$

$x_1$	$x_{1,1}$	$x_{1,2}$	...	$x_{1,j}$	...	$x_{1,n}$	$\mathfrak{I}(x_1)$	$\omega_1$
$x_2$	$x_{2,1}$	$x_{2,2}$	...	$x_{2,j}$	...	$x_{2,n}$	$\mathfrak{I}(x_2)$	$\omega_2$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
$x_l$	$x_{l,1}$	$x_{l,2}$	...	$x_{l,j}$	...	$x_{l,n}$	$\mathfrak{I}(x_l)$	$\omega_l$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
$x_\kappa$	$x_{\kappa,1}$	$x_{\kappa,2}$	...	$x_{\kappa,j}$	...	$x_{\kappa,n}$	$\mathfrak{I}(x_\kappa)$	$\omega_\kappa$
	$G_1$	$G_2$	...	$G_j$	...	$G_n$		

Figure 2. Pheromone representation in IO-ACO. The components in gray are the same as in ACO<sub>R</sub>, and those in black are the adaptations for IO-ACO

Then, ACO<sub>R</sub> can be applied to construct solutions and approximate the best compromise solution.

To emphasize the distinctive characteristics of IO-ACO, Table 2 summarizes the differences between iMOACO<sub>R</sub> and our proposed algorithm.

Table 2. Differences between IO-ACO and iMOACO<sub>R</sub>

Algorithm	Preference incorporation	Optimality based on:	Criteria to sort $\tau$	$\omega_l =$	$\varphi(l) =$
IO-ACO	<i>A priori</i>	Interval Outranking	Weakness and strength (Equation 16)	$\frac{1}{\zeta \cdot \mathfrak{S}(x_\kappa) \sqrt{2\pi}} e^{-\varphi(l)}$	$\frac{[\mathfrak{S}(x_l) - 1]^2}{2\zeta^2 \mathfrak{S}(x_\kappa)^2}$
iMOACO <sub>R</sub>	<i>A posteriori</i>	Pareto Dominance	R2 scores	$\frac{1}{\zeta \cdot \kappa \sqrt{2\pi}} e^{-\varphi(l)}$	$\frac{[l-1]^2}{2\zeta^2 \kappa^2}$

Furthermore, Algorithm 1 presents an outline of IO-ACO. First,  $\kappa$  solutions are generated at random (Line 1) and normalized (Line 2) following the approach of Hernández Gómez and Coello Coello (2015) using  $\alpha = 0.5$  and  $\epsilon = 0.001$ . Afterward, solutions in  $\tau$  are ranked and sorted according to the domination front they belong to.

---

**Algorithm 1.** The Interval Outranking-based Ant Colony Optimization algorithm

---

**Input:**  $\lambda, \vec{w}, \vec{v}, \vec{q}, \zeta$  and  $\xi$

**Output:** An approximation of the best compromise solution ( $\mathcal{F}_1$ )

1. Randomly initialize the pheromone trail ( $\tau$ )
2. `Normalize`( $\tau$ )
3. Rank solutions in  $\tau$
4.  $t \leftarrow 0$
5. **while**  $t < iter_{\max}$  **do**
6.     **for each** ant  $a \in A$
7.         Generate a new solution based on  $\tau$
8.     **end for**
9.      $\mathcal{O} \leftarrow A \cup \tau$
10.     `Normalize`( $\mathcal{O}$ )
11.     Rank solutions in  $\mathcal{O}$
12.      $\tau \leftarrow \emptyset$
13.     Copy into  $\tau$  the first  $\kappa$  elements of  $\mathcal{O}$
14.      $t \leftarrow t + 1$

➤  $A$  is the colony, which has  $\kappa$  ants

**15. end while**

**16. return**  $\tau$

---

Lines 5–15 contain the main loop of the algorithm. The ants construct solutions (Lines 6–8) following the directions provided in Section 3.1. The previous solutions ( $\tau$ ) and the new ones ( $A$ ) are merged into  $\mathcal{O}$  (Line 9). Then, the solutions in  $\mathcal{O}$  are normalized, ranked, and sorted accordingly (Lines 10–11). The  $\kappa$  solutions that better match the DM's preferences are set in  $\tau$  to influence the ants during the next iteration (Lines 12–14). When the algorithm finishes, IO-ACO provides the best compromise solution in  $\tau$  (Line 16).

In terms of computational complexity, the most costly operation of Algorithm 1 is the assessment of the solutions to identify the best compromise (Line 11). The relation  $xSy$  may be computed through  $O(n)$  operations (see Equations 10–14), and the calculation of both objective functions of Problem 16 requires determining this relation

for all pairs of solutions in  $\mathcal{O}$ . Ergo, the complexity of Algorithm 1 is  $O(|\mathcal{O}|^2 m)$ . In this paper,  $|\mathcal{O}| = 2\kappa$  because  $\kappa$  is interchangeably used as the number of ants and the size of the pheromone archive  $\tau$ .

## 5. Experimental Results

We implemented IO-ACO using C under Linux (Ubuntu 18) on a computer with an Intel Core i7-6700 3.4 GHz with 16GB of RAM. All runs of IO-ACO reported here were conducted in that computer setting. The parameter values of IO-ACO are  $\zeta = 0.1$  and  $\xi = 0.5$ ; a particular case is the number of ants ( $\kappa$ ), which depends on the dimensionality of the problem ( $m$ ):  $\kappa = 120$  for  $m = 3$ ,  $\kappa = 126$  for  $m = 5$ ,  $\kappa = 84$  for  $m = 7$ , and  $\kappa = 220$  for  $m = 10$ . This parameter setting is suggested by Falcón-Cardona and Coello Coello (2017).

Because IO-ACO considers the DM’s preferences, 10 outranking parameter settings representing different DMs were run 30 times (consequently, IO-ACO was run 300 times for each problem tested in this section). The reference algorithms —iMOACO<sub>R</sub>, MOEA/D, RVEA-iGNG (Liu et al., 2020a), and AR-MOEA (Tian et al., 2018)— were also run 300 times. To provide comparisons on an equal footing, these algorithms were set to stop in the iteration just before exceeding 50,000 evaluations of the vector objective function. In this paper, the adjective ‘significant’ means that a Wilcoxon rank-sum test (a.k.a. Mann-Whitney  $U$  test) with a 0.95-confidence level validates the research hypothesis. Moreover, all tests for statistical significance were performed using the STAC platform<sup>†</sup>: Statistical Tests for Algorithms Comparison (Rodríguez-Fdez et al., 2015).

The remaining of this section is structured as follows: Subsection 5.1 describes the two benchmarks of synthetic problems utilized to validate the efficiency of IO-ACO; Subsection 5.2 introduces the indicators used to measure performance; Subsection 5.3 presents the results about the effect of enriching iMOACO<sub>R</sub> with preference relations based on interval outranking; Subsection 5.4 compares the results of IO-ACO with some evolutionary algorithms that are relevant in the scientific literature; lastly, Subsection 5.5 presents a rank (through the Borda score) of all the metaheuristic algorithms run in this section.

### 5.1 Benchmark Problems

DTLZ (Deb et al., 2002) and WFG (Huband et al., 2005) have become the standard test suites used to assess the performance of multi-objective algorithms. These continuous problems are scalable with respect to the number of objective functions and decision variables; additionally, they offer Pareto frontiers with a wide range of properties: concavity, convexity, multi-frontality, linearity, bias, connectivity, degeneration, and separability.

Table 3. Parameters for the standard problems used in the experiments

Problems	Number of objectives ( $m$ )	Position ( $k$ )	Number of decision variables ( $n$ )
DTLZ1	{3, 5, 7, 10}	5	$m + k - 1$
DTLZ2–DTLZ6	{3, 5, 7, 10}	10	$m + k - 1$
DTLZ7	{3, 5, 7, 10}	20	$m + k - 1$
DTLZ8, DTLZ9	{3, 5, 7, 10}	$m-1$	$10m$
WFG1–WFG9	3	$2(m-1)$	24
WFG1–WFG9	5	$2(m-1)$	47
WFG1–WFG9	7	$2(m-1)$	70
WFG1–WFG9	10	$2(m-1)$	105

In this paper, we have run IO-ACO on the nine problems for both the DTLZ and the WFG test suites, named

<sup>†</sup> Available at <http://tec.citius.usc.es/stac/>

DTLZ1–DTLZ9 and WFG1–WFG9. Also, we explored each problem by varying the number of objectives, considering 3, 5, 7, and 10 objective functions. Table 3 summarizes the settings for the standard problems, including the number of decision variables and the position-related variables.

