



Multi-objective optimization using metaheuristics: non-standard algorithms

El-Ghazali Talbi^{a,b}, Matthieu Basseur^c, Antonio J. Nebro^d and Enrique Alba^d

^a*INRIA-University of Lille, Bat.M3 59655, Villeneuve d'Ascq, France*

^b*King Saud University, Riyadh, Saudi Arabia*

^c*Laboratoire d'Etudes et de Recherche en Informatique d'Angers, 2 bd Lavoisier, 49000 Angers, France*

^d*Departamento de Lenguajes y Ciencias de la Computación, University of Málaga, Campus de Teatinos, Málaga 29071, Spain*

E-mail: talbi@lifl.fr [Talbi];asseur@info.univ-angers.fr [Basseur]; antonio@lcc.uma.es [Nebro]; eat@lcc.uma.es [Alba]

Received 30 May 2010; received in revised form 5 January 2011; accepted 6 January 2011

Abstract

In recent years, the application of metaheuristic techniques to solve multi-objective optimization problems (MOPs) has become an active research area. Solving these kinds of problems involves obtaining a set of Pareto-optimal solutions in such a way that the corresponding Pareto front fulfills the requirements of convergence to the true Pareto front and uniform diversity. Most studies on metaheuristics for multi-objective optimization are focused on Evolutionary Algorithms, and some of the state-of-the-art techniques belong to this class of algorithms. Our goal in this paper is to study open research lines related to metaheuristics but focusing on less explored areas to provide new perspectives to those researchers interested in multi-objective optimization. In particular, we focus on non-evolutionary metaheuristics, hybrid multi-objective metaheuristics, parallel multi-objective optimization, and multi-objective optimization under uncertainty. We analyze these issues and discuss open research lines.

Keywords: multi-objective optimization; metaheuristics; hybridization; parallelism; optimization under uncertainty

1. Introduction

Many sectors of industry (mechanical, chemistry, telecommunication, environment, transport, etc.) are concerned with complex problems of large dimensions that must be optimized. These optimization problems are seldom single-objective; on the contrary, they frequently have several contradictory criteria or objectives that must be satisfied simultaneously. Multi-objective optimization is a discipline centered on the resolution of these kinds of problems. It has its roots in the 19th century in the economics works of Edgeworth and Pareto (Pareto, 1896). Initially, it was applied to the economic sciences and management, and gradually to the engineering sciences.

As in single-objective optimization, the techniques to solve a *multi-objective optimization problem* (MOP) can be classified into *exact* and *approximate* (also called *heuristic*) algorithms. Exact methods such as *Branch and Bound* (Sen et al., 1988; Visée et al., 1998), the *A* algorithm* (Stewart and White, 1991), and *Dynamic Programming* (Carraway et al., 1990) are effective for problems of small sizes. When problems become harder, usually because of their NP-hard complexity, approximate algorithms are mandatory.

In recent years, a group of approximate optimization techniques, known as *metaheuristics*, have become an active research area. Although there is no commonly accepted definition of metaheuristics (Blum and Roli, 2003), they can be considered as high-level strategies that guide a set of simpler techniques in the search for an optimum. Among these techniques, evolutionary algorithms for solving MOPs are very popular in multi-objective optimization, giving rise to a wide variety of algorithms, such as NSGA-II (Deb et al., 2002), SPEA2 (Zitzler et al., 2001), PAES (Knowles and Corne, 1999), and many others (Coello et al., 2002; Deb, 2001).

In general, optimization problems (single or multi-objective) can be divided into two categories (Blum and Roli, 2003): those whose solutions are encoded with *real-valued* variables, also known as *continuous optimization problems*, and those where the solutions are encoded using *discrete* variables. Among the latter ones, we find a class of problems named *combinatorial optimization*–problems. When these problems are multi-objective, they are usually called *multi-objective combinatorial optimization problems* (MCOPs) (also multi-objective combinatorial optimization – MOCO – problems; Gandibleux and Ehrgott, 2005). In the past few years, many metaheuristics for solving MOPs were designed to deal with continuous problems; however, there is now a growing interest in solving MCOPs, because many real problems belong to this class.

Multi-objective optimization seeks to optimize several components of a cost function vector. Contrary to single-objective optimization, the solution of a MOP is not a single solution, but a set of solutions known as a *Pareto optimal set*, which is called a Pareto border or a *Pareto front* when it is plotted in the objective space. Any solution of this set is optimal in the sense that no improvement can be made on a component of the objective vector without worsening at least another of its components. The main goal in the resolution of a multi-objective problem is to obtain a set of solutions within the Pareto optimal set and, consequently, the Pareto front. Notwithstanding, when approximate techniques such as metaheuristics are applied, the goal becomes to obtain a set of solutions having two properties: convergence to the true Pareto front and homogeneous diversity. The first property ensures that we are dealing with non-dominant solutions, while obtaining a uniform-spaced set of solutions indicates that we have carried out a good exploration of the search space, and so we are not losing valuable information.

Currently, there are many open research lines on multi-objective optimization using metaheuristics. Among them are the design of new algorithms, the definition of metrics for performance assessment, to establish a commonly accepted benchmark of MOPs, and the definition of accurate density estimators. These are major topics that appear frequently in international conferences and journals on multi-objective optimization.

In this paper, we focus on some open research lines in order to provide new perspectives in the field. As mentioned before, many works on metaheuristics for multi-objective optimization are related to evolutionary algorithms, and some books have been published about this issue, such as the well-known ones by Coello et al. (2002) and Deb (2001). Nevertheless, there is a trend to adapt

other kinds of metaheuristics (sometimes called *alternative methods*, with reference to Evolutionary Algorithms), such as Particle Swarm Optimization (Reyes-Sierra and Coello, 2005), Scatter Search (Nebro et al., 2005a), or Ant Colony Optimization (Doerner et al., 2002). The second topic is hybrid multi-objective metaheuristics, which combine concepts of different techniques aimed at providing a more efficient behavior, which is of particular interest when solving real-world problems. In this sense, parallel multi-objective optimization is also a way to obtain more powerful techniques to solve large-scale problems. Parallelism has been widely applied to single-objective optimization (Alba and Tomassini, 2002; Cantú-Paz, 2000); however, the parallel models that have been successfully used in the single-objective field are not directly applicable to solve MOPs, due to the difficulties related to obtain a Pareto front with good convergence and diversity. Finally, our last interesting point is multi-objective optimization under uncertainty, which is related to the fact that some real-world MOPs are subject to uncertainties caused by missing information or stochastic models.

Our goal in this work is to provide an overview of these four topics to provide the reader with some insights into promising research lines related to multi-objective metaheuristics for solving MOPs. In general, we will not discuss the techniques in detail, because many of them are widely known in single-objective optimization (see Blum and Roli, 2003, for example); instead, we will concentrate on analyzing representative works. As we do not intend to provide a complete survey of these issues, our preferences will be biased, without loss of generality, to examples mainly related to solving MCOPs.

The paper is organized as follows: in Section 2, we include some background on multi-objective optimization fundamentals. Section 3 is devoted to alternative metaheuristics for solving MOPs. In Section 4, we analyze hybrid metaheuristics. A review of parallel metaheuristics for multi-objective optimization is included in Section 5. Section 6 studies multi-objective metaheuristics under uncertainty. Finally, in Section 7, we present the conclusions and some research prospects.

2. Multi-objective optimization fundamentals

In this section, we include some background on multi-objective optimization. We first define basic concepts, such as Pareto optimality, Pareto dominance, Pareto optimal set, and Pareto front. In these definitions, we are assuming, without loss of generality, the minimization of all the objectives.

A general multi-objective optimization problem (MOP) can be formally defined as follows:

$$[\text{MOP}] \text{ (MOP)} = \begin{cases} \min F(x) = (f_1(x), f_2(x), \dots, f_n(x)), \\ \text{s.t. } x \in \Omega, \end{cases}$$

$n \geq 2$ being the number of objective functions and $x = (x_1, x_2, \dots, x_r)$ the decision variable vector. The set of all values satisfying the constraints defines the *feasible region* Ω and any point $\vec{x} \in \Omega$ is a *feasible solution*. As mentioned before, we seek the *Pareto optima*. Its formal definition is provided next:

[Pareto Optimality] A point $\vec{x}^* \in \Omega$ is the Pareto optimal if for every $\vec{x} \in \Omega$ and $I = \{1, 2, \dots, k\}$ either $\forall_{i \in I} (f_i(\vec{x}) = f_i(\vec{x}^*))$ or there is at least one $i \in I$ such that $f_i(\vec{x}) > f_i(\vec{x}^*)$.

This definition states that \vec{x}^* is Pareto optimal if no feasible vector \vec{x} exists that would improve some criterion without causing a simultaneous worsening in at least one other criterion. Other important definitions associated with Pareto optimality are as follows:

[Pareto Dominance] A vector $\vec{u} = (u_1, \dots, u_k)$ is said to dominate $\vec{v} = (v_1, \dots, v_k)$ (denoted by $\vec{u} \preceq \vec{v}$) if and only if \vec{u} is partially less than \vec{v} , i.e., $\forall i \in \{1, \dots, k\}, u_i \leq v_i \wedge \exists i \in \{1, \dots, k\}: u_i < v_i$.