## 5.2 Performance Assessment

Unlike most multi-objective metaheuristics, IO-ACO is not intended to approximate uniformly distributed samples of the Pareto frontier. Instead, IO-ACO searches for the solutions that meet the conditions to be the best compromise solution, according to the preference relations based on interval outranking. Thus, the RoI is a privileged zone of the Pareto frontier. Hence, most current metrics (e.g., spread, spacing, and hypervolume) are not adequate to assess its performance because none of them can be directly applicable when only a partial Pareto frontier is considered (Li et al., 2017).

Alternatively, IO-ACO would show competitiveness if it generates solutions that are close enough to the true RoI. With this aim in mind, we approximated a 100000-point sample of the Pareto frontier for each problem and applied the outranking model of Section 3.5 to identify the RoI. The distance to this approximated RoI (named A-RoI hereon) is used to measure the quality of the solutions provided by IO-ACO in every single run; particularly, we consider the Euclidean distance and the Chebyshev distance. In line with this notion, given the state of the latest set of solutions  $X$  of an algorithm, denoted as  $X^*$ , and the A-RoI of a particular DM for a specific problem, the following four indicators are utilized:

- Minimum Euclidian distance. This indicator is the Euclidean distance between the closest solution from  $X^*$  to the A-RoI.
- Average Euclidian distance. It is the average Euclidean distance among the solutions from  $X^*$  to those of the A-RoI.
- Minimum Chebyshev distance. This indicator is the Chebyshev distance between the closest solution from  $X^*$  to the A-RoI.
- Average Chebyshev distance. It is the average Chebyshev distance among the solutions from  $X^*$  to those of the A-RoI.

These four indicators measure the similarity between the solutions in  $X^*$  and those in the RoI. In general, the Euclidean indicators would be applicable in decision problems whose DM has a compensatory perspective. In contrast, the Chebyshev-based indicators become relevant if the DM's perspective is non-compensatory (that is, gains in some objectives do not justify losses in others). Furthermore, the minimum distances measure the performance of the algorithm considering the best solution alone, and the averages measure the overall trend of the solutions. The minimum and average distances are complementary indicators for the same type of distance.

## 5.3 On the Impact of Incorporating the Preference Relations based on Interval Outranking

Let's consider IO-ACO as an extension of  $iMOACO_{\mathbb{R}}$ , whose contribution is to incorporate the relational system of interval outranking preferences described in Subsection 3.5. The objective of this section is to provide experimental evidence of the resulting enhancement.

Table 4 compares the results obtained by IO-ACO and  $iMOACO_{\mathbb{R}}^{\ddagger}$  on the standard problems. The parameter setting of  $iMOACO_{\mathbb{R}}$  is that suggested by Falcón Cardona and Coello Coello (2017). Here, the first and second

---

<sup>‡</sup> Source code taken from: <http://computacion.cs.cinvestav.mx/~jfalcon/iMOACOR/imoacor.html>

columns identify the test suite and the dimensionality of the problems. The third column presents the list of problems (identified by numbers) in which iMOACO<sub>R</sub> is significantly outperformed by IO-ACO in terms of any of the four indicators (see Section 5.2). In contrast, the fourth column lists the problems in which iMOACO<sub>R</sub> significantly outperformed IO-ACO. The fifth column specifies the indicator being considered.

Table 4. Comparison in performance between IO-ACO and iMOACO<sub>R</sub>

Benchmark	Number of objectives	Problems in which IO-ACO		Indicator	
		(a) outperformed iMOACO <sub>R</sub>	(b) is outperformed by iMOACO <sub>R</sub>		
DTLZ	3	1, 3, 4, 5, 6, 8, 9	7	Min. Chebyshev	
	5	1, 2, 3, 4, 5, 6, 8	9		
	7	1, 2, 3, 4, 5, 6, 8	9		
	10	1, 2, 3, 4, 5, 6, 7, 9			
WFG	3	1, 3, 4, 6, 7, 8, 9	2		
	5	1, 2, 3, 7, 9	5, 6, 8		
	7	1, 2, 4, 5, 8, 9	6, 7		
	10	1, 3, 5, 6, 7, 9	2, 8		
Counting of problems		53	11		
DTLZ	3	1, 3, 4, 5, 8, 9	2, 7		Avg. Chebyshev
	5	1, 3, 5, 6, 8	4, 9		
	7	1, 2, 3, 5, 6, 8			
	10	1, 3, 5, 6, 7, 9	4		
WFG	3	1, 3, 4, 6, 7, 8	2, 5, 9		
	5	1, 3, 4, 5, 9	2, 6, 7, 8		
	7	1, 2, 3, 4, 5, 7, 8, 9			
	10	1, 3, 4, 6, 7, 8, 9	2, 5		
Counting of problems		48	14		
DTLZ	3	1, 3, 4, 5, 6, 7, 8, 9	2	Min. Euclidean	
	5	1, 2, 3, 4, 5, 6, 7	8		
	7	1, 2, 3, 4, 5, 6, 7, 8	9		
	10	1, 2, 3, 4, 5, 6, 9	7		
WFG	3	1, 3, 4, 6, 8, 9	2, 5		
	5	2, 4, 5, 9	6, 7, 8		
	7	2, 3, 4, 8, 9	5, 6, 7		
	10	1, 3, 5, 6, 7, 9	2, 8		
Counting of problems		51	14		
DTLZ	3	1, 3, 4, 5, 6, 7, 8, 9			Avg. Euclidean
	5	1, 2, 3, 4, 5, 6, 7	8, 9		
	7	1, 2, 3, 4, 5, 6, 7, 8			
	10	1, 2, 3, 4, 5, 6, 9			
WFG	3	1, 4, 6, 7, 8	2, 3, 9		
	5	1, 2, 4, 7, 9	5, 6, 8		
	7	2, 3, 4, 7, 8, 9	5		
	10	1, 3, 6, 7, 9	4, 5, 8		
Counting of problems		51	12		

As shown in Table 4, IO-ACO outperformed iMOACO<sub>R</sub> in most of the problems in both test suites. Considering

72 problems —18 standard problems (DTLZ1–9, and WFG1–9) with four levels of objectives (3, 5, 7, and 10)—IO-ACO obtained better results in:

- 53 problems considering the minimum Chebyshev distance,
- 48 problems considering the average Chebyshev distance,
- 51 problems considering the minimum Euclidean distance, and
- 51 problems considering the average Euclidean distance.