[Pareto Optimal Set] For a given MOP $\vec{f}(\vec{x})$, the Pareto optimal set is defined as

$$\mathcal{P}^* = \{ \vec{x} \in \Omega \mid \neg \exists \vec{x}' \in \Omega, \vec{f}(\vec{x}') \preceq \vec{f}(\vec{x}) \}.$$

[Pareto Front] For a given MOP $\vec{f}(\vec{x})$ and its Pareto optimal set \mathcal{P}^* , the Pareto front is defined as $\mathcal{PF}^* = \{ \vec{f}(\vec{x}), \vec{x} \in \mathcal{P}^* \}$.

Obtaining the Pareto front of a MOP is the main goal of multi-objective optimization. However, given that a Pareto front can contain a large number of points, which are not easily enumerable, a good solution would be to find a limited number of them, which should be as close as possible to the exact Pareto front and would also be uniformly spread. Otherwise, they would not be very useful to the decision maker. Let us examine the three fronts included in Fig. 1. The set of solutions represented by triangles shows a front having a very good spread of solutions, but the points are far from the true Pareto front; this front is not attractive because it does not provide efficient points. The set of solutions represented by crosses contains a set of solutions that are very close to the true Pareto front; thus, some regions of the true Pareto front are not covered, and so the decision maker could lose important trade-off solutions. Finally, the set of solutions represented by circles has the two desirable properties of good convergence and diversity.

It is worth pointing out that ideally, we would like to obtain a solution minimizing all the objectives. Let us suppose that the optimum for each objective function is known; then, we can define the concept of an *ideal vector*:

[Ideal Vector] A point $\vec{y}^* = [y_1^*, y_2^*, \dots, y_n^*] \in \Omega$ is an ideal vector if it minimizes each objective function in $f(\vec{x})$, i.e., $y_k^* = \min(f_k(\vec{x}))$.

Unfortunately, ideal points rarely exist in real problems where the criteria are in conflict. Related to this concept is the concept of *reference vector*, which can be used when we are interested in defining the goal intended to be reached for each objective.

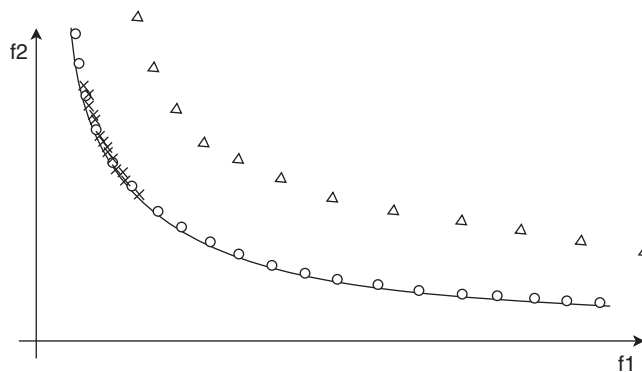


Fig. 1. Examples of Pareto fronts (solid line: Pareto optimal set): bad convergence and good diversity (triangles), good convergence and bad diversity (crosses), good convergence and diversity (circles).

[Reference Vector] A point $\vec{z}^* = [z_1^*, z_2^*, \dots, z_n^*] \in \Omega$ is a reference vector when it defines the goals to be reached for each objective $f_k(\vec{x})$.

In single-objective optimization, the concept of an optimal solution is frequently related to *suboptimal solutions*, also known as *local minima*. In general, metaheuristic algorithms try to avoid becoming trapped in local optima when searching for an optimal solution. In multi-objective optimization, the concept of local minima is replaced by a *locally Pareto optimal solution*. This notion is related to the concept of *neighborhood*, usually applied in metaheuristics based on Local Search:

[Neighborhood] A neighborhood N is a function $N(\vec{x}), x \in \Omega$ that assigns to each vector \vec{x} a set of points called neighbors.

We can define now the concept of local optimality in a multi-objective context:

[Locally Pareto Optimal Solution] A solution \vec{x} is locally Pareto optimal if and only if $\forall \vec{w} \in N(\vec{x}), w$ does not dominate x .

3. Non-evolutionary approaches for multi-objective optimization

Although most metaheuristics designed to solve MOPs are evolutionary algorithms, there exist other techniques that can be used for the resolution of these problems. For example, metaheuristics such as Tabu Search or Scatter Search have proved their ability to find good solutions in many optimization problems; hence, there are proposals focused on extending these strategies to deal with multiple objectives, hoping that their performance will also be extended to the multiple objective case. Moreover, as shown in Section 4, some of these multi-objective extensions are proposed in order to design hybrid approaches between a classical multi-objective algorithm and other different techniques, aimed at producing a more efficient search method.

In this section, we analyze alternative approaches to multi-objective optimization. We consider the following techniques: Local Search, Simulated Annealing, Tabu Search, Path Relinking, Scatter Search, Ant Colony Optimization, and Particle Swarm Optimization.

3.1. Pareto optimization and Local Search

Local Search is a general technique based on an iterative process in which a current solution is perturbed somehow, producing a new solution. If this solution is better than the former, the new solution becomes the current one. In single-objective optimization, Local Search has the property of promoting search intensification; moreover, it allows a fast convergence rate for problems of many types and sizes. Many Local Search proposals are applied in the context of other metaheuristics, leading to hybrid approaches (see Section 4); for this reason, here, we discuss a few general ideas, analyzing two works in this field.

The main question in a multi-objective Local Search is how to find a diversified set of good solutions, as the basic principle of Local Search involves the evolution of a single solution. Let us consider the Pareto Local Search (PLS) algorithm (Basseur et al., 2003). This algorithm works with a population of non-dominated solutions, called PO . For each Local Search step, the neighborhood PN of each solution of PO is generated, and PO takes the non-dominated solutions of $PO \cup PN$ as a new value. In many cases, the set of non-dominated solutions to be stored can be too large, and so the user has to apply a clustering step during PLS. A study on dominance-based

Local Search approaches for multi-objective combinatorial optimization is proposed in Liefvooghe et al. (2009). In this paper, different simple versions of dominance-based Local Searches are proposed and tested on different combinatorial problems.

Other approaches work by reducing the multi-objective problem to a single-objective one using a weighted sum approach. An example is the algorithm presented in Ishibuchi and Murata (1998), which is applied to flow-shop scheduling. The problem with these techniques is how to define the weights; in Ishibuchi and Murata (1998), they are randomly specified. This work also takes into account only a small number of neighbor solutions to prevent the Local Search procedure from using too much computing time. The approach of converting a multi-objective problem into a single-objective one is not very popular in the multi-objective research community, because it only provides a unique solution, thus reducing the chances of the decision maker choosing a good trade-off solution to the problem.

The main difficulty with multi-objective Local Searches is control of the population size. In Pareto approaches, the number of non-dominated solutions considered have to be controlled, and in weighted sum approaches, the searches have to be iterated intelligently in order to obtain a set of diversified non-dominated solutions. Hence, the approach proposed in Basseur and Burke (2007) is interesting. In this paper, a multi-objective search in which the selection is realized using a quality indicator is proposed, and the Local Search is carried out using a fixed number of solutions in the population.

3.2. Multi-objective Simulated Annealing

Simulated Annealing algorithms are commonly said to be the oldest among the metaheuristics. It is basically a Local Search with an explicit strategy to avoid local minima. The basic idea is to allow solutions of a quality worse than the current solution; this is done with a probability that is decreased during the search.

One of the first attempts to use simulated annealing to solve MOPs was presented in Serafini (1992). Since then, there have been many proposals in the literature, such as Czyzack and Jaszkiwicz (1998); Chipperfield et al. (1999) and Karasakal and Köksalan (2000). Most of these algorithms do not have a population, but they store the non-dominated solutions discovered during a Local Search process. Rather than using Pareto dominance, weighted metrics are used to aggregate the objectives into a single score to be used in the acceptance function. Thus, multi-objective Simulated Annealing has the same disadvantages as those mentioned in the previous section concerning Local Search. The latest papers dealing with multi-objective Simulated Annealing are trying to incorporate the concept of dominance into their algorithm, such as in Smith et al. (2008).

3.3. Multi-objective Tabu Search

Tabu Search is a metaheuristic based on explicitly using the history of the search, both to escape from local minima and to implement an exploration strategy. Here, we discuss the proposals MOTS and TAPaS.

M.P. Hansen proposed a multi-objective Tabu Search procedure called MOTS (MultiObjective Tabu Search) (Hansen, 1997). This algorithm works with a set of current solutions, which,

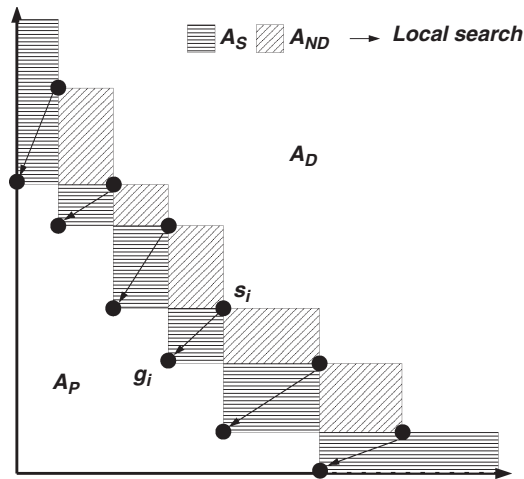


Fig. 2. TAPaS: the goal g_i of a solution s_i is defined as a function of s_i neighbors (in the objective space). Then, the search is realized in A_S (space dominated by a solution g_i and dominating a solution s_i) and in A_{ND} (space dominated by two solutions g_i and g_j and not dominated by any solution s_i). A_D represents the objective space dominated by at least one solution s_i and A_P represents the objective space that is not dominated by any solution g_i .

through manipulation of weights, are optimized towards the non-dominated front while at the same time attempting to disperse them over the Pareto front. The MOTS algorithm was updated later to TAMOCO (Tabu Search for MultiObjective Combinatorial Optimization) (Hansen, 2000). In TAMOCO, a mechanism is introduced in order to force solutions to drift over the Pareto front. In this paper, a parallel version of TAMOCO has also been implemented.

A Tabu Search algorithm using a trade-off between a Pareto dominance-based and an aggregation search is described by Jozefowicz (2004), and is called the Target Aiming Pareto Search (TAPaS). This algorithm considers a set of non-dominated solutions found by an evolutionary algorithm, with a good quality and diversity. Then, a search l_i is applied on each solution s_i of an initial Pareto set. A specific single-objective function θ_i is defined for each search l_i . The defined function takes into account that two searches carried out simultaneously do not have to explore the same area of the objective space. The goal is to intensify the search around the solutions found by the evolutionary algorithm, without having a loss in terms of diversity. The goal definition is illustrated in Fig. 2.

Several other Tabu Search methods have been proposed in the literature, such as Hertz et al. (1994) and Beausoleil (2001). Beausoleil's proposal uses a weighted objective Tabu Search to build an initial population, and Hertz defines three different non-Pareto approaches (weighted, lexicographic, and ε -constraint). In general, most of the proposed techniques do not adopt an entire Pareto dominance approach. Other multi-objective Tabu Search approaches were also proposed by Ulungu et al. (1999) and Gandibleux et al. (1997).

3.4. Multi-objective Scatter Search

Scatter Search is a metaheuristic technique that, compared with evolutionary algorithms, intends to avoid using stochastic operators as much as possible. The working principle of Scatter Search is

to have a reference set of solutions composed of, on the one side, a subset of solutions having good fitness values and, on the other side, another subset of solutions promoting diversity. The elements of this reference set are then systematically combined producing new trial solutions that are used to update that set. The combination of solutions is based on making linear combinations of the decision variable values, and the trial solutions are enhanced using a Local Search algorithm. Scatter search is described in detail in Glover et al. (2003).

Recently, the application of Scatter Search to multi-objective optimization problems has received some attention from some researchers (Beausoleil, 2001; Corberán et al., 2002). In Beausoleil (2006), the MOSS algorithm is presented, which proposes a Tabu/Scatter Search hybrid approach for solving nonlinear multi-objective optimization problems. SSPMO, described in Molina et al. (2007), proposes a similar hybrid approach, but with a different Tabu Search.

There are studies indicating that the search capabilities of Scatter Search can be improved using stochastic operators in the algorithm (Herrera et al., 2006). In this respect, Nebro et al. (2005a) introduce an SSMO, a multi-objective Scatter Search using a mutation operator as the Local Search. This proposal is characterized by using the initial set as an external population to store the non-dominated solutions found during the search. An evolution of SSMO is AbYSS (Archive-based hybrid Scatter Search) (Nebro et al., 2006), which uses an external archive to store the non-dominated solutions. Several configuration of AbYSS, starting from an orthodox Scatter Search approach (using non-stochastic operators) to configurations using genetic operators, have been studied in Nebro et al. (2006), showing that the algorithm produces very high-quality Pareto fronts when using a polynomial mutation in the Local Search method and a simulated binary crossover (SBX) to recombine solutions.

These works indicate that Scatter Search is a promising approach to multi-objective optimization. Some open research lines are related to how to re-use the search experience present in the non-dominated solutions found by the algorithm, the update policies to be applied to the reference set in order to enhance the diversity of the solutions, and configuration analyses to study how to accelerate the convergence of the technique.

3.5. Multi-objective Path Relinking

Path Relinking (Glover et al., 2000) is a generalization of scatter search, in which the concept of combining solutions by making linear combinations of reference points is generalized to neighborhood spaces. The idea is to generate a path between two solutions in the search space by generating a sequence of neighboring solutions in the decision space.

In Beausoleil (2001), a first investigation was proposed to include Path Relinking algorithms in a multi-objective Scatter Search algorithm. Two neighborhood operators were used to generate the paths, and the distance used was not correlated with these measures. After the path relinking process, the non-dominated solutions are selected to pursue the Scatter Search algorithm. A Path Relinking algorithm using the same individual representation is proposed in Basseur et al. (2005), although it is applied to a different problem. In this study, only the most powerful neighborhood operator is used, and the proposed distance measure is correlated with the neighborhood operator. This allows the generation of only the shortest paths without generating any other solution. Then, a Multi-Objective Path-Relinking (MOPR) algorithm is presented, which is used in combination with genetic and Local Search algorithms.

Several questions are formulated in this paper concerning how to adapt Path Relinking mechanisms to the multi-objective case.

More recently Jaszkiwicz (2005) has proposed including the path relinking principle in a Pareto Memetic Algorithm, which uses random scalarizations of the objective functions during the Local Search step. Although this approach does not use the concept of Pareto dominance, it proposes an interesting algorithm that could be applied to Pareto optimization by defining several selection methods that can replace the scalar selection.

In multi-objective optimization, as well as in single-objective optimization, there are only a few papers that propose Path Relinking algorithms. One reason is that for some optimization problems, the path generation is very difficult to define. However, the Path Relinking principle is an interesting and growing research area for multi-objective optimization, and some other studies were proposed recently (Beausoleil et al., 2008; Pasia et al., 2010). There are many research perspectives in order to adapt this method to multi-objective optimization, such as the path generation, how to select the solutions in the path, or choosing the solutions to participate in the parent selection.

3.6. Multi-objective Ant Colony Optimization

Ant Colony Optimization (ACO) is a metaheuristic approach proposed by Dorigo and Blum (2005). It is a bio-inspired technique taking the foraging behavior of real ants as a model. When ants walk from food sources to the nest and vice versa, they deposit a chemical substance called pheromone on the ground. To make a decision on a direction to take, ants choose with a higher probability those paths marked by stronger pheromone concentrations. ACO algorithms use a probabilistic model using a pheromone matrix to model the chemical pheromone trails.

In Gambardella et al. (1999), Gambardella and colleagues propose solving a bi-objective routing problem by ant colony systems. They propose MACS-VRPTW (Multiple Ant Colony System for Vehicle Routing Problems with Time Windows). MACS-VRPTW is organized as a hierarchy of artificial ant colonies designed to successively optimize a multiple-objective function: the first colony minimizes the number of vehicles while the second colony minimizes the distances traveled. Cooperation between colonies occurs by exchange of information through pheromone updating. MACS-VRPTW improves some of the best solutions known for a number of problem instances in the literature.

Mariano and Morales proposed ANT-Q, which was also designed to deal with multiple objectives (Mariano and Morales, 1999). In this study, the proposed algorithm could be compared with parallel selection (Schaffer, 1985), i.e. one ant colony is associated with one objective function. In Iredi et al. (2001), Iredi and colleagues detail a similar scheme, but each population is associated with an objective function corresponding to a weighted sum of the different criteria to optimize. More recently, in López-Ibáñez and Stützle (2010), although a similar approach is proposed, the paper also proposes allowing the application of the automatic algorithm configuration technique.

Over several years, a growing number of studies have focused on multi-objective Ant Colony systems. In fact most of the proposed approaches use several ant colonies during the optimization process, i.e. one colony per objective function or one colony per objective weight vector. Approaches involving only one colony and using Pareto dominance are interesting ideas to be exploited in this area.

3.7. Multi-objective Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a bio-inspired metaheuristic based on the flight of a flock of birds when they search for food (other analogies are considered, such as the behavior of fish schooling) (Kennedy and Eberhart, 1995). Compared with evolutionary algorithms, PSO is also a population-based technique with a population of random solutions aimed at finding an optimum by successive generations. However, in PSO, there are no evolutionary operators such as crossover and mutation; instead, the particles (solutions) fly through the problem space by following the best particle known so far.

PSO has become a popular metaheuristic in single-objective optimization and, recently, there has been a growing interest in its application to multi-optimization. In a survey published in 2006 (Reyes-Sierra and Coello Coello, 2006), more than 20 works were covered. Here, we comment on the proposals of Li (2004) and Reyes-Sierra and Coello Coello (2005).

In Li (2004), Li describes maximinPSO, a PSO algorithm characterized by using a fitness function derived from the maximin strategy to determine Pareto domination. With this strategy, there is no need to use an additional clustering technique. The reported results show that the algorithm produces very good Pareto fronts when solving some benchmarks.

Reyes-Sierra and Coello Coello (2005) present a multi-objective PSO based on Pareto dominance and the use of a nearest-neighbor density estimator for the selection of the best solutions. The algorithm uses two archives: one for the final population and the other for storing the best solutions currently used. An interesting feature is that the flock is divided into three subswarms, and a different mutation operator is applied to the particles of each subswarm.

Since 2006, a growing number of studies have been dedicated to multi-objective PSO (Janson et al., 2008; Ono and Nakayama, 2009), because of the combined growing popularity of multi-objective optimization and PSO algorithms. Some open issues in multi-objective PSO are strategies for choosing the best particles, studies on self-adaptation of some of the parameters in the algorithm, and the application of this technique to real-world problems.