Table 5. Average runtimes (ms) of IO-ACO and iMOACO<sub>R</sub>

Problem	Number of objectives	iMOACO <sub>R</sub>	IO-ACO	Problem	Number of objectives	iMOACO <sub>R</sub>	IO-ACO
DTLZ1	3	7713	7053	WFG1	3	12470	12428
	5	14973	12752		5	21974	20060
	7	17544	16463		7	26658	21759
	10	27583	25844		10	42178	29425
DTLZ2	3	8223	7715	WFG2	3	12383	11906
	5	17441	14264		5	18063	17024
	7	21317	15180		7	31906	31766
	10	33425	24925		10	39737	28818
DTLZ3	3	8374	8030	WFG3	3	12176	12103
	5	18402	15669		5	20260	18500
	7	22289	19635		7	26722	24575
	10	27313	17750		10	42689	34846
DTLZ4	3	8333	8115	WFG4	3	11913	11626
	5	16165	12966		5	20872	18943
	7	19981	19946		7	29289	24254
	10	32864	20740		10	38647	28027
DTLZ5	3	9719	9211	WFG5	3	13334	13140
	5	19310	16019		5	18429	16816
	7	28039	20943		7	31858	26800
	10	31754	26361		10	43437	31254
DTLZ6	3	11982	10971	WFG6	3	13695	13450
	5	18218	16598		5	17134	16736
	7	25296	18846		7	30083	26330
	10	38671	36017		10	46511	28015
DTLZ7	3	11370	11325	WFG7	3	13795	13248
	5	21334	19837		5	17321	15761
	7	25842	19052		7	29887	27701
	10	39055	25448		10	40666	33364
DTLZ8	3	12234	11067	WFG8	3	12048	11993
	5	24278	23054		5	18914	17620
	7	31584	27811		7	31390	28457
	10	48608	37172		10	44597	42769
DTLZ9	3	12389	12147	WFG9	3	11737	11242
	5	23384	21721		5	22509	20536
	7	32745	28222		7	29998	25349
	10	45662	41393		10	45932	31158

These differences were statistically significant. Regardless of the number of objectives, IO-ACO offered solutions

that better match the DM’s preferences, even assuming an effective *a posteriori* multi-criteria analysis on the solutions by iMOACO<sub>R</sub>. According to the results, the incorporation of preferences had a higher impact on DTLZ than on WFG. These results are clear evidence of the effects on the performance when the DM’s preferences are incorporated, increasing the capability to reach the RoI.

Table 5 presents the average time IO-ACO and iMOACO<sub>R</sub> took, measured in milliseconds. Here, the first and fifth columns indicate the problem, the second and sixth columns indicate the dimensionality, the third and seventh columns present the average time taken by iMOACO<sub>R</sub>, and the fourth and eighth columns show the average time taken by IO-ACO. Columns 1–4 present the data on the DTLZ suite, while columns 5–8 present the data on the WFG suite.

According to Table 5, IO-ACO took only 87% of the time took by iMOACO<sub>R</sub>, even though: (a) both algorithms have the same time complexity in terms of big-O notation, (b) they are programmed in C, and (c) they were run in the same computer using the same parameter settings. The difference may be attributed to the relational system of preferences based on interval outranking. On the one hand, *a posteriori* metaheuristics have to search for large sets of Pareto-efficient solutions. On the other hand, *a priori* metaheuristics search for the best compromise solution, which belongs to a relatively short set of points (the RoI). Consequently, the number of solutions that meet optimality conditions is much lower in IO-ACO than in iMOACO<sub>R</sub>; ergo, the selective pressure increases. Let’s keep in mind that each pair of these solutions is compared in each iteration. Additionally, the calculation of the R2 metric is no longer required.

#### 5.4 Comparison with Evolutionary Approaches

This section aims to validate the efficiency of our proposal in comparison with other relevant metaheuristics in the literature. We have considered the following three evolutionary algorithms: the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D), the Reference Vector-Guided Evolutionary Algorithm using Improved Growing Neural Gas<sup>§</sup> (RVGEA-iGNG), and the Indicator-based Multi-Objective Evolutionary Algorithm with Reference Point Adaptation<sup>\*\*</sup> (AR-MOEA). The parameter settings for these evolutionary algorithms were taken as suggested in their sources (cf. Zhang & Li, 2007; Liu et al., 2020a; Tian et al., 2018).

Table 6 compares the results obtained by IO-ACO, MOEA/D, RVGEA-iGNG, and AR-MOEA. The first and second columns present the benchmark and the dimensionality of the problems; columns 3–5 list the problems with a significant difference in favor of IO-ACO, and columns 6–8 list the problems with a significant difference in favor of the evolutionary approaches; lastly, the ninth column shows the indicator being considered.

The results of Table 6 can be summarized as follows:

- Considering the minimum Chebyshev distance,
  - IO-ACO obtained better results than MOEA/D in 37 problems, than RVEA-iGNG in 42 problems, and than AR-MOEA in 48 problems.
  - IO-ACO was outperformed by MOEA/D in 31 problems, by RVEA-iGNG in 19 problems, and by AR-MOEA in 13 problems.
  - IO-ACO obtained results similar to MOEA/D in 4 problems, to RVEA-iGNG in 11 problems, and to AR-MOEA in 11 problems.

<sup>§</sup> Source code taken from <https://github.com/BIMK/PlatEMO/tree/master/PlatEMO/Algorithms/Multi-objective%20optimization/RVEA-iGNG>

<sup>\*\*</sup> Source code taken from <https://github.com/BIMK/PlatEMO/tree/master/PlatEMO/Algorithms/Multi-objective%20optimization/AR-MOEA>

Table 6. Comparison of IO-ACO with evolutionary approaches

Benchmark	Number of objectives	Problems in which IO-ACO outperformed:			Problems in which IO-ACO is outperformed by:			Indicator	
		(a) MOEA/D	(b) RVEA-iGNG	(c) AR-MOEA	(a) MOEA/D	(b) RVEA-iGNG	(c) AR-MOEA		
DTLZ	3	4, 5, 6, 9	2, 5, 6, 8, 9	2, 3, 4, 5, 6, 8, 9	1, 2, 3, 7, 8	3, 7	1, 7	Min. Chebyshev	
	5	2, 6, 8, 9	2, 4, 5, 6, 8, 9	2, 4, 5, 6, 8, 9	1, 3, 5, 7	3, 7	1, 7		
	7	2, 4, 7, 9	2, 4, 7, 8, 9	1, 2, 3, 4, 9	1, 3, 5, 6, 8	1, 5, 6	6, 8		
	10	2, 4, 7	2, 3, 4, 7, 8	1, 2, 3, 4, 7, 8	1, 3, 5, 9	5, 9	5, 9		
WFG	3	2, 3, 6, 7, 9	3, 6, 7, 8, 9	3, 5, 6, 7, 9	1, 4, 5, 8	1, 2, 5	1, 8		
	5	1, 2, 3, 4, 6, 7	1, 2, 3, 4, 8	1, 2, 3, 4, 5, 7, 8, 9	5, 8, 9	5, 9	6		
	7	1, 2, 3, 4, 6, 8	1, 2, 4, 6, 8	1, 3, 4, 7, 8	5, 7	3, 5, 7			
	10	1, 3, 4, 7, 9	1, 3, 4, 7, 8, 9	1, 3, 4, 5, 6, 7	2, 5, 6, 8	2, 5	2		
Counting of problems		37	42	48	31	19	13		
DTLZ	3	2, 4, 5, 6, 7, 9	2, 4, 5, 7, 8, 9	2, 3, 4, 5, 8, 9	1, 3, 8	1, 3	1, 7		Avg. Chebyshev
	5	1, 2, 3, 4, 6, 8, 9	1, 3, 4, 6, 8, 9	1, 2, 3, 4, 5, 6, 8, 9	5, 7	7			
	7	1, 2, 4, 5, 7, 9	1, 2, 3, 4, 7, 9	1, 2, 3, 4, 5, 9	3, 6, 8	5, 6	6, 8		
	10	2, 4, 5, 8	1, 2, 3, 5, 8	2, 3, 4, 5, 7	1, 3, 9	4	1		
WFG	3	2, 3, 6, 7	3, 6, 7, 8	3, 6, 7	1, 4, 5, 9	1, 2, 5, 9	1, 4, 5, 9		
	5	1, 2, 3, 4, 6, 7, 8	2, 3, 4, 6, 8	1, 2, 3, 4, 5, 9	5, 9	5, 9	6		
	7	1, 3, 6, 8	1, 2, 3, 6, 8	1, 3, 5, 6, 8	2, 4, 5, 7	4, 5, 7	4		
	10	1, 2, 3, 7, 8, 9	1, 2, 6, 8	1, 4, 6, 7, 8	4, 5, 6	3, 4, 5, 9	3, 5, 9		
Counting of problems		44	41	44	24	20	14		
DTLZ	3	4, 6, 7, 8, 9	5, 7, 9	3, 4, 5, 6, 8, 9	1, 2, 3	1, 2, 3	1	Min. Euclidean	
	5	2, 4, 6, 7, 9	2, 4, 5, 6, 8, 9	2, 4, 5, 6, 8, 9	1, 3, 5, 8	3	1, 3		
	7	2, 4, 9	2, 3, 4, 8	1, 2, 3, 4, 9	1, 3, 5, 6, 8	1, 5, 6	6, 8		
	10	2, 4, 7	2, 3, 4, 7, 9	1, 3, 4, 7, 9	1, 3, 5, 9	1, 5	5		
WFG	3	2, 3, 4, 6, 7, 9	3, 6, 7, 8, 9	3, 5, 6, 7, 9	1, 5, 8	1	2, 4, 8		
	5	1, 2, 4, 6, 7	1, 2, 4	1, 2, 3, 9	3, 8, 9	8, 9			
	7	1, 2, 3, 4, 6, 8	2, 4, 6, 8	1, 2, 3, 4, 6	5, 7	5, 7	8		
	10	1, 2, 3, 4, 5, 7, 8, 9	1, 2, 3, 4, 6, 7, 8, 9	1, 3, 4, 6, 7, 9	6		2, 8		
Counting of problems		41	38	42	25	14	12		
DTLZ	3	2, 4, 5, 6, 7, 8, 9	2, 4, 5, 7, 8, 9	2, 3, 4, 5, 6, 8, 9	1, 3	3, 6	1, 7		Avg. Euclidean
	5	1, 2, 3, 4, 6, 7, 8, 9	1, 3, 4, 5, 6, 8, 9	1, 2, 3, 4, 6, 7, 8, 9					
	7	1, 2, 4, 5, 7	2, 4, 7	1, 2, 4, 5, 7, 9	3, 6, 8	1, 5, 6	6, 8		
	10	2, 4, 5, 8	2, 3, 4, 5, 8, 9	1, 2, 3, 5, 6, 8, 9	1, 3, 6, 9	6			
WFG	3	2, 4, 6, 7	3, 4, 5, 6, 7	3, 5, 6, 7	1, 5, 8, 9	1, 9	2, 8, 9		
	5	1, 3, 4, 7	2, 3, 4	1, 2, 3, 4, 9	5, 6, 8, 9	1, 5, 6, 7, 8, 9	5, 6, 7		
	7	1, 2, 3, 6, 8	2, 6, 8	1, 2, 3, 6, 7	4, 5, 7	3, 4, 5, 7	5		
	10	1, 2, 3, 9	1, 2, 6, 9	3, 6, 7	4, 5, 6, 7	3, 4, 5, 8	2, 5, 8, 9		
Counting of problems		41	37	45	24	22	15		