4. Hybrid multi-objective metaheuristics

Until the 1990s, the main focus of research in the metaheuristic field was on the application of single techniques to concrete problems. Nowadays, it has become evident that using a unique metaheuristic is sometimes not sufficient. A skillful combination of concepts from different metaheuristics, called hybrid metaheuristics, can provide a more efficient behavior and a higher flexibility when dealing with real-world and large-scale problems.

The design and implementation of hybrid metaheuristics give rise to problems beyond those encountered in the design of a single metaheuristic; for example, the choice and tuning of parameters leads to the issue of how to achieve a proper interaction of the different algorithm components. Interaction can take place at a low level, using functions from different metaheuristics, but also at a high level, e.g., using a portfolio of metaheuristics for automated hybridization. In Talbi (2002), a taxonomy of hybrid metaheuristics (in the single-objective domain) is proposed. This taxonomy considers two features, leading to four different types of hybrid metaheuristics if we combine them:

- Low-level/High-level: the low-level hybrids address the functional composition of a single optimization method. In this class, a given function of a metaheuristic is replaced by another metaheuristic. In high-level hybrid algorithms, the different metaheuristics are self-contained.
- Relay/Teamwork: in relay optimization, a set of metaheuristics is applied one after another, each using the output of the previous as its input, acting in a pipeline fashion. Teamwork hybrids represent cooperative optimization models, in which several agents cooperate in a parallel way, each agent carrying out a search in a solution space.

This classification could be adapted for multi-objective algorithms. In general, with very large problems and/or multi-objective problems, the efficiency of single metaheuristics may be compromised. Hence, in this context, it is necessary to integrate metaheuristics into more general schemes in order to develop more efficient methods. For instance, a well-known cooperation scheme consists in using explorative methods, such as Evolutionary Algorithms, with intensification methods, such as Local Searches. In fact, many of the multi-objective hybrid approaches proposed in the literature deal with hybridization between genetic algorithms and Local Search. For example, the well-known Genetic Local Search (called also *memetic*) algorithms are popular in the multi-objective optimization community. Some examples are Jaszkiewicz (1998) and Ishibuchi and Murata (1998). The basic principle consists of incorporating the Local Search procedure during a genetic algorithm search. The Local Search part could be included by replacing, for example, the mutation operator, but it can also be added after each complete generation of the genetic algorithm. The classical structure of a Multi-Objective Genetic Local Search (MOGLS) algorithm is shown in Fig. 3, which depicts the relationships between the evolutionary multi-objective (EMO) component and the Local Search.

In the following, we analyze a number of works involving hybrid algorithms between Genetic Algorithms and Local Search. In Gen and Lin (2004), the “Multi-objective Hybrid Genetic Algorithm” is described. The paper focuses on how the performance of multi-objective Genetic Algorithms can be improved by hybridization with fuzzy logic control and Local Search. In Basseur et al. (2003), an Adaptive Genetic/Memetic Algorithm, called AGMA, is described. In

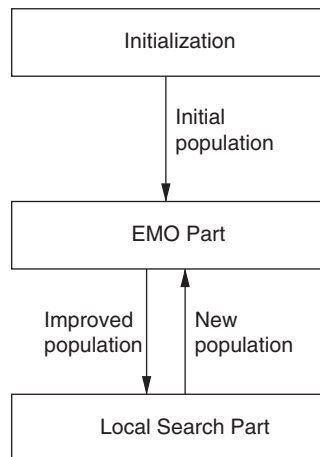


Fig. 3. Generic form for MOGLS algorithms.

this study, a Memetic Algorithm is hybridized with a Genetic Algorithm, and the transitions between the two metaheuristics are performed according to convergence speed criteria. In Deb and Goel (2001), the proposal consists of a sequential approach, where a genetic algorithm is executed and then a Local Search is applied to upgrade the solution quality and thus reduce the number of compromise solutions at the same time. This algorithm is applied on engineering shape design problems. In Jozefowicz et al. (2002), the hybrid approach proposed involves Genetic Algorithm and a Tabu Search using the Target Aiming Pareto Search principle (TAPaS), where search goals are defined according to the shape of the current non-dominated set of solutions.

Some works deal with hybrid techniques involving original multi-objective metaheuristics. In Chaiyaratana and Zalzalá (1999), a hybrid approach between neural networks and Genetic Algorithms for multi-objective time-optimal control optimization is introduced. In Burke et al. (2001), some evolutionary principles are introduced in a multi-objective Simulated Annealing algorithm to solve a bi-objective space allocation problem. In Basseur et al. (2005), a hybrid approach, combining Memetic Algorithm, Local Search, and Path Relinking, is applied to solve a bi-objective flow-shop scheduling problem.

Another recent popular issue is the cooperation between multi-objective metaheuristics and exact methods. There exists a growing number of studies involving this type of cooperation in the single-objective context, and two surveys have been published recently on this topic (Dumitrescu and Stützle, 2003; Puchinger and Raidl, 2005). These papers attempt to extract classical cooperation schemes between exact and metaheuristics approaches. For example, some hybrid schemes mainly aim to provide efficient solutions in a shorter time, while others primarily focus on achieving better solutions. In a multi-objective context, only a few studies tackle this type of approaches. In T'kindt et al. (2002), a bi-objective 2-machine flow-shop problem is solved. One objective is not NP-hard and then solved exactly, while the second one is solved using an Ant Colony algorithm. The objectives are treated in a lexicographic way.

In Basseur et al. (2004), Basseur and colleagues investigate several cooperative approaches for a bi-objective flow-shop problem. These schemes are designed around AGMA (presented above) and the two-phase method (Visée et al., 1998), a multi-objective exact method based on a Branch and Bound approach. The first cooperation described uses optimum solutions obtained by the metaheuristics as the initial bounds for the exact approach. Then, the search space explored by the exact method is reduced with respect to these bounds. This is a multi-objective application of classical cooperation found in the single-objective context. They also propose two heuristic cooperations, where the multi-objective exact part is running to intensify the search around the best solutions obtained by the metaheuristic. This search is carried out applying two different methods: large neighborhood techniques and partitioning methods. In Jozefowicz (2004), a bi-objective routing problem is solved using a combination of a Genetic Algorithm and a Branch and Cut algorithm. In this study, the Genetic Algorithm uses the Branch and Cut algorithm to exactly solve one of the two considered criteria.

Other interesting issues are the approaches called “hyper-heuristics”, first introduced by Burke et al. (2003a,b). Hyper-heuristics are dedicated to choosing the right heuristic for the right operation at the right time during the search. It must be noted that the hyper-heuristic operates in the heuristic space, as opposed to most implementations of metaheuristics, which operate in the solution space. This principle is relatively new, although the concept of “optimizing heuristics” is not a recent one. As an example, a multi-objective Tabu Search hyper-heuristic, which optimizes

the use of different neighborhoods in a Local Search algorithm, has been proposed by Burke et al. (2003b). This approach, tested on space allocation and timetabling, uses a Tabu list of neighborhood search heuristics that is actualized by adding the last used heuristic and/or the worst one, in terms of performance.

5. Parallel multi-objective optimization

When trying to solve real-world problems, multi-objective metaheuristics may not be powerful enough to provide a good approximation to the Pareto front in a reasonable time. Let us consider a MOP whose function evaluation requires 1 minute; then, to carry out 25,000 evaluations (a typical value in many experiments), around 17 days of computing time is necessary. This becomes worse if we consider that to evaluate the meta heuristic when solving the problem, a minimum of 30 independent runs need to be performed. In this situation, parallelism must be considered to tackle these kinds of MOPs.

Several taxonomies have been proposed to classify parallel implementations of metaheuristics (Crainic and Toulouse, 2003; Cung et al., 2003). Existing works review and discuss the general design and the main strategies used in their parallelization. A widely accepted classification mainly distinguishes between strategies whose goal is basically to speed up the sequential algorithm (*Single-walk parallelization*) and those that modify the behavior of the sequential implementation not only to search for higher speed but to hopefully improve the solution quality (*Multiple-walk parallelization*) (Cung et al., 2003).

This taxonomy has been adapted for multi-objective optimization algorithms in Nebro et al. (2005b), where a survey of works in the field is included. The adaptation to the multi-objective domain needs a further specification for two reasons. First, real-world MOPs have to deal with the utilization of complex solvers and simulators. We therefore differentiate those strategies aimed solely at speeding up the computations from those that parallelize the function evaluation of the problem to optimize, and from those that parallelize one or more operators of the search technique. Second, the results of a multi-objective optimization procedure are not restricted to finding a single solution, but a set of non-dominated solutions. This should be taken into account in the parallelization strategy because several parallel entities (threads or processes), at the same time, are exploring new potential solutions whose Pareto optimality must be checked. Here, we distinguish between two strategies: the Pareto front is distributed and locally managed by each search thread during the computation (local non-dominated solutions) or it is a centralized element of the algorithm (global non-dominated solutions). An outline of this hierarchical classification is shown in Fig. 4. Hence, we define the following categories:

1. *Single-walk parallelization*

This kind of parallelism is aimed at speeding up the computations, and the basic behavior of the underlying algorithms is not changed. It is the easiest and the most widely used parallelization in multi-objective optimization because the MOPs that are usually solved in this field are real-world problems involving high time-consuming tasks. Parallelism is applied in two ways:

- (a) *Parallel Function Evaluation* (PFE): The evaluations of the objective functions of MOPs are performed in parallel (Mäkinen et al., 1996; Golovkin et al., 2000; Radtke et al., 2003).

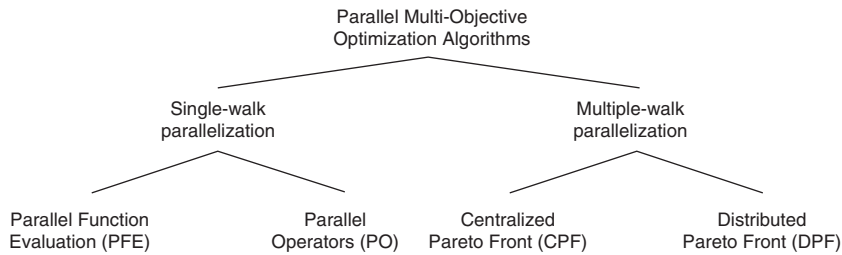


Fig. 4. Classification of parallel metaheuristics for multi-objective optimization.

(b) *Parallel Operator (PO)*: The search operators are run in parallel (Rowe et al., 1996; Watanabe et al., 2001).

2. Multiple-walk parallelization

Besides the search for speed up, improvements in the solution quality should also be sought in parallel implementations. Although the latter is likely to be the most important contribution of parallelism to metaheuristics (Cung et al., 2003), few such parallel search models have been especially designed for multi-objective optimization until recently (van Veldhuizen et al., 2003). A main issue in the development of these kinds of algorithms is how the Pareto front is built during the optimization process. Two different approaches can be considered:

- (a) *Centralized Pareto Front (CPF)*: The front is a centralized data structure of the algorithm that is built by the search threads during the entire computation. In this way, the new non-dominated solutions in the Pareto optimal set are global Pareto optima (Basseur et al., 2003; de Toro et al., 2004; Coello Coello and Reyes-Sierra, 2004).
- (b) *Distributed Pareto Front (DPF)*: The Pareto front is distributed among the search threads so that the algorithm works with local non-dominated solutions that must be somehow combined at the end of their work (Rowe et al., 1996; Duarte and Barán, 2001; Parsopoulos et al., 2004).

Among the studies analyzed in Nebro et al. (2005b), no pure CPF implementation has been found clearly motivated by efficiency issues. All the found CPF parallelizations are combined with DPF phases where local non-dominated solutions are considered. After each DPF phase, a single optimal Pareto front is built using these local Pareto optima. Then, the new Pareto front is again distributed for local computation, and so on.

Most works on parallel metaheuristic implementations deal with some kind of population-based metaheuristics, mainly evolutionary algorithms, as it seems natural to evolve a population of solutions on a population of machines. We next describe three major parallel models that are commonly applied in population-based algorithms (Alba and Tomassini, 2002): the master/slave model, the distributed or island model, and the cellular or the diffusion model. Although these models are comprised in the taxonomy explained before, we describe them because they have become very popular:

- *The master/slave model*: The evaluation step of a metaheuristic is generally the most time-consuming. Therefore, in order to speed up the search, the master–slave model distributes the

evaluation of the evolving population. As they require a global management of the population, the selection, transformation, and replacement operations are applied by a master process. At each generation, the master distributes the set of newly generated solutions between different slaves or workers, which evaluate and return these solutions as well as their fitness values. A particularly efficient execution is often obtained when the ratio between communication and computation is high. Otherwise, most of the time can be wasted in communications, leading to a poor parallel algorithm. The master/slave model fits into the single-walk parallelization category.

- *The distributed island model*: A number of metaheuristics are simultaneously deployed to cooperate with the aim of improving the quality of the solutions. Each of them performs a search on a sub-population; then, exchanges of genetic materials are performed in a synchronous or an asynchronous way to diversify the search into the target sub-populations. This allows a delay to the global convergence, especially when the metaheuristics are heterogeneous with respect to the variation operators. Individual migrations are conducted by various parameters and are performed in a regular or an irregular way. The distributed island model, along with the cellular model (see below), belongs to the multiple-walk parallelization category.
- *The cellular model*: This model uses the concept of small neighborhood in the sense that an individual may only interact with its nearby neighbors in the breeding loop (Tomassini, 2005). The overlapped small neighborhoods in cellular evolutionary algorithms help in exploring the search space because the induced slow diffusion of solutions through the population provides a kind of exploration, while exploitation takes place inside each neighborhood by evolutionary operators. This model is suited to massively parallel computers but, although such computers do not exist nowadays, it is still popular because of the competitive search capabilities of the resulting algorithms.

Although most works on parallel multi-objective metaheuristics are related to genetic algorithms, there are also proposals related to alternative methods, such as Tabu Search (Al-Yamani et al., 2002), Simulated Annealing (Agrafiotis, 2001; Chang and Huang, 1998), Ant Colony Systems (Delisle et al., 2001), and Memetic Algorithms (Basseur et al., 2003).

Parallel multi-objective metaheuristics are an open research area, where parallel models are needed to cope with the effective management of distributed Pareto fronts. Furthermore, the growing popularity of large-scale distributed systems based on the infra structure provided by the Internet, known as grid computing systems (Foster and Kesselman, 2003), leads to new perspectives to provide solutions to real-world problems whose complexity makes them intractable with typical distributed systems based on local area networks of workstations. There are some recent studies related to grid computing and multi-objective optimization, such as Luna et al. (2006), where the use of Globus (Foster and Kesselman, 1997), the *de facto* standard grid computing software, is analyzed to implement distributed multi-objective metaheuristics.

6. Pareto optimization under uncertainty

Real-world optimization problems are often subject to uncertainties caused by, for example, missing information in the problem domain or stochastic models. In this section, we are not

focused on fuzzy problems (there are currently many metaheuristic applications for this type of MOPs; Sakawa, 2001); in fact, we do not consider problems with variables having several fuzzy values, but variables that have a certain distribution of possible values in an interval (bounded or not). These uncertainties can take different forms in terms of distribution, bounds, and central tendency. In a general way, different types of uncertainty are distinguished. Generally speaking, uncertainties can be divided into four main categories. According to Jin and Branke (2005), these categories are:

1. Noisy fitness functions.
2. The design variables and/or the environmental parameters may change after the optimization process. Then, the quality of the solution obtained should be robust against environmental changes or deviations from the point(s).
3. The fitness function is approximated, which means that the fitness function suffers from approximation errors.
4. The optimum of the problem to be solved changes over time and, thus, the optimizer should be able to track the optimum continuously.

In all these cases, Jin and Branke note that evolutionary algorithms are able to work satisfactorily. In the next paragraphs, we analyze several multi-objective approaches proposed in the literature.

While uncertainty in the objective functions received some attention in the single-objective context (Arnold, 2003; Jin and Branke, 2005), only a few studies address this problem within a multiple-criteria setting. Let us note that the problem considered here is different from the issue of robustness, where the goal is to find solutions that are robust regarding parameter perturbations. The work of Horn and Nafpliotis (1993) was among the first to discuss uncertainty in the light of generating methods, although they did not propose a particular multi-objective optimizer for this purpose.

Several years later, Hughes (2001) and Teich (2001) independently proposed stochastic extensions of Pareto dominance and suggested similar ways to integrate probabilistic dominance in the fitness assignment procedure; both studies consider special types of probability distributions. More precisely, Teich proposes an approach modifying the SPEA algorithm, assuming that the probability density function is constant over the property interval of each random variable. First, he defines a probability dominance notion in the single-objective case as follows: given two points a and b with objectives $a \in [a^s, \dots, a^u]$ and $b \in [b^s, \dots, b^u]$, respectively, the probability of a to dominate b , written as $P[a \succ b]$ for uniform distribution functions, is given as:

$$P[a \succ b] = \begin{cases} 0 & \text{if } b^u < a^s, \\ 1 & \text{if } a^u < b^s, \\ \frac{1}{a^u - a^s} \left(\int_{y=a^s}^{b^s} dy + \int_{y=\max(a^s, b^s)}^{\min(a^u, b^u)} \left(1 - \frac{y - b^s}{b^u - b^s} \right) dy \right) & \text{otherwise.} \end{cases}$$

Based on the single-objective case, he proposes an extension to the multi-objective case. For any two n -dimensional decision vectors a and b , and m statistically independent objective functions f_1, f_2, \dots, f_m ,

$$P[a \succ b] = \prod_{i=1}^m P[f_i(a) \leq f_i(b)].$$

The SPEA ranking computation is adapted according to these probability values (N : population size, M_t : population at time t):

$$R(i) = \frac{1}{N-1} \sum_{j \in M_t; j \neq i} P[m(j) \succ m(i)].$$

It is also necessary to redefine the calculation of distance between two solutions. He uses the expected objective values of the solutions to evaluate the distance.

Hughes (2001) proposes the same type of approach, with the probabilistic dominance notion. He compares the use of probabilistic ranking against classical ranking algorithms, such as NSGA and MOGA. The probabilistic ranking proposed is the following:

$$R_i = \sum_{j=1}^N P[a \succ b] + \frac{1}{2} \cdot \sum_{j=1}^N P[a \equiv b] - 0.5.$$

Teich and Hughes' studies were among the first to propose approaches dedicated to establish new concepts for multi-objective optimization under uncertainty. But, with these studies, there are still several open problems: the uncertainties considered are completely defined (uniform/normal distribution, bounded, central tendency equal to 0), the proposed approaches are not directly adaptable to unknown uncertainties, and the proposed algorithms need to compute the distance between solutions, which is dependent on uncertainties.