- Considering the average Chebyshev distance,
  - IO-ACO obtained better results than MOEA/D in 44 problems, than RVEA-iGNG in 41 problems, and than AR-MOEA in 44 problems.
  - IO-ACO was outperformed by MOEA/D in 24 problems, by RVEA-iGNG in 20 problems, and by AR-MOEA in 14 problems.
  - IO-ACO obtained results similar to MOEA/D in 4 problems, to RVEA-iGNG in 11 problems, and

to AR-MOEA in 14 problems.

- Considering the minimum Euclidean distance,
  - IO-ACO obtained better results than MOEA/D in 41 problems, than RVEA-iGNG in 38 problems, and than AR-MOEA in 42 problems.
  - IO-ACO was outperformed by MOEA/D in 25 problems, by RVEA-iGNG in 14 problems, and by AR-MOEA in 12 problems.
  - IO-ACO obtained results similar to MOEA/D in 6 problems, to RVEA-iGNG in 20 problems, and to AR-MOEA in 18 problems.
- Considering the average Euclidean distance,
  - IO-ACO obtained better results than MOEA/D in 41 problems, than RVEA-iGNG in 37 problems, and than AR-MOEA in 45 problems.
  - IO-ACO was outperformed by MOEA/D in 24 problems, by RVEA-iGNG in 22 problems, and by AR-MOEA in 15 problems.
  - IO-ACO obtained results similar to MOEA/D in 7 problems, to RVEA-iGNG in 13 problems, and to AR-MOEA in 12 problems.

According to Table 6, IO-ACO provides better solutions than these evolutionary approaches on a regular basis. The effectiveness of our algorithm depends on the problem, the distance indicator, and the number of objective functions. Despite this fact, we observed the following consistent patterns regardless of the selected indicator:

- With seven objectives functions, none of these evolutionary approaches outperformed IO-ACO in DTLZ2, DTLZ4, DTLZ7, DTLZ9, WFG1, and WFG6.
- With ten objectives functions, none of these evolutionary approaches outperformed IO-ACO in DTLZ2, DTLZ7, DTLZ8, and WFG1.

DTLZ2 and DTLZ4 are multi-frontal; additionally, these problems are Pareto many-to-one. Their objectives are non-separable, and the geometry of the Pareto frontier is concave. Remarkably, the Pareto optimal front of DTLZ4 is biased. DTLZ7 is singularly challenging because the Pareto frontier is disconnected and has mixed concave/convex regions, and the fitness landscape is one-to-one (Huband et al., 2006). Unlike the aforementioned DTLZ problems, DTLZ8 and DTLZ9 are the only problems that present side constraints in this suite; their Pareto fronts are partially degenerate, and their results are difficult to interpret in high dimensions (Meneghini et al., 2020).

On the other hand, the WFG suite is also challenging because these problems are many-to-one and have no extremal nor medial parameters. Regarding geometry, the Pareto front of WFG1 is convex with flat regions in the objective space; and the Pareto front of WFG6 is concave (Huband et al., 2006; Meneghini et al., 2020).

Although the previous insights are helpful and provide evidence of the advantages of IO-ACO, in practice, it would be highly demanding to identify the properties of the Pareto frontier for a given real-world problem. In the following subsection, we suggest how to select the best metaheuristics utilizing the well-known Borda score.

### *5.5 Rank of the Tested Metaheuristic Algorithms*

In this section, we rank the effectiveness of the five algorithms on both test suites simultaneously. For each of the 72 problems, the algorithms are sorted according to the conducted tests for statistical significance and post-hoc Holm-Bonferroni analysis with  $\alpha = 0.05$ . So, for each problem, the best algorithm obtains position 1, and the worst gets position 5 (in case of a draw, the position would be averaged). Then, the Borda score is calculated by

the sum of such positions over every single problem. Consequently, the Borda sum would provide a general ranking of the algorithms according to their average performance.

Table 7 presents the Borda score of IO-ACO, iMOACO<sub>R</sub>, MOEA/D, RVEA-iGNG, and AR-MOEA on the standard benchmarks.

Table 7. Comparison among IO-ACO, iMOACO<sub>R</sub>, MOEA/D, RVEA-iGNG and AR-MOEA

Indicator	Number of objectives	The Borda score of				
		(a) IO-ACO	(b) iMOACO <sub>R</sub>	(c) MOEA/D	(d) RVEA-iGNG	(e) AR-MOEA
Min. Chebyshev	3	42.0	61.0	51.0	56.5	59.5
	5	39.0	55.5	55.5	58.5	61.5
	7	39.5	61.0	54.5	58.0	57.0
	10	41.0	62.0	50.0	52.5	62.5
Avg. Chebyshev	3	46.5	60.0	54.0	52.5	57.0
	5	37.0	46.5	65.0	58.5	63.0
	7	41.0	60.0	49.0	60.0	60.0
	10	43.0	58.0	57.5	56.5	58.0
Min. Euclidean	3	41.0	59.0	59.0	52.0	59.0
	5	41.5	56.0	56.0	55.0	61.5
	7	44.0	53.5	58.5	49.0	65.0
	10	34.0	58.0	60.5	56.5	58.0
Avg. Euclidean	3	42.0	60.5	56.5	54.0	57.0
	5	38.0	59.0	61.5	50.5	61.0
	7	43.5	61.0	60.0	45.0	60.5
	10	42.5	60.5	58.0	50.5	55.5
<b>Summary grouping the results</b>						
Sum for Min. Chebyshev		161.5	239.5	211.0	225.5	240.5
Sum for Avg. Chebyshev		167.5	224.5	225.5	227.5	238.0
Sum for Min. Euclidean		160.5	226.5	234.0	212.5	243.5
Sum for Avg. Euclidean		166.0	241.0	236.0	200.0	234.0
Sum for the 3-objective problems		171.5	240.5	220.5	215.0	232.5
Sum for the 5-objective problems		155.5	217.0	238.0	222.5	247.0
Sum for the 7-objective problems		168.0	235.5	222.0	212.0	242.5
Sum for the 10-objective problems		160.5	238.5	226.0	216.0	234.0

Let us assume that the number of objectives and the indicator of interest are the minimum information the DM knows about the problem. For specific combinations of these two entries, we recommend that the DM follows the priority implicitly suggested by the Borda scores in Table 7 (rows 1–16). Although IO-ACO was always the algorithm in the top position in the ranking, we recognize that there were problems where IO-ACO was outperformed by other algorithms (see Table 6). If the DM is not confident of the solution offered by IO-ACO alone, they could run some more algorithms following the ranking.