Bui et al. (2005) use this same principle, but they introduce fitness inheritance in the Genetic Algorithm, which replaces the multiple evaluation with respect to a criterion of confidence.

In Büche et al. (2001), Büche and colleagues are critical about the redefinition of Pareto dominance approaches in the case of non-bounded noise, i.e., potential aberrant solutions, which confer a large deviation to the algorithm running. They propose three modifications for an extended multi-objective algorithm to overcome the problem of noise:

- **Domination-dependent lifetime:** in contrast to elitism, which may preserve elitist (non-dominated) solutions for an infinite time, a maximal lifetime is assigned to each individual. They propose to adapt the lifetime of each individual according to the dominance relation. The lifetime is shortened if the solution dominates a major part of the present non-dominated solutions. This limits the impact of a solution.
- **Re-evaluation of solutions:** all non-dominated solutions whose lifetime has expired are re-evaluated and added to the population. This enables good solutions to remain in the evolutionary process, but their objective values will change due to the noise in the re-evaluation.
- **Extended update of the secondary population:** the elite is updated only according to the non-expired lifetime solutions.

In Babbar et al. (2003), another ranking method is proposed, and it is based on the average value per objective and the variance of the set of evaluations. Similarly, Deb and Gupta (2005) suggested considering for each dimension the mean over a given sample of objective vectors and applying standard multi-objective optimizers for deterministic objective functions.

Most of the existing studies assume certain characteristics (symmetry, shape, etc.) of the probability distribution that determines to which objective vectors a solution may be mapped. In

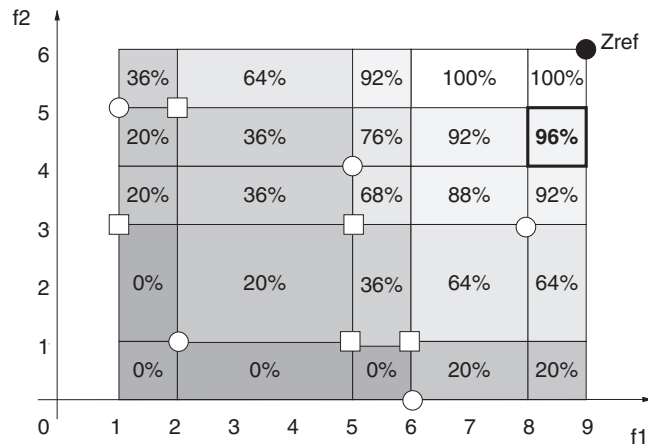


Fig. 5. Example with two objective functions and two solutions evaluated five times (circle and square points). There exists only one alternative over 25 for the point (8, 4) to not be dominated by at least as one solution: if we take the circle and square sample on the top of the figure. Then $P(S \preceq (8, 4)) = 24/25 = 96\%$.

other words, the corresponding methods rely on and exploit problem knowledge, which may not be available, particularly with real-world applications. Basseur and Zitzler (2006) depict a slightly different scenario, where the optimization goal is specified in terms of a quality indicator, as in Zitzler and Künzli (2004). A general indicator model that can handle any type of distribution representing the uncertainty allows different distributions for different solutions, and does not assume a “true” objective vector per solution, but in general regards a solution to be inherently associated with an unknown probability distribution in the objective space. Then, the solutions are evaluated several times, and the set of evaluations obtained corresponds to approximations of the “true” probability distribution of this solution in the objective space. The solutions are evaluated according to a quality indicator value, which is also computed in a stochastic way. They also propose an algorithm to compute an empirical attainment function, which evaluates the area of the objective space that is dominated by the output with different confidence levels (see Fig. 5).

An interesting analysis of multi-objective optimization under uncertainty can be found in Tan and Goh (2008), where the impact of different type of uncertainties is described and the approaches designed for better and robust EMO performance are presented. To summarize, a small, but increasing number of studies are dedicated to solve uncertain MOPs, but this aspect of multi-objective optimization needs to be further explored. Particularly, some performance assessment indicators are needed in order to evaluate the effectiveness of different algorithms, and to evaluate the effect of the mechanism that takes into account the uncertainty during the optimization process.

7. Conclusions and perspectives

Multi-objective optimization is certainly a crucial research area for engineering and research science because many real-world problems are of a multi-objective nature. The most popular approach nowadays to tackle these problems is the use of metaheuristics, and especially Evolutionary Algorithms.

In this paper, we have analyzed a number of emergent metaheuristic techniques to solve MOPs. Our purpose has been to introduce open research lines related to metaheuristics but with a focus on less explored areas that are of growing interest. We have focused on alternative metaheuristic techniques, hybrid methods, parallel metaheuristics, and multi-objective optimization under uncertainty.

The analysis of these techniques reveals that, although there are a number of works related to them, they have not been fully explored and there are many open research lines. We expect this paper to be useful, in particular to those researchers looking for new lines of work in the field of metaheuristics for multi-objective optimization and, in general to the multi-objective research community.

References

- Agrafiotis, D.K., 2001 Multiobjective optimization of combinatorial libraries. *IBM Journal of Research and Development* 45, 545–566.
- Al-Yamani, A., Sait, S., Youssef, H., 2002. Parallelizing tabu search on a cluster of heterogeneous workstations. *Journal of Heuristics* 8, 3, 277–304.
- Alba, E., Tomassini, M., 2002. Parallelism and evolutionary algorithms. *IEEE Transitional on Evolutionary Computation* 6, 5, 443–462.
- Arnold, D.V., 2003. A comparison of evolution strategies with other direct search methods in the presence of noise. *Computational Optimization and Applications* 24, 135–159.
- Babbar, M., Lakshmikantha, A., Goldberg, D.E., 2003. A modified NSGA-II to solve noisy multiobjective problems. In Cantü-Paz, E., et al. (eds) *Genetic and Evolutionary Computation Conference (GECCO'2003), late breaking papers, volume 2723 of Lecture Notes in Computer Science*. Springer, Chicago, IL, pp. 21–27.
- Basseur, M., Burke, E.K., 2007. Indicator-based multiobjective local search. In *IEEE Congress on Evolutionary Computation (CEC 2007)*, Singapore, September, pp. 3100–3107.
- Basseur, M., Lemesre, J., Talbi, E.-G., Dhaenens, C., 2004. Cooperation between branch and bound and evolutionary approaches to solve a biobjective flow shop problem. In *Workshop on Evolutionary Algorithms (WEA'04)*, volume 3059, pp. 72–86.
- Basseur, M., Seynhaeve, F., Talbi, E.-G., 2003. Adaptive mechanisms for multi-objective evolutionary algorithms. In *Congress on Engineering in System Application CESA'03*, Lille, France, pp. 72–86.
- Basseur, M., Seynhaeve, F., Talbi, E.-G., 2005. Path relinking in Pareto multi-objective genetic algorithms. In Coello Coello, C.A., Aguirre, A.H., Zitzler, E. (eds) *Evolutionary Multi-Criterion Optimization, EMO'2005, volume 3410 of Lecture Notes in Computer Science*. Springer-Verlag, Guanajuato, Mexico, pp. 120–134.
- Basseur, M., Zitzler, E., 2006. Handling uncertainty in indicator-based multiobjective optimization. *International Journal of Computational Intelligence Research (IJCIR)* 2, 3, 255–272.
- Beausoleil, R.P., 2001. Multiple criteria scatter search. In *4th Metaheuristics International Conference (MIC'01)*, Porto, Portugal, pp. 539–544.
- Beausoleil, R.P., 2006. “MOSS”, multiobjective scatter search applied to non-linear multiple criteria optimization. *European Journal of Operational Research* 169, 2, 426–449.
- Beausoleil, R., Baldoquin, G., Montejo, R., 2008. Multi-start and path relinking methods to deal with multiobjective knapsack problems. *Annals of Operations Research* 157, 1, 105–133.
- Blum, C., Roli, A., 2003. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys* 35, 3, 268–308.
- Büche, D., Stoll, P., Koumoutsakos, P., 2001. An evolutionary algorithm for multi-objective optimization of combustion processes. Technical report, Center for Turbulence Research, Annual research briefs.
- Bui, L.T., Abbass, H.A., Essam, D., 2005. Fitness inheritance for noisy evolutionary multi-objective optimization. In Beyer, H.-G., O'Reilly, U.-M. (eds) *Proceedings of the Genetic and Evolutionary Computation Conference GECCO '05*, ACM, Washington, DC, June, pp. 779–785.