If the DM is not certain about the indicator, we recommend following the ranking according to the number of objectives:

- For problems with 3 objectives: IO-ACO, RVEA-iGNG, MOEA/D, AR-MOEA, and iMOACO<sub>R</sub>.
- For problems with 5 objectives: IO-ACO, iMOACO<sub>R</sub>, RVEA-iGNG, MOEA/D, and AR-MOEA.
- For problems with 7 objectives: IO-ACO, RVEA-iGNG, MOEA/D, iMOACO<sub>R</sub>, and AR-MOEA.

- For problems with 10 objectives: IO-ACO, RVEA-iGNG, MOEA/D, AR-MOEA, and iMOACO<sub>R</sub>.

Lastly, according to the Borda scores, IO-ACO performed especially well in problems with ten objective functions and when the minimum Euclidean distance was the indicator of interest.

## 6. Conclusions and Directions for Future Research

This paper has proposed a metaheuristic algorithm to address many-objective optimization problems. One of the motivations was that DMs are often only interested in a partial region of the Pareto frontier (the RoI). Solutions outside this RoI are often noisy to the DM during the decision analysis. This difficulty is treated through IO-ACO, an ACO algorithm enriched with the preferences of the DM articulated in a relational system of preferences based on interval outranking. This preference model was considered for sorting the solutions in the pheromone trail to bias the search towards the best compromise. As far as we know, IO-ACO is the first ant colony-based metaheuristic using the interval outranking relations to get the edge in addressing continuous optimization problems with many objective functions.

Admittedly, the primary construct threat of methods based on outranking is the possibility that the values of the model parameters do not actually reflect the DM's preferences. Although the use of intervals does not eliminate the need for inferring the parameters of the outranking model (i.e., weights, veto and indifference thresholds,  $\beta$ , and  $\lambda$ ), the DM may feel more confident setting intervals than precise values for these parameters. This feature allows mitigating the consequences of this threat even in problems with vagueness and imprecision in the DM's preferences.

The convergence of IO-ACO towards the RoI was validated by measuring the distance from the Approximated RoI (A-RoI) to the solution set generated by IO-ACO; four indicators—based on the Euclidean distance and the Chebyshev distance—were used to determine closeness.

Like all metaheuristics, the main internal threat of IO-ACO is its stochastic nature. Therefore, IO-ACO was run 300 times, and the quality of its solutions was validated through tests for statistical significance with four competitive metaheuristic algorithms (expressly, iMOACO<sub>R</sub>, MOEA/D, RVEA-iGNG, and AR-MOEA). IO-ACO often offered solutions closer to the RoI in both the DTLZ and WFG test suites. These results provide evidence that IO-ACO increased selective pressure compared to *a posteriori* multi-objective metaheuristics: it focused the computational effort to approximate the RoI instead of the entire Pareto frontier. Consequently, our algorithm offered solutions with better convergence towards the RoI.

The 18 test problems also allow foreseeing possible external threats because they present Pareto frontiers with a wide range of geometries (separable, non-separable, unimodal, multimodal, deceptive, convex, linear, concave, disconnected, and degenerate). These properties are likely to be present in challenging real-world applications. Still, we admit that there were conditions (defined by the number of objectives, the indicator of interest, and the properties of the problem) where other algorithms outperformed IO-ACO.

Overall, we suggest using IO-ACO under the presence of many objective functions because its best performance was reached in problems with ten objectives when the Euclidean indicators were taken. Additionally, our algorithm was particularly competitive in a problem with a Pareto frontier whose geometry is considered challenging: the Pareto frontier of DTLZ7 is disconnected and has mixed concave/convex regions.

Though the outranking-based approaches require close interaction with the DM to reach a setting that acceptably reflects their preferences, this paper presented evidence that such an effort may be advantageously compensated in the framework of many-objective optimization via swarm-intelligence metaheuristics.

Further research is needed to explain the performance in connection with the properties of the problem and the number of objective functions.

## Acknowledgments

The authors thank CONACYT for supporting the Cátedras CONACYT program with number 3058 and the CONACYT project with number A1-S-11012. Carlos A. Coello Coello particularly acknowledges the support from CONACyT (grant no. 1920), and the SEP-Cinvestav grant (application no. 4). He was also partially supported by the Basque Government through the BERC 2018-2021 program by the Spanish Ministry of Science.

## References

- Abouhawwash, M., & Deb, K. (2021). Reference point based evolutionary multiobjective optimization algorithms with convergence properties using KKTPM and ASF metrics. *Journal of Heuristics*, 1–40. <https://doi.org/10.1007/s10732-021-09470-4>
- Acciarini, G., Izzo, D., & Mooij, E. (2020). MHACO: A Multiobjective Hypervolume-Based Ant Colony Optimizer for Space Trajectory Optimization. In 2020 IEEE Congress on Evolutionary Computation (CEC) (pp. 1–8). IEEE. <https://doi.org/10.1109/CEC48606.2020.9185694>
- Alaya, I., Solnon, C., & Ghédira, K. (2007). Ant colony optimization for multiobjective optimization problems. In: ICTAI 2007. Vol. 1. IEEE Computer Society Press, Los Alamitos, CA, 450–457. <https://doi.org/10.1109/ICTAI.2007.108>
- Balderas, F., Fernandez, E., Gómez, C., Rivera, G., Cruz-Reyes, L., & Rangel-Valdez, N. (2016). Uncertainty modelling for project portfolio problem using interval analysis, *International Journal of Combinatorial Optimization Problems and Informatics*, 7 (3) (2016) 20–27. <https://ijcopi.org/ojs/article/view/24>
- Balderas, F., Fernandez, E., Gomez, C., Rangel, N., & Cruz-Reyes, L. (2019). An interval-based approach for evolutionary multiobjective optimisation of project portfolios. *International Journal of Information Technology & Decision Making*, 18 (4), 1317–1358. <https://doi.org/10.1142/S021962201950024X>
- Bastiani, S., Cruz-Reyes, L., Fernandez, E., & Gomez, C. (2015). Portfolio optimization from a set of preference ordered projects using an ant colony based multiobjective approach, *International Journal of Computational Intelligence Systems*, 41–53. <https://doi.org/10.1080/18756891.2015.1129590>
- Bechikh S., Elarbi M., & Ben Said L. (2017). Many-objective Optimization Using Evolutionary Algorithms: A Survey, in: Recent Adv. Evol. Multiobjective Optim., Springer, Cham, 2017, pp. 105–137. [https://doi.org/10.1007/978-3-319-42978-6\\_4](https://doi.org/10.1007/978-3-319-42978-6_4)
- Branke, J., Kaubler, T., & Schmeck, H. (2001) Guidance in evolutionary multiobjective optimization, *Adv. Eng. Softw.*, 32, 499–507. [https://doi.org/10.1016/S0965-9978\(00\)00110-1](https://doi.org/10.1016/S0965-9978(00)00110-1)
- Branke J., Deb K. (2005). Integrating User Preferences into Evolutionary Multi-Objective Optimization. In: Jin Y. (eds) Knowledge Incorporation in Evolutionary Computation. Studies in Fuzziness and Soft Computing, vol 167. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-540-44511-1\\_21](https://doi.org/10.1007/978-3-540-44511-1_21)
- Branke, J., Corrente, S., Greco, S., Słowiński, R., & Zielniewicz, P. (2016). Using Choquet integral as preference model in interactive evolutionary multiobjective optimization, *European Journal of Operational Research*, 250(3), 884–901. <https://doi.org/10.1016/j.ejor.2015.10.027>
- Brams, J., & Mareschal, B. (2005). Multiple criteria decision analysis: state of the art surveys, *International Series on Operations Research & Management Science*, Springer-Verlag, Berlin, 2005, Ch. Promethee methods, pp. 163–190. [https://doi.org/10.1007/978-1-4939-3094-4\\_6](https://doi.org/10.1007/978-1-4939-3094-4_6)
- Brockhoff, D., Wagner, T., & Trautmann, H. (2012). On the properties of the R2 indicator. In 2012 Genetic and evolutionary computation

- conference (GECCO'2012) (pp. 465–472). Philadelphia: ACM Press. ISBN: 978-1-4503-1177-9. <https://doi.org/10.1145/2330163.2330230>
- Brockhoff, D., Bader J., Thiele L., & Zitzler E. (2013). Directed multiobjective optimization based on the weighted hypervolume indicator, *Journal of Multi-Criteria Decision Analysis*, 20 (5-6) 291–317. <https://doi.org/10.1002/mcda.1502>
- Chandra Mohan, B., & Baskaran, R. (2012). A survey: Ant Colony Optimization based recent research and implementation on several engineering domain, *Expert Systems with Applications*, 39(4), 4618–4627. <https://doi.org/10.1016/j.eswa.2011.09.076>
- Chica, M., Cordon O., Damas, S., & Bautista, J. (2011). Including different kinds of preferences in a multiobjective ant algorithm for time and space assembly line balancing on different Nissan scenarios, *Expert Systems with Applications*, 38(1), 709–720. <https://doi.org/10.1016/j.eswa.2010.07.023>
- Coello Coello, C. Veldhuizen D. & Lamont, G. (2007). *Evolutionary Algorithms for Solving MultiObjective Problems Second Edition*. <https://doi.org/10.1007/978-0-387-36797-2>
- Cruz, L., Fernandez, E., Gomez, C., Rivera, G., & Perez, F. (2014). Many-objective portfolio optimization of interdependent projects with 'a priori' incorporation of decision-maker preferences, *Applied Mathematics & Information Sciences*, 8 (4), 1517. <https://doi.org/10.12785/amis/080405>
- Cruz-Reyes, L., Fernandez, E., Sanchez, P., Coello Coello, C.A. & Gomez, C. (2017). Incorporation of implicit decision-maker preferences in multiobjective evolutionary optimization using a multi-criteria classification method, *Appl Soft Comput*, 50, 48–57. <https://doi.org/10.1016/j.asoc.2016.10.037>
- Cruz-Reyes, L., Fernandez, E., Sanchez-Solis, J.P., Coello Coello, C.A. & Gomez, C. (2020). Hybrid evolutionary multiobjective optimisation using outranking-based ordinal classification methods, *Swarm and Evolutionary Computation*, 54 (100652), <https://doi.org/10.1016/j.swevo.2020.100652>
- Cvetkovic, D., & Parmee, I.C. (2002) Preferences and their application in evolutionary Multiobjective optimization, *IEEE Trans. Evol. Comput.*, 6, 42–57. <https://doi.org/10.1109/4235.985691>
- Deb, K. (2001). Multiobjective optimization using evolutionary algorithms (Vol. 16). John Wiley & Sons.
- Deb, K., Thiele, L., Laumanns, M., & Zitzler, E. (2002). Scalable multiobjective optimization test problems. In Congress on evolutionary computation (CEC'2002) (vol. 1, pp. 825–830). Piscataway, NJ: IEEE Service Center. <https://doi.org/10.1109/CEC.2002.1007032>
- Deb, K., Sinha, A., Korhonen, P.J. & Wallenius, J. (2010). An interactive evolutionary multiobjective optimization method based on progressively approximated value functions, *IEEE Trans. Evol. Comput.*, 14 (5), 723–739. <https://doi.org/10.1109/TEVC.2010.2064323>
- Deb, K. & Jain, H. (2014). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints, *IEEE Trans. Evol. Comput.* 18 (4) (2014) 577–601. <https://doi.org/10.1109/TEVC.2013.2281535>
- Do Nascimento Ferreira, T., Araújo, A. A., Neto, A. D. B., & de Souza, J. T. (2016). Incorporating user preferences in ant colony optimization for the next release problem. *Applied Soft Computing*, 49, 1283-1296. <https://doi.org/10.1016/j.asoc.2016.06.027>
- Dorigo, M. & Stützle, T. (2004). Ant colony optimization. Cambridge, MA: MIT Press.
- Dorigo, M., & Blumb, C. (2005). Ant colony optimization theory: A survey, *Theoretical Computer Science*, 344, 243–278. <https://doi.org/10.1016/j.tcs.2005.05.020>
- Du, B., Chen, H., Huang, G.Q., & Yang, H.D. (2011). Preference Vector Ant Colony System for Minimising Make-span and Energy Consumption in a Hybrid Flow Shop. In: Wang L., Ng A., Deb K. (eds) Multi-objective Evolutionary Optimisation for Product Design and Manufacturing. Springer, London. [https://doi.org/10.1007/978-0-85729-652-8\\_9](https://doi.org/10.1007/978-0-85729-652-8_9)
- Dunwei, G., Fenglin, S., Jing, S., & Xiaoyan, S. (2017). Set-Based Many-Objective Optimization Guided by a Preferred Region, *Neurocomputing*, 228, 241–255. <https://doi.org/10.1016/j.neucom.2016.09.081>
- Emmerich, M. & Deutz, H. (2018). A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Natural Computing*, 17(3), 585–609. <http://dx.doi.org/10.1007/s11047-018-9685-y>
- Falcón-Cardona, J.G., & Coello Coello, C.A. (2016) iMOACO<sub>R</sub>: A New Indicator-Based Multiobjective Ant Colony Optimization Algorithm for Continuous Search Spaces. In: Handl J., Hart E., Lewis P., López-Ibáñez M., Ochoa G., Paechter B. (eds) Parallel