- Burke, E., Cowling, P., Landa Silva, J., Petrovic, S., 2001. Combining hybrid metaheuristics and populations for the multiobjective optimisation of space allocation problems. In *Proceedings of GECCO 2001*, Morgan Kaufmann, San Francisco, July, pp. 1252–1259.
- Burke, E.K., Kendall, G., Newall, J., Hart, E., Ross, P., Schulemburg, S., 2003a. *Handbook of Metaheuristics, chapter Hyper-heuristics: An Emerging Direction in Modern Search Technology*. Kluwer Academic Publishers, Boston, MA.
- Burke, E.K., Landa Silva, J.D., Soubeiga, E., 2003b. Hyperheuristic approaches for multiobjective optimisation. In *5th Metaheuristics International Conference (MIC 2003)*, Kyoto, Japan, August.
- Cantú-Paz, E., 2000. *Efficient and Accurate Parallel Genetic Algorithms*. Kluwer Academic Publishers, Boston, MA.
- Carraway, R.L., Morin, T.L., Moskowitz, H., 1990. Generalized dynamic programming for multicriteria optimization. *European Journal of Operational Research* 44, 95–104.
- Chaiyaratana, N., Zalzal, A., 1999. Hybridisation of neural networks and genetic algorithms for time-optimal control. In *Congress on Evolutionary Computation (CEC'99)*, volume 1, IEEE Service Center, July, pp. 389–396.
- Chang, C.S., Huang, J.S., 1998. Optimal multiobjective SVC planning for voltage stability enhancement. *IEEE Proceedings on Generation, Transmission and Distribution* 145, 2, 203–209.
- Chipperfield, A.J., Whidborne, J.F., Fleming, P.J., 1999. *Evolutionary Algorithms and Simulated Annealing for MCDM, chapter 16*. Kluwer Academic Publishing, Boston, MA.
- Coello Coello, C.A., Reyes-Sierra, M., 2004. A Study of the Parallelization of a Coevolutionary Multi-Objective Evolutionary Algorithm. In *MICAI 2004*, LNAI 2972, pp. 688–697.
- Coello Coello, C.A., Van Veldhuizen, D.A., Lamont, G.B., 2002. *Evolutionary Algorithms for Solving Multi-Objective Problems. Genetic Algorithms and Evolutionary Computation*. Kluwer Academic Publishers, Boston, MA.
- Corberán, A., Fernández, E., Laguna, M., Martí, R., 2002. Heuristic solutions to the problem of routing school buses with multiple objectives. *Journal of the Operational Research Society* 53, 4, 427–435.
- Crainic, T.G., Toulouse, M., 2003. Parallel strategies for metaheuristics. In Glover, F.W., Kochenberger, G.A. (eds) *Handbook of Metaheuristics*, Kluwer Academic Publishers, Boston, MA.
- Cung, V.-D., Martins, S.L., Ribeiro, C.C., Roucairol, C., 2003. Strategies for the parallel implementation of metaheuristics. In Ribeiro, C.C., Hansen, P. (eds) *Essays and Surveys in Metaheuristics*. Kluwer, Boston, MA, pp. 263–308.
- Czyzack, P., Jaszkievicz, A., 1998. Pareto simulated annealing – a metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis* 7, 1, 34–47.
- de Toro, F., Ortega, J., Ros, E., Mota, S., Paechter, B., Martín, J.M., 2004. PSFGA: Parallel processing and evolutionary computation for multiobjective optimisation. *Parallel Computing* 30, 5–6, 721–739.
- Deb, K., 2001. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, New York.
- Deb, K., Goel, T., 2001. A hybrid multi-objective evolutionary approach to engineering shape design. In Zitzler, E., Deb, K., Thiele, L., Coello Coello, C., Corne, D. (eds) *First International Conference on Evolutionary Multi-Criterion Optimization, volume 1993 of Lecture Notes in Computer Science*. Zurich, Switzerland, pp. 385–399.
- Deb, K., Gupta, H., 2005. Searching for robust Pareto-optimal solutions in multi-objective optimization. In Coello Coello, C.A., Aguirre, A.H., Zitzler, E. (eds) *Conference on Evolutionary Multi-Criterion Optimization (EMO'05), volume 3410 of Lecture Notes in Computer Science (LNCS)*. Springer-Verlag, Guanajuato, Mexico, pp. 150–164.
- Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6, 2, 182–197.
- Delisle, P., Krajecki, M., Gravel, M., Gagn, C., 2001. Parallel implementation of an ant colony optimization metaheuristic with OpenMP. In *3rd European workshop on OpenMP (EWOMP'01)*, pp. 8–12.
- Doerner, K., Gutjahr, W., Hartl, R., Strauss, C., 2002. Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection. In *Proceedings of the 4th Metaheuristics International Conference*, Porto, pp. 243–248.
- Dorigo, M., Blum, C., 2005. Ant colony optimization theory: A survey. *Theoretical Computer Science* 344, 2–3, 243–278.
- Duarte, S., Barán, B., 2001. Multiobjective Network Design Optimisation Using Parallel Evolutionary Algorithms. In *XXVII Conferencia Latinoamericana de Informática CLEI'2001*.

- Dumitrescu, I., Stützle, T., 2003. Combinations of local search and exact algorithms. In Raidl, G., et al (eds) *Applications of Evolutionary Computing, Proceedings of EvoWorkshops 2003, Number 2611 in Lecture Notes in Computer Science*. Springer Verlag, Berlin, pp. 211–224.
- Foster, I., Kesselman, C., 1997. Globus: A metacomputing infrastructure toolkit. *International Journal of Supercomputer Applications and High Performance Computing* 11, 2, 115–128.
- Foster, I., Kesselman, C., 2003. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, Burlington, MA.
- Gambardella, L.M., Taillard, E., Agazzi, G., 1999. *MACS-VRPTW: a multiple ant colony system for vehicle routing problems with time windows*. McGraw-Hill, Maidenhead, pp. 63–76.
- Gandibleux, X., Ehrgott, M., 2005. 1984–2004 20 years of multiobjective metaheuristics. But what about the solution of combinatorial problems with multiple objectives? In Hernández Aguirre, A., Coello Coello, C., Zitzler, E. (eds) *Evolutionary Multi-Criterion Optimization: Third International Conference, EMO 2005*. Springer-Verlag, Guanajuato, Mexico, pp. 33–46.
- Gandibleux, X., Mezdaoui, N., Freville, A., Caballero, R., Ruiz, F., Steuer, R., 1997. A tabu search procedure to solve multiobjective combinatorial optimization problems. In *Advances in Multiple Objective and Goal Programming*, volume 455 of *Lecture Notes in Economics and Mathematical Systems*, pp. 291–300.
- Gen, M., Lin, L., 2004. Multiobjective hybrid genetic algorithm for bicriteria network design problem. In *The 8th Asia Pacific Symposium on Intelligent and Evolutionary Systems*, Cairns, Australia, December, pp. 73–82.
- Glover, F., Laguna, M., Martí, R., 2000. Fundamentals of scatter search and path relinking. *Control and Cybernetics* 29, 3, 653–684.
- Glover, F., Laguna, M., Martí, R., 2003. Scatter search. In Ghosh, A., Tsutsui, S. (eds) *Advances in Evolutionary Computing: Theory and Applications*. Springer, Berlin.
- Golovkin, I.E., Mancini, R.C., Louis, S.J., 2000. Parallel Implementation of Niche Pareto Genetic Algorithm Code for X-Ray Plasma Spectroscopy. In *Late-Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference*.
- Hansen, M.P., 1997. Tabu Search in Multiobjective Optimisation: MOTS. In *Proceedings of the 13th International Conference on Multiple Criteria Decision Making (MCDM'97)*, Cape Town, South Africa.
- Hansen, M.P., 2000. 1997 Tabu search for multiobjective combinatorial optimization: Tamoco. *Control and Cybernetics* 29, 3, 799–818.
- Herrera, F., Lozano, M., Molina, D., 2006. Continuous scatter search: An analysis of the integration of some combination methods and improvement strategies. *European Journal of Operational Research* 169, 450–476.
- Hertz, A., Jaumard, B., Ribeiro, C.C., Formosinho Filho, W.P., 1994. A multi-criteria tabu search approach to cell formation problems in group technology with multiple objectives. *RAIRO Recherche Opérationnelle/Operations Research* 28, 3, 303–328.
- Horn, J., Nafpliotis, N., 1993. Multiobjective optimization using the niched Pareto genetic algorithm. IlliGAL Report 93005, Illinois Genetic Algorithm Laboratory, University of Illinois at Urbana-Champaign, IL.
- Hughes, E., 2001. Evolutionary multi-objective ranking with uncertainty and noise. In *EMO'01: Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, Springer-Verlag, London, pp. 329–343.
- Iredi, S., Merkle, D., Middendorf, M., 2001. Bi-criterion optimization with multi colony ant algorithms. In *Conference on Evolutionary Multi-Criterion Optimization (EMO'01)*, volume 1993 of *Lecture Notes in Computer Science (LNCS)*, March, pp. 358–372.
- Ishibuchi, H., Murata, T., 1998. Multi-objective genetic local search algorithm and its application to flowshop scheduling. *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews* 28, 3, 392–403.
- Janson, S., Merkle, D., Middendorf, M., 2008. Molecular docking with multi-objective particle swarm optimization. *Applied Soft Computing* 8, 1, 666–675.
- Jaszkiewicz, A., 1998. Genetic local search for multiple objective combinatorial optimization. Technical Report RA-014/98, Institute of Computing Science, Poznan University of Technology.
- Jaszkiewicz, J., 2005. Path relinking for multiple objective combinatorial optimization. TSP case study. In *The 16th Mini - EURO Conference and 10th Meeting of EWGT (Euro Working Group Transportation)*.