- Problem Solving from Nature – PPSN XIV. PPSN 2016. Lecture Notes in Computer Science, vol 9921. Springer, Cham. [https://doi.org/10.1007/978-3-319-45823-6\\_36](https://doi.org/10.1007/978-3-319-45823-6_36)
- Falcón-Cardona, J.G., & Coello Coello, C.A. (2017) A new indicator-based many-objective ant colony optimizer for continuous search spaces. *Swarm Intelligence*, 11(1), 71–100. <https://doi.org/10.1007/s11721-017-0133-x>
- Farina, M., & Amato, P. (2002): On the optimal solution definition for many-criteria optimization problems. In: Keller, J., Nasraoui, O. (eds.) Proc. of the NAFIPS-FLINT Int'l Conf. 2002, pp. 233–238. IEEE Computer Society Press, Piscataway. <https://doi.org/10.1109/NAFIPS.2002.1018061>
- Fernandez, E., Lopez, E., Bernal, S., Coello Coello, C. A. C., & Navarro, J. (2010). Evolutionary multiobjective optimization using an outranking-based dominance generalization. *Computers & Operations Research*, 37(2), 390–395. <https://doi.org/10.1016/j.cor.2009.06.004>
- Fernandez, E., Lopez, E., Lopez, F., & Coello Coello, C. A. (2011). Increasing selective pressure towards the best compromise in evolutionary multiobjective optimization: The extended NOSGA method. *Information Sciences*, 181(1), 44–56. <https://doi.org/10.1016/j.ins.2010.09.007>
- Fernandez, E., Gomez, C., Rivera, G., & Cruz-Reyes, L. (2015). Hybrid metaheuristic approach for handling many objectives and decisions on partial support in project portfolio optimisation, *Information Sciences*, 315, 102–122, <https://doi.org/10.1016/j.ins.2015.03.064>
- Fernandez, E., Gómez, C., Cruz Reyes, L., Rangel-Valdez, N., & Bastiani, S. (2017). Design and Solution of a Surrogate Model for Portfolio Optimization Based on Project Ranking. *Scientific Programming*, 2017, 1–10. <https://doi.org/10.1155/2017/1083062>
- Fernandez, E., Figueira, J.R., & Navarro, J. (2019). An indirect elicitation method for the parameters of the ELECTRE TRI-nB model using genetic algorithms, *Appl. Soft Comput.*, 77, 723–733. <https://doi.org/10.1016/J.ASOC.2019.01.050>
- Fernandez, E., Navarro, J., Solares, E., & Coello Coello C. A. (2020). Using evolutionary computation to infer the decision maker's preference model in presence of imperfect knowledge: A case study in portfolio optimization, *Swarm and Evolutionary Computation*, 54 (100648). <https://doi.org/10.1016/j.swevo.2020.100648>
- Figueira, J., & Roy, B. (2002). Determining the weights of criteria in the ELECTRE type methods with a revised Simos' procedure. *European Journal of Operational Research*, 139(2), 317–326. [https://doi.org/10.1016/S0377-2217\(01\)00370-8](https://doi.org/10.1016/S0377-2217(01)00370-8)
- Fliedner, T., & Liesio, J. (2016). Adjustable robustness for multi-attribute project portfolio selection, *European Journal of Operational Research*, 252(3), 931–946., <https://doi.org/10.1016/j.ejor.2016.01.058>
- French, S., (1993). Decision Theory: An Introduction to the Mathematics of Rationality. Ellis Horwood Series in Mathematics and its Applications, ISBN 9780132010139, Ellis Horwood Press. <https://books.google.com.mx/books?id=HcxBPwAACAAJ> (accessed: June 1, 2019).
- Gomez, C., Cruz-Reyes, L., Rivera, G., Rangel-Valdez, N., Morales-Rodriguez, L., & Perez-Villafuerte, M. (2018) Interdependent projects selection with preference incorporation, in: New Perspectives on Applied Industrial Tools and Techniques, Springer, 2018, pp. 253–271 [https://doi.org/10.1007/978-3-319-56871-3\\_13](https://doi.org/10.1007/978-3-319-56871-3_13)
- Grassé, P.-P. (1959) La reconstruction du nid et les coordinations interindividuelles chez bellicoitermes natalenis et cubitermes sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs, *Insect Sociaux*, 6, 41–81. <https://doi.org/10.1007/BF02223791>
- He, Y., He, Z., Kim, K. J., Jeong, I. J., & Lee, D. H. (2021). A Robust Interactive Desirability Function Approach for Multiple Response Optimization Considering Model Uncertainty. *IEEE Transactions on Reliability*, 70(1), 175–187. <https://doi.org/10.1109/TR.2020.2995752>
- Hernández Gómez, R., & Coello Coello, C. A. (2015). Improved metaheuristic based on the R2 indicator for many-objective optimization. In 2015 Genetic and evolutionary computation conference (GECCO 2015) (pp. 679–686). Madrid: ACM Press. ISBN 978-1-4503-3472-3. <https://doi.org/10.1145/2739480.2754776>
- Huband, S., Barone, L., While, L., & Hingston, P. (2005). A scalable multiobjective test problem toolkit. In C. A. Coello Coello, A. Hernández Aguirre, & E. Zitzler (Eds.), Third international conference evolutionary multi-criterion optimization (EMO 2005), Lecture Notes in Computer Science (vol. 3410, pp. 280–295). Springer. [https://doi.org/10.1007/978-3-540-31880-4\\_20](https://doi.org/10.1007/978-3-540-31880-4_20)
- Huband, S., Hingston, P., Barone, L., & While, L. (2006). A review of multiobjective test problems and a scalable test problem toolkit.

- IEEE Transactions on Evolutionary Computation*, 10(5), 477–506. <https://doi.org/10.1109/TEVC.2005.861417>
- Ishibuchi, H., Matsumoto, T., Masuyama, N., & Nojima, Y. (2020). Effects of dominance resistant solutions on the performance of evolutionary multiobjective and many-objective algorithms. In Proceedings of the 2020 Genetic and Evolutionary Computation Conference (pp. 507–515). <https://doi.org/10.1145/3377930.3390166>
- Jakubovski Filho, H.L., Nascimento Ferreira, T. & Vergilio Silvia, R. (2018). Incorporating User Preferences in a Software Product Line Testing Hyper-Heuristic Approach, 2018 IEEE Congress on Evolutionary Computation (CEC'2018), 2283–2290, IEEE Press, Rio de Janeiro, Brazil, July 8-13, ISBN: 978-1-5090-6017-7. <https://doi.org/10.1109/CEC.2018.8477803>
- Khelifa, B., & Laouar, M. R. (2020). A holonic intelligent decision support system for urban project planning by ant colony optimization algorithm. *Applied Soft Computing*, 106621. <https://doi.org/10.1016/j.asoc.2020.106621>
- Kulturel-Konak, S., Coit, D. W., & Baheranwala, F. (2008). Pruned Pareto-optimal sets for the system redundancy allocation problem based on multiple prioritized objectives. *Journal of Heuristics*, 14(4), 335. <https://doi.org/10.1007/s10732-007-9041-3>
- Li, K. (2019). Progressive preference learning: Proof-of-principle results in MOEA/D. In International Conference on Evolutionary Multi-Criterion Optimization, Springer, Cham. 631–643. [https://doi.org/10.1007/978-3-030-12598-1\\_50](https://doi.org/10.1007/978-3-030-12598-1_50)
- Li, K., Liao, M., Deb, K., Min, G., & Yao, X. (2020): Does preference always help? A holistic study on preference based evolutionary multiobjective optimisation using reference points. *IEEE Trans. Evolut. Comput.*, 24(6), 1078–1096. <https://doi.org/10.1109/TEVC.2020.2987559>
- Li, K., Deb, K., & Yao, X. (2017). R-metric: Evaluating the performance of preference-based evolutionary multiobjective optimization using reference points. *IEEE Transactions on Evolutionary Computation*, 22(6), 821–835. <https://doi.org/10.1109/TEVC.2017.2737781>
- Liu, Q., Jin, Y., Heiderich, M., Rodemann, T., & Yu, G. (2020a). An adaptive reference vector-guided evolutionary algorithm using growing neural gas for many-objective optimization of irregular problems. *IEEE Transactions on Cybernetics* (Early Access). <https://doi.org/10.1109/TCYB.2020.3020630>
- Liu, R., Zhou, R., Ren, R., Liu, J., & Jiao, L. (2020b). Multi-layer interaction preference based multiobjective evolutionary algorithm through decomposition. *Inf. Sci.*, 509, 420–436. <https://doi.org/10.1016/j.ins.2018.09.069>
- Mandal, A.K., & Dehuri, S. (2020) A Survey on Ant Colony Optimization for Solving Some of the Selected NP-Hard Problem. In: Dehuri S., Mishra B., Mallick P., Cho SB., Favorskaya M. (eds) Biologically Inspired Techniques in Many-Criteria Decision Making. BITMDM 2019. Learning and Analytics in Intelligent Systems, vol 10. Springer, Cham. [https://doi.org/10.1007/978-3-030-39033-4\\_9](https://doi.org/10.1007/978-3-030-39033-4_9)
- Meneghini, I. R., Alves, M. A., Gaspar-Cunha, A., & Guimaraes, F. G. (2020). Scalable and customizable benchmark problems for many-objective optimization. *Applied Soft Computing*, 90, 106139. <https://doi.org/10.1016/j.asoc.2020.106139>
- Miettinen, K., Ruiz, F., & Wierzbicki, A. P. (2008). Introduction to multiobjective optimization: Interactive approaches, in Multiobjective Optimization: Interactive and Evolutionary Approaches. Berlin, Germany: Springer, pp. 27–57. [https://doi.org/10.1007/978-3-540-88908-3\\_2](https://doi.org/10.1007/978-3-540-88908-3_2)
- Miller, G.A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information, *Psychol. Rev.*, 63 (2), 81–97. <https://doi.org/10.1037/h0043158>
- Moore, R.E. (1979). Methods and applications of interval analysis, Studies in Applied and Numerical Mathematics, *Society for Industrial and Applied Mathematics*, Philadelphia, USA.
- Parreiras, R. O., & Vasconcelos, J. A., (2007). A multiplicative version of Promethee II applied to multiobjective optimization problems. *European Journal of Operational Research*, 183(2), 729–740. <https://doi.org/10.1016/j.ejor.2006.10.002>
- Qi, Y., Li, X., Yu, J., & Miao, Q. (2019). User-preference based decomposition in MOEA/D without using an ideal point, *Swarm and Evolutionary Computation*, 44, 597–611. <https://doi.org/10.1016/j.swevo.2018.08.002>
- Rangel-Valdez, N., Gómez-Santillán, C., Hernández-Marín, J., Morales-Rodríguez, M., Cruz-Reyes, L., & Fraire-Huacuja, H. (2020). Parallel designs for metaheuristics that solve portfolio selection problems using fuzzy outranking relations, *International Journal of Fuzzy Systems*, 22, 1–13. <https://doi.org/10.1007/s40815-019-00794-9>
- Rivera, G., Gómez, G., Cruz, L., García, R., Balderas, F., Fernández, E., & López, F. (2012). Solution to the social portfolio problem by evolutionary algorithms, *International Journal of Combinatorial Optimization Problems and Informatics*, 3 (2), 21–30.

<https://www.redalyc.org/pdf/2652/265224454004.pdf>

- Rivera, G., Florencia, R., Guerrero, M., Porras, R., & Sánchez-Solís, J. P. (2021). Online multi-criteria portfolio analysis through compromise programming models built on the underlying principles of fuzzy outranking. *Information Sciences*, 580, 734–755. <https://doi.org/10.1016/j.ins.2021.08.087>
- Rodríguez-Fdez, I., Canosa, A., Mucientes, M., & Bugarín, A. (2015). STAC: a web platform for the comparison of algorithms using statistical tests. In 2015 IEEE international conference on fuzzy systems (FUZZ-IEEE) (pp. 1-8). IEEE. <https://doi.org/10.1109/FUZZ-IEEE.2015.7337889>
- Roy, B., & Vanderpooten, D. (1996). The European school of MCDA: Emergence, basic features and current works. *Journal of Multi-Criteria Decision Analysis*, 5, 22–38. [https://doi.org/10.1002/\(SICI\)1099-1360\(199603\)5:1<22::AID-MCDA93>3.0.CO;2-F](https://doi.org/10.1002/(SICI)1099-1360(199603)5:1<22::AID-MCDA93>3.0.CO;2-F)
- Roy, B., (1990a). Reading in multiple criteria decision aid, Springer-Verlag, Ch. The outranking approach and the foundations of ELECTRE methods, pp. 155–183. [https://doi.org/10.1007/978-3-642-75935-2\\_8](https://doi.org/10.1007/978-3-642-75935-2_8)
- Roy, B., (1990b). The Outranking Approach and the Foundations of Electre Methods, in: Readings Mult. Criteria Decis. Aid, Springer Berlin Heidelberg, Berlin, Heidelberg, 155–183. [https://doi.org/10.1007/978-3-642-75935-2\\_8](https://doi.org/10.1007/978-3-642-75935-2_8).
- Saldanha, W., Arrieta, F., Machado-Coelho, T., Santos, E., Maia, C., Ekel, P., & Soares, G. (2019). Evolutionary algorithms and the Preference Ranking Organization Method for Enrichment Evaluations as applied to a multiobjective design of shell-and-tube heat exchangers. *Case Studies in Thermal Engineering*, 17, 100564. <https://doi.org/10.1016/j.csite.2019.100564>
- Socha, K., & Dorigo, M. (2008). Ant colony optimization for continuous domains. *European Journal of Operational Research*, 185(3), 1155–1173. <https://doi.org/10.1016/j.ejor.2006.06.046>
- Taboada, H.A., Baheranwala, F. & Coit, D.W. (2007). Practical solutions for multiobjective optimization: an approach to system reliability design problems. *Reliability Engineering and System Safety*, 92, 314–322. <https://doi.org/10.1016/j.res.2006.04.014>
- Tian, Y., Cheng, R., Zhang, X., Cheng, F., & Jin, Y. (2018). An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility. *IEEE Transactions on Evolutionary Computation*, 22(4), 609-622. <https://doi.org/10.1109/TEVC.2017.2749619>
- Tomczyk, M. K. & Kadziński, M. (2020). Decomposition-Based Interactive Evolutionary Algorithm for Multiple Objective Optimization, *IEEE Transactions on Evolutionary Computation*, 24(2), 320–334. <https://doi.org/10.1109/TEVC.2019.2915767>
- Wagner, T., & Trautmann, H. (2010). Integration of preferences in hypervolume-based multiobjective evolutionary algorithms by means of desirability functions, *IEEE Trans. Evol. Comput.*, 14, 688–701. <https://doi.org/10.1109/TEVC.2010.2058119>
- Wang, R., Purshouse, R.C., & Fleming P.J. (2015). Preference-inspired co-evolutionary algorithms using weight vectors. *Eur J Oper Res* 243(2), 423–441. <https://doi.org/10.1016/j.ejor.2014.05.019>
- Xin, B., Chen, L., Chen, J., Ishibuchi, H., Hirota, K., & Liu, B. (2018). Interactive multiobjective optimization: A review of the state-of-the-art. *IEEE Access*, 6, 41256–41279. <https://doi.org/10.1109/ACCESS.2018.2856832>
- Wang, Y., Limmer, S., Olhofer, M., Emmerich, M., & Bäck, T. (2021). Automatic Preference Based Multi-objective Evolutionary Algorithm on Vehicle Fleet Maintenance Scheduling Optimization. *Swarm and Evolutionary Computation*, 243(2), 423–441. <https://doi.org/10.1016/j.ejor.2014.05.019>
- Yi, J., Bai, J., He, H., Peng, J., & Tang D. (2019). ar-MOEA: A Novel Preference-Based Dominance Relation for Evolutionary Multiobjective Optimization, *IEEE Transactions on Evolutionary Computation*, 23(5), 788–802. <https://doi.org/10.1109/TEVC.2018.2884133>
- Yuan, M.H., Chiueh, P.T., & Lo, S.L. (2021). Measuring urban food-energy-water nexus sustainability: Finding solutions for cities. *Science of the Total Environment*, 752, 141954. <https://doi.org/10.1016/j.scitotenv.2020.141954>
- Zhang, Q., & Li, H. (2007). MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation*, 11, 712–731. <https://doi.org/10.1109/tevc.2007.892759>
- Zhang, Y., Yang, G.W., Guo, D.L., Sun, Z.X., & Chen, D.W. (2019). A novel CACOR-SVR multi-objective optimization approach and its application in aerodynamic shape optimization of high-speed train. *Soft Computing*, 23, 5035–5051. <https://doi.org/10.1007/s00500-018-3172-3>
- Zhao, C., Zhou, Y., & Chen, Z. (2021). Decomposition-based evolutionary algorithm with automatic estimation to handle many-objective optimization problem. *Information Sciences*, 546, 1030-1046. <https://doi.org/10.1016/j.ins.2020.08.084>

Zhou, S.-Z., Zhan, Z.-H., Chen, Z.-G., Kwong, S., & Zhang, J. (2021). A Multiobjective Ant Colony System Algorithm for Airline Crew Rostering Problem with Fairness and Satisfaction. *IEEE Transactions on Intelligent Transportation Systems*, 22(11), 6784–6798.  
<https://doi.org/10.1109/tits.2020.2994779>