- Jin, Y., Branke, J., 2005. Evolutionary optimization in uncertain environments – a survey. *IEEE Transactions on Evolutionary Computation* 9, 3, 303–317.
- Jozefowicz, N., 2004. *Modélisation et résolution approchée de problèmes de tournées multi-objectif*. PhD thesis, University of Lille, Lille, France, December.
- Jozefowicz, N., Semet, F., Talbi, E.-G., 2002. Parallel and hybrid models for multi-objective optimization: Application to the vehicle routing problem. In Guervos, J., Adamidis, P., Beyer, H.-G., Fernández-Villacanas, J.-L., Schwefel, H.-P. (eds) *Parallel Problem Solving from Nature (PPSN VII), Number 2439 in Lecture Notes in Computer Science*. Springer-Verlag, Granada, Spain, pp. 271–280.
- Karasakal, E.K., Köksalan, M., 2000. A simulated annealing approach to bicriteria scheduling problems on a single machine. *Journal of Heuristics* 6, 3, 311–327.
- Kennedy, J., Eberhart, R.C., 1995. Particle swarm optimization. In *IEEE International Conference on Neural Networks*, Piscataway, NJ, pp. 1942–1948.
- Knowles, J., Corne, D., 1999. The Pareto archived evolution strategy: A new baseline algorithm for multiobjective optimization. In *Proceedings of the 1999 Congress on Evolutionary Computation*, IEEE Press, Piscataway, NJ, pp. 9–105.
- Li, X., 2004. Better spread and convergence: Particle swarm multiobjective optimization using the maximin fitness function. In Deb, K., et al. (eds) *Genetic and Evolutionary Computation Conference (GECCO 2004), Number 3102 in Lecture Notes in Computer Science*. Springer-Verlag, Seattle, WA.
- Liefooghe, A., Mesmoudi, S., Humeau, J., Jourdan, L., Talbi, E.-G., 2009. A study on dominance-based local search approaches for multiobjective combinatorial optimization. In *Second International Workshop on Engineering Stochastic Local Search Algorithms: Designing, Implementing and Analyzing Effective Heuristics (SLS 2009)*, volume 5752 of *Lecture Notes in Computer Science*, Brussels, Belgium, September, pp. 120–124.
- López-Ibáñez, M., Stützle, T., 2010. Automatic configuration of multi-objective ACO algorithms. In *Seventh International Conference on Swarm Intelligence (ANTS'2010)*, pp. 95–106.
- Luna, F., Nebro, A.J., Alba, E., 2006. Observations in using grid-enabled technologies for solving multi-objective optimization problems. *Parallel Computing* 32, 4–6, 377–393.
- Mäkinen, R.A.E., Neittaanmäki, P., Periaux, J., Sefrioui, M., Toivanen, J., 1996. Parallel Genetic Solution for Multiojective MDO. In *Parallel CFD'96 Conference*, pp. 352–359.
- Mariano, C.E., Morales, E., 1999. MOAQ and ANT-Q algorithm for multiple objective optimization problems. In *Proceedings of the Genetic and Evolutionary Computation Conference*, Orlando, FL, July, pp. 894–901.
- Molina, J., Laguna, M., Martí, R., Caballero, R., 2007. SSPMO: A scatter search procedure for non-linear multiobjective optimization. *INFORMS Journal on Computing* 19, 1, 91–100.
- Nebro, A.J., Luna, F., Alba, E., 2005a. New ideas in applying scatter search to multiobjective optimization. In Coello Coello, C.A., Hernandez, H., Zitzler, E. (eds) *EMO 2005*, volume 3410 of *Lecture Notes in Computer Science*, pp. 443–458.
- Nebro, A.J., Luna, F., Alba, E., Beham, A., Dorronsoro, B., 2006. AbYSS: Adapting Scatter Search for Multiobjective Optimization. Technical Report ITI-2006-2, Departamento de Lenguajes y Ciencias de la Computación, University of Málaga, E.T.S.I. Informática, Campus de Teatinos.
- Nebro, A.J., Luna, F., Talbi, E.-G., Alba, E., 2005b. Parallel multiobjective optimization. In Alba, E. (ed) *Parallel Metaheuristics*. Wiley, New York, pp. 371–394.
- Ono, S., Nakayama, S., 2009. Multi-objective particle swarm optimization for robust optimization and its hybridization with gradient search. In *IEEE Congress on Evolutionary Computation, CEC '09*, pp. 1629–1636.
- Pareto, V., 1896. *Cours d'Economie Politique*. Rouge, Lausanne, Switzerland.
- Parsopoulos, K.E., Tasoulis, D.K., Pavlidis, N.G., Plagianakos, V.P., Vrahatis, M.N., 2004. Vector Evaluated Differential Evolution for Multiobjective Optimization. In *Proceedings of the IEEE 2004 Congress on Evolutionary Computation (CEC 2004)*.
- Pasia, J., Aguirre, H., Tanaka, K., 2010. Path relinking on many-objective nk-landscapes. In *Parallel Problem Solving from Nature PPSN XI*, volume 6238 of *Lecture Notes in Computer Science*, pp. 677–686.
- Puchinger, J., Raidl, G.R., 2005. Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. In *Proceedings of the First International Work-Conference on the Interplay Between Natural and Artificial Computation*, volume 3562 of *Lecture Notes in Computer Science*, Springer, pp. 41–53.

- Radtke, P.W.W., Oliveira, L.S., Sabouring, R., Wong, T., 2003. Intelligent Zoning Design Using Multi-Objective Evolutionary Algorithms. In *Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)*, pp. 824–828.
- Reyes-Sierra, M., Coello Coello, C.A., 2005. Improving PSO-based multi-objective optimization using crowding, mutation and ϵ -dominance. In Coello Coello, C.A., Hernandez, H., Zitzler, E. (eds) *Third International Conference on Evolutionary Multi-Criterion Optimization, EMO 2005*, volume 3410 of *Lecture Notes in Computer Science*, Springer, pp. 505–519.
- Reyes-Sierra, M., Coello Coello, C.A., 2006. Multi-objective particle swarm optimizers: A survey of the state-of-the-art. *International Journal of Computational Intelligence Research* 2, 287–308.
- Rowe, J., Vinsen, K., Marvin, N., 1996. Parallel GAs for Multiobjective Functions. In *Proceedings of the 2nd Nordic Workshop on Genetic Algorithms and Their Applications (2NWGA)*, pp. 61–70.
- Sakawa, M., 2001. *Genetic Algorithms and Fuzzy Multiobjective Optimization*. Operations Research/Computer Science Interfaces Series. Springer.
- Schaffer, J.D., 1985. Multiple objective optimization with vector evaluated genetic algorithms. In Grefenstette, J.J. (ed.) *ICGA International Conference on Genetic Algorithms*. Lawrence Erlbaum, Mahwah, NJ, pp. 93–100.
- Sen, T., Raiszadeh, M.E., Dileepan, P., 1988. A branch and bound approach to the bicriterion scheduling problem involving total flowtime and range of lateness. *Management Science* 34, 2, 254–260.
- Serafini, P., 1992. Simulated annealing for multiple objective optimization problems. In *Proceedings of the Tenth International Conference on Multiple Criteria Decision Making*, volume 1, Taipei, Taiwan, pp. 87–96.
- Smith, K., Everson, R., Fieldsend, J., Murphy, C., Misra, R., 2008. Dominance-based multiobjective simulated annealing. *IEEE Transactions on Evolutionary Computation* 12, 3, 323–342.
- Stewart, B.S., White, C.C., 1991. Multiobjective A*. *Journal of the ACM* 38, 4, 775–814.
- Talbi, E.-G., 2002. A taxonomy of hybrid metaheuristics. *Journal of Heuristics* 8, 541–564.
- Tan, K.C., Goh, C.K., 2008. Handling uncertainties in evolutionary multi-objective optimization. In *Proceedings of the 2008 IEEE world conference on Computational intelligence: research frontiers, WCCI'08*, Springer-Verlag, Berlin, pp. 262–292.
- Teich, J., 2001. Pareto-front exploration with uncertain objectives. In *Conference on Evolutionary Multi-Criterion Optimization (EMO'01)*, volume 1993 of *Lecture Notes in Computer Science (LNCS)*, March, pp. 314–328.
- T'kindt, V., Monmarché, N., Tercinet, F., Laugt, D., 2002. An ant colony optimization algorithm to solve a 2-machine bicriteria flowshop scheduling problem. *European Journal of Operational Research* 142, 2, 250–257.
- Tomassini, M., 2005. *Spatially Structured Evolutionary Algorithms: Artificial Evolution in Space and Time*. *Natural Computing Series*. Springer-Verlag, Heidelberg.
- Ulungu, E.L., Teghem, J., Fortemps, P., Tuytens, D., 1999. MOSA method: A tool for solving multiobjective combinatorial optimization problems. *Journal of Multi-Criteria Decision Analysis* 8, 4, 221–236.
- van Veldhuizen, D.A., Zydallis, J.B., Lamont, G.B., 2003. Considerations in engineering parallel multiobjective evolutionary algorithms. *IEEE Trans. Evolutionary Computation* 7, 2, 144–173.
- Visée, M., Teghem, J., Pirlot, M., Ulungu, E.L., 1998. Two-phases method and branch and bound procedures to solve knapsack problem. *Journal of Global Optimization* 12, 139–155.
- Watanabe, S., Hiroyasu, T., Miki, M., 2001. Parallel Evolutionary Multi-Criterion Optimization for Mobile Telecommunication Networks Optimization. In *Proceedings of the EUROGEN' 2001*, pp. 167–172.
- Zitzler, E., Künzli, S., 2004. Indicator-based selection in multiobjective search. In *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, Birmingham, UK, September, pp. 832–842.
- Zitzler, E., Laumanns, M., Thiele, L., 2001. SPEA2: Improving the strength Pareto evolutionary algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland.