Top

Volume 12, Number 1, 1-90
June 2004

REPRINT

M. Ehrgott and X. Gandibleux

Approximative Solution Methods for Multiobjective Combinatorial Optimization

Published by
Sociedad de Estadística e Investigación Operativa
Madrid, Spain

Top

Volume 12, Number 1
June 2004

Published by
Sociedad de Estadística e Investigación Operativa
Madrid, Spain

Sociedad de Estadística e Investigación Operativa
Top (2004) Vol. 12, No. 1, pp. 1–89

Approximative Solution Methods for Multiobjective Combinatorial Optimization

Matthias Ehrgott
Department of Engineering Science
University of Auckland
Private Bag 92019, Auckland – New Zealand
E-mail: M.Ehrgott@auckland.ac.nz

Xavier Gandibleux
LAMIH – UMR CNRS 8530
University of Valenciennes, Campus "Le Mont Houy"
F-59313 Valenciennes cedex 9 – France
E-mail: Xavier.Gandibleux@univ-valenciennes.fr

Abstract

In this paper we present a review of approximative solution methods, that is, heuristics and metaheuristics designed for the solution of multiobjective combinatorial optimization problems (MOCO). First, we discuss questions related to approximation in this context, such as performance ratios, bounds, and quality measures. We give some examples of heuristics proposed for the solution of MOCO problems. The main part of the paper covers metaheuristics and more precisely non-evolutionary methods. The pioneering methods and their derivatives are described in a unified way. We provide an algorithmic presentation of each of the methods together with examples of applications, extensions, and a bibliographic note. Finally, we outline trends in this area.

Key Words: Multiobjective optimization, combinatorial optimization, heuristics, metaheuristics, approximation.

AMS subject classification: 90C29, 90C27, 90C59.

1 Introduction

The last two decades have seen the development and the improvement of approximative solution methods – usually called "heuristics and metaheuristics" (Osman and Laporte (1996)). The success of metaheuristics, e.g., simulated annealing, tabu search, genetic algorithms, on hard single

The research of M. Ehrgott has been partially supported by University of Auckland grant 3602178/9275 and grant Ka 477/27-1 of the Deutsche Forschungsgemeinschaft (DFG).

objective optimization problems is well recognized today. In an optimization context the term heuristic is used in contrast to methods that guarantee to find a global optimum such as, e.g., the "Hungarian Method" for solving the assignment problem or implicit enumeration schemes such as branch and bound and dynamic programming.

Although combinatorial optimization models have been successfully used in a vast number of applications, these models often neglect the fact that many real-life problems require taking into account several conflicting points of view corresponding to multiple objectives. Here are some examples:

- In portfolio optimization risk and return are the criteria that have generally been considered. Recently the classical Markowitz model has been criticized and other criteria have been mentioned: ratings by agencies, dividend, long-term performance, etc., see, e.g., Ehrgott et al. (2004).
- In airline operations, scheduling technical and cabin crew has a major effect on cost and small percentage improvements may translate to multi-million dollar savings. However, cost is not the only concern in airline operations. Robust solutions are desired, which avoid the propagation of delays due to crew changing aircraft, see Ehrgott and Ryan (2002).
- In railway transportation, the planning of railway network infrastructure capacity has the goals of maximizing the number of trains that can use the infrastructure element (e.g. a station) and to maximize robustness of the solution to disruptions in operation. This problem can be modelled as a large scale set packing problem with two objectives, Delorme et al. (2003).
- In radiation therapy planning for cancer treatment conflicting goals are to achieve a high dose level in the tumour whereas the dose absorbed by healthy tissue is to be limited. For anatomical and physical reasons these objectives cannot be achieved simultaneously. A multicriteria model is described in Küfer et al. (2003).
- In computer networks, internet traffic routing may be enhanced if based on a multi-objective routing procedure to prevent network congestion. Multi-objective shortest paths between one router and all the

other routers of the network must be computed in real-time, by simultaneously optimizing linear objectives (cost, delay) and bottleneck ones (quality, bandwidth), see Randriamasy et al. (2002).

In combinatorial optimization the consideration of multiple objectives has received attention for about the last two decades but multiobjective combinatorial optimization (MOCO) has become a very active area of research only since the 1990's as shown in the bibliography by Ehrgott and Gandibleux (2000). Multiple objective (combinatorial) optimization differs from traditional single objective optimization in several ways:

- The usual meaning of the optimum makes no sense in the multiple objective case because a solution optimizing all objectives simultaneously does in general not exist. Instead, a search is launched for a feasible solution yielding the best compromise among objectives on a set of so-called efficient (Pareto optimal, non-dominated) solutions.
- The identification of a best compromise solution requires the preferences expressed by the decision maker to be taken into account.
- The multiple objectives encountered in real-life problems can often be expressed as mathematical functions of a variety of forms. I.e., not only do we deal with conflicting objectives, but with objectives of different structures.
- The multiple objectives add to the difficulty of combinatorial optimization problems so that MOCO problems are very hard to solve exactly (Ehrgott (2000)), even if they are derived from easy single objective optimization problems.

With increasing interest in multiobjective models and the difficulties encountered when solving multiobjective optimization problems, interest in approximative methods for solving multiobjective optimization problems arose. However, this interest is relatively recent. It began with the work of Schaffer (1984) on multiple objective genetic algorithms. The work of Serafini (1992) started a stream of research on multiple objective extensions of local search based metaheuristics. Since then research on these methods has mushroomed, giving birth to many ideas.

In the broad area of optimization, multiobjective approximation methods attract a growing community of researchers, witnessed by a growing number of publications. Several PhD and habilitation theses dealing specifically with these methods have been written, Coello (1996), Mira de Fonseca (1995), El-Sherbeny (2001), Godart (2001), Hansen (1998), Jaszkiewicz (2001c), Schaffer (1984), Ulungu (1993), and Zitzler (1999). Recently, international publications have dedicated special issues to these methods (Journal of Heuristics 6(3), 2000; Foundations of Computing and Decision Sciences 25(4), 2000 and 26(1), 2001; Lecture Notes in Economics and Mathematical Systems 535, 2004).

Numerous methods are inspired by Evolutionary Multiobjective Optimization (EMO). To give an idea about the strong activity in this field we remark that a repository dedicated to EMO, Coello (2004), counts more than 1600 entries. Lately four books dedicated to EMO have been published, Bagchi (1999), Coello et al. (2001), Deb (2001), Osyczka (2001), as well as the proceedings of the EMO international conferences, Fonseca et al. (2003), Zitzler et al. (2001). In addition, several surveys have been published regularly over the last 10 years, e.g. Coello (1999), Coello (2000), Fonseca and Fleming (1995), and Jones et al. (2002).

In this paper we present a review of approximation methods for solving multiobjective combinatorial optimization problems (MOCO), considering especially non-evolutionary methods. The paper is organised as follows. In Section 2 we provide the basic definitions of multiobjective (combinatorial) optimization and approximation methods. In Section 3 we briefly discuss issues related to the quality of approximations, such as bounds, performance ratios and other quality measures. In Section 4 we present the multiobjective versions of the greedy and local search heuristics and give some examples of their application. The main metaheuristic paradigms of evolutionary and neighbourhood search based are explained in Section 5. In Section 6 we briefly review multiobjective evolutionary methods. Sections 7 and 8 cover Simulated Annealing and Tabu Search Metaheuristics for MOCO. Section 9 is about a metaheuristic based on the ant colonies paradigm, that is becoming more and more popular for MOCO. In Section 10 we mention other techniques and the current trends in the field. Let us remark that we believe that our review covers all significant methods, although we may have missed some of the pertinent literature.

2 Basic Prerequisites

2.1 Multiobjective Optimization

A multiobjective optimization problem is defined as

$$\min_{x \in X} (f_1(x), \ldots, f_p(x)), \tag{MOP}$$

where $X \subset \mathbb{R}^n$ is a feasible set and $f : \mathbb{R}^n \to \mathbb{R}^p$ is a vector valued objective function. By $Y = f(X) \subset \mathbb{R}^p$ we denote the image of the feasible set in the criterion space. We consider optimal solutions of (MOP) in the sense of efficiency (or Pareto optimality), that is, a feasible solution $x \in X$ is called efficient if there does not exist $x' \in X$ such that $f_k(x') \leq f_k(x)$ for all $k = 1, \ldots, p$ and $f_j(x') < f_j(x)$ for some j. In other words, no solution is at least as good as x for all criteria, and strictly better for at least one.

Efficiency refers to solutions x in decision space. In terms of the criterion space, with objective vectors $f(x) \in \mathbb{R}^p$ we use the notion of non-dominance: If x is efficient then $f(x) = (f_1(x), \ldots, f_p(x))$ is called non-dominated (or also efficient). The set of efficient solutions is X_E, the set of non-dominated vectors is Y_N. We may also refer to Y_N as the non-dominated frontier or the trade-off surface. For $y^1, y^2 \in \mathbb{R}^p$ we shall use the notation $y^1 \leqq y^2$ if $y^1_k \leq y^2_k$ for all $k = 1, \ldots, p$; $y^1 \leq y^2$ if $y^1 \leqq y^2$ and $y^1 \neq y^2$; and $y^1 < y^2$ if $y^1_k < y^2_k$ for all $k = 1, \ldots, p$. $\mathbb{R}^p_{\geqq}$ denotes the nonegative orthant $\{y \in \mathbb{R}^p : y \geqq 0\}$, $\mathbb{R}^p_{>}$ is defined analogously.

To solve a multiobjective optimization problem means to find the set of efficient solutions or, in case of multiple x mapping to the same non-dominated point, for each $y \in Y_N$ find an $x \in X_E$ with $f(x) = y$. This concept of a set of efficient solutions is the major challenge of multicriteria optimization. Most methods require the repeated solution of single objective problems which are in some sense related to the multiobjective problem, see e.g. Miettinen (1999) or Ehrgott and Wiecek (2004).

In the following text we shall adopt the notation $x \succ x'$ if x dominates x', i.e. if $f(x) \leq f(x')$.

2.2 Multiobjective Combinatorial Optimization Problems

Multiobjective combinatorial optimization problems can be formulated as follows:

$$\min \{Cx : Ax \geq b, x \in \mathbb{Z}^n\}. \tag{MOCO}$$

Here C is a $p \times n$ objective function matrix, where c^k denotes the k-th row of C. A is an $m \times n$ matrix of constraint coefficients and $b \in \mathbb{R}^m$. Usually the entries of C, A and b are integers. The feasible set $X = \{Ax \geq b, x \in \mathbb{Z}^n\}$ may describe a combinatorial structure such as, e.g., spanning trees of a graph, paths, matchings etc. We shall assume that X is a finite set. By $Y = CX$ we denote the image of X under C in $\mathbb{R}^p$.

The biggest additional challenge in solving MOCOs as compared to multiobjective linear programmes (MOLPs) $\min\{Cx : Ax \geq b, x \geqq 0\}$ results from the existence of efficient solutions which are *not* optimal for any scalarization using weighted sums

$$\min_{x \in X} \sum_{k=1}^{p} \lambda_k f_k(x), \tag{2.1}$$

called unsupported efficient solutions X_{NE}. Those that are optimal for some weighted sum problem (2.1) are called supported efficient solutions X_{SE}. A method called the 2-phase method has been applied to various problems to compute supported (in Phase 1) and unsupported (in Phase 2, using information obtained in Phase 1) efficient solutions, Ehrgott (1999), Lee and Pulat (1993), Ramos et al. (1998), Ulungu and Teghem (1994), Visée et al. (1998). Many methods generalizing single objective algorithms for the use with multiple objectives have been developed, see the survey on exact methods for MOCO in Ehrgott and Gandibleux (2000) and references in that bibliography.

An interesting observation in the context of heuristics is, that an exact algorithm which finds all supported solutions, i.e., solves (2.1) for all $\lambda \in \Lambda = \{\lambda \in \mathbb{R}^p_> : \sum_{k=1}^p \lambda_j = 1\}$, becomes a heuristic for determination of X_E. Also, a heuristic applied to solve (2.1) for some hard problem may actually yield a truly unsupported efficient solution.

From a methodological point of view unsupported efficient solutions are the main reason why the computation of X_E is hard, from a more theoretical point of view the computational complexity is another. Most (MOCO) problems are $\mathbb{NP}$-hard as well as $\#\mathbb{P}$-hard as demonstrated by Ehrgott (2000). The best illustration of this fact is perhaps the unconstrained problem

$$\min_{x\in\{0,1\}^n} \left(\sum_{i=1}^{n} c_i^1 x_i, \sum_{i=1}^{n} c_i^2 x_i \right),$$

with $c_i^1, c_i^2 \geq 0; i = 1, \ldots, n$ which is trivial with only one objective.

2.3 Approximation Methods

As in the single objective case, a reasonable alternative to exact methods for solving difficult MOPs is to derive an approximation method. An *approximation method* in a multiobjective optimization context is a method which finds either sets of locally potentially efficient solutions, that are later merged to form a set of potentially efficient solutions – the approximation – or globally potentially efficient solutions according to the current approximation. *Multiple objective heuristics (MOH)* and *multiple objective metaheuristic (MOMH)* are methods that aim to provide a good tradeoff between an approximation of the efficient solution set, denoted by PE, and the time and memory requirements to obtain them.

In Section 3 we discuss some theoretical issues related to approximation, such as bounds, performance ratios, and quality measures. In Section 4 we describe some genuine multiobjective heuristics. In this section we will not mention any methods to find the set X_{SE}. These are plentiful, but are all based on the principle of repeated application of single objective procedures (exact or heuristic) to (2.1) combined with a recursive dichotomic search procedure to generate the relevant values of λ. They are also very often restricted to two objectives.

Let us now define what we understand by heuristics and metaheuristics.

Heuristics. A heuristic is defined by Reeves (1995) as a technique which seeks good (i.e. near-optimal) solutions at a reasonable computational cost without being able to guarantee optimality, to state how close to optimality a particular feasible solution is or, in some cases, even to guarantee feasibility. Often heuristics are problem-specific, so that a method which works for one problem cannot be used to solve a different one.

Metaheuristics. In contrast, metaheuristics are powerful techniques applicable generally to a large number of problems. A metaheuristic refers to an iterative master strategy that guides and modifies the operations of subordinate heuristics by combining intelligently different concepts for exploring and exploiting the search space (Glover and Laguna (1997), and Osman and Laporte (1996)). A metaheuristic is a solution *concept*. The adaptation to a specific problem uses heuristics as solution methods.

A metaheuristic may manipulate a complete or incomplete single solution or a collection of solutions at each iteration. The family of metaheuristics includes, but is not limited to, constraint logic programming, genetic algorithms, evolutionary methods, neural networks, simulated annealing, tabu search, non-monotonic search strategies, greedy randomized adaptive search, ant colony systems, variable neighbourhood search, scatter search and their hybrids (Osman and Laporte (1996)). The success of these methods is due to their capacity "to solve in practice" some hard combinatorial problems.

3 Quality of Approximation

3.1 Bounds and Bound Sets

The quality of a solution of a combinatorial optimization problem can be estimated by comparing lower and upper bounds on the optimal objective function value. The success of optimization methods, in particular branch and bound, relies on the quality of available bounds.

In multiobjective optimization the concept of bounds is not well developed. The best possible lower and upper bounds on values of all nondominated points are given by the ideal and nadir point y^I and y^N defined by

$$y_k^I = \min_{x \in X} f_k(x) = \min_{y \in Y_N} y_k, \quad k = 1, \dots, p$$

and

$$y_k^N = \max_{x \in X_E} y_k(x) = \max_{y \in Y_N} y_k, \quad k = 1, \dots, p,$$

respectively. We sometimes refer to a utopian point $y^U \leftarrow y^I - \varepsilon \mathbf{1}$, where $\mathbf{1}$ is a vector of all ones and ε is a small positive number. However, the ideal and nadir points are usually far away from non-dominated points and do not provide a good estimate of the non-dominated set. In addition, the nadir point is hard to compute for problems with more than two objectives, see Ehrgott and Tenfelde-Podehl (2003).

To better capture the multiobjective nature of the problems and the fact that we are looking for a set of efficient solutions it is natural to generalize the notion of bounds to bound sets. The following definition is adapted from Ehrgott and Gandibleux (2001). For definitions of $\mathbb{R}^p_{\geqq}$-closedness and -boundedness see e.g. Sawaragi et al. (1985).

Definition 3.1. Let $\bar{Y} \subset Y_N$.

1. A lower bound set L for $\bar{Y}$ is an $\mathbb{R}^p_{\geqq}$-closed and $\mathbb{R}^p_{\geqq}$-bounded set $L \subset \mathbb{R}^p$ such that $\bar{Y} \subset L + \mathbb{R}^p_{\geqq}$ and $L \subset (L + \mathbb{R}^p_{\geqq})_N$.
2. An upper bound set U for $\bar{Y}$ is an $\mathbb{R}^p_{\geqq}$-closed and $\mathbb{R}^p_{\geqq}$-bounded set $U \subset \mathbb{R}^p$ such that $\bar{Y} \in \operatorname{cl}\left[(U + \mathbb{R}^p_{\geqq})^c\right]$ and $U \subset (U + \mathbb{R}^p_{\geqq})_N$.

Given a lower bound set and an upper bound set the quality of an approximation can then be estimated by the "distance" between the two sets. If we observe that a lower bound set is often a convex piecewise linear curve (obtained, e.g., by solving LP relaxations of (MOCO) and connecting the obtained points) and the upper bound set is given by a set of heuristically determined feasible solutions, the distance can be measured by the length of the orthogonal projection of the y values of the upper bound set on the linear pieces of the lower bound set. Alternatively, Tenfelde-Podehl (2002) proposes various measures of the area between a lower and an upper bound set for bicriteria problems.

Ehrgott and Gandibleux report first results on lower and upper bound sets in for the bicriteria knapsack problem (Ehrgott and Gandibleux (2001)) and for the bicriteria TSP, set covering and set packing problems, and the bicriteria assignment problem (Ehrgott and Gandibleux (2004)). Fernández and Puerto (2000) use bound sets in their exact and heuristic methods to solve the multiobjective uncapacitated facility location problem.

3.2 Performance Ratio

A lot of research has been carried out in the area of approximability of $\mathbb{NP}$-hard combinatorial optimization problems, Ausiello et al. (1999). Surprisingly little is known about approximability of multicriteria problems. In single objective optimization, an approximation algorithm A is a (polynomial time) algorithm that finds a feasible solution x_A that is guaranteed to be within a certain ratio of the optimal solution x^*, i.e. $f(x_A) \leq rf(x^*)$. The constant $r \geq 1$ is often called the performance or approximation ratio.

In the multicriteria context, a new definition of performance ratio is needed. If we want to approximate the set X_E by one single heuristic solution, norms can be used. Ehrgott (2000) proposes ratios r_1 and r_2 if the solution x_A guarantees

$$\frac{|\|f(x_A)\| - \|f(x^*)\||}{\|f(x^*)\|} \leq r_1,$$

respectively

$$\frac{\|f(x_A) - f(x^*)\|}{\|f(x^*)\|} \leq r_2$$

for all efficient solutions $x^* \in X_E$. An algorithm with performance ratio $r_1 = 1$ is obtained by solving

$$\min_{x \in X} \|f(x)\|,$$

which, however, can in itself be a hard problem. It is easy to see that in general algorithms with $r_1 < 1$ or $r_2 < 1$ are not possible.

A more interesting question of course is the approximation of X_E by sets PE with bounds on the difference of the set from the true efficient set. To achieve this, it is necessary to define a componentwise notion of approximation. A set of feasible solutions $X' \subset X$ is called ε-efficient (or ε-optimal) if for all $x \in X$ there is some $x' \in X'$ such that

$$f_k(x') \leq (1+\varepsilon) f_k(x), \quad k = 1, \ldots, p,$$

or $(f_k(x') - f_k(x)) \leq \varepsilon f_k(x)$, see Erlebach et al. (2002), Ruhe and Fruhwirth (1990), and Warburton (1987). This is the natural multiobjective version of the approximation ratio with the same ratio $r = (1+\varepsilon)$ for all objectives.

An r-approximation algorithm A is an algorithm that runs in polynomial time and produces an $(r-1)$-efficient solution set for a MOCO problem.

A polynomial time approximation scheme (PTAS) is a family of algorithms that contains for each $\varepsilon > 0$ a $(1+\varepsilon)$-approximation algorithm A_ε. If in addition A_ε is polynomial in ε^{-1} the family is called a fully polynomial time approximation scheme (FPTAS).

The existence of FPTAS for MOCO problems has been discussed by Safer and Orlin (Safer (1992), Safer and Orlin (1995a), Safer and Orlin (1995b)). They obtain results on the existence of FPTAS for some multicriteria network flow, knapsack, and scheduling problems. A general result has been given by Papadimitriou and Yannakakis (2000).

Ruhe and Fruhwirth (1990) (see also Burkard et al. (1989), and Fruhwirth et al. (1989)) present an algorithm which for given $r > 1$ finds an r-approximation of the efficient set of the (continuous) bicriteria network flow problem. Their method is based on an algorithm to approximate a convex curve (note that the efficient set of a bicriteria LP is a piecewise linear convex curve). The algorithm is therefore a pseudopolynomial approximation algorithm for the set X_{SE} of the integer bicriteria network flow problem.

Warburton (1997) and Hansen (1979) give FPTASs for the multicriteria shortest path problem. Erlebach et al. (2001) and Erlebach et al. (2002) develop a fast FPTAS for the multiobjective 1-dimensional knapsack problem and a PTAS for the multiobjective multi-dimensional knapsack problem (an FPTAS for the latter cannot exist unless $\mathbb{P} = \mathbb{NP}$). In both papers, a partition of the objective space into intervals of increasing length is used. The fact that the knapsack algorithms are based on dynamic programming is another link between the two methods.

White (1986) proposes a number of definitions of ε-efficiency. Among them an additive version of the definition above, i.e. $\hat{x} \in X$ is called ε-efficient (or ε-optimal) if there is no $x' \in X$ such that $f(x') \leq f(\hat{x}) - \varepsilon \mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^p$ is a vector of all ones. These definitions, however, have not found attention in MOCO literature.

3.3 Other Quality Measures

In the previous sections we have mentioned the distance between lower and upper bound sets and the performance or approximation ratios as quality measures for approximative solutions. For problems for which test instances with known Pareto optimal solutions are available the quality of an approximation method can be estimated by the percentage of truly efficient points that it can detect.

There are a few other ideas in the literature. Kim et al. (2001) propose a new measure, the integrated convex preference (ICP), to compare the quality of algorithms for MOCO problems with two objectives.

Sayin (2000) proposes the criteria of coverage, uniformity, and cardinality as quality measures. Although developed for continuous problems the ideas may be interesting for MOCO problems. However, the methods proposed in Sayin (2000) can be efficiently implemented for linear problems only.

Other authors propose distance based measures, Viana and Sousa (2000) and visual comparisons of the generated approximations. The latter are restricted to bi-objective problems. Jaszkiewicz (2001c) also distinguishes between cardinal and geometric quality measures. He gives further references and suggests preference-based evaluation of approximations of the non-dominated set using outperformance relations.

None of these measures have been universally adopted in the multiobjective optimization literature, and further research is clearly needed.

4 Heuristics for Multiobjective Combinatorial Optimization

Heuristics for MOCO problems can often be derived from heuristics for the single objective version of a combinatorial optimization problem. Two main strategies have been used. The first is to apply the single objective heuristic directly to solve (2.1) and combine it with a procedure to select appropriate values of λ.

We are interested in heuristics that deal with the multiple objectives directly. This can often be achieved by modifying the operators in the heuristic to deal with vectors rather than scalars. Thus, vector addition is used and the min or max operators are understood in the sense of the partial order $\leq$ and the preference relation $\succ$: $\min\{f(x) : x \in X\} = \{f(x) : x \in X$ such that there is no $x' \in X$ with $f(x') \leq f(x)\}$. Accordingly we write argmin $\{f(x) : x \in X\} = \{x \in X :$ there is no $x' \in X$ such that $x' \succ x\}$.

4.1 General Multiobjective Heuristics

In this section we describe two main heuristic principles in the multiobjective framework: the greedy and the local search principle. Let us consider $X \subset 2^A$ for some finite set A and assume, for ease of exposition that if $x \in X$ and $x' \subset x$ then $x' \in X$. Due to the partial order and the resulting incomparability of objective vectors, solutions are constructed in parallel (Algorithm 1).

Algorithm 1 Multiobjective Greedy Algorithm

`MOGreedy`: **procedure** $(A,\ PE)$

  input : A, finite set
  output : PE, the set of potentially efficient solutions

$PE \leftarrow$ `nondom`(A)
while $(\exists x \in PE, a \in A \setminus x$ such that $x \cup a \in X)$ **do**
  for all $x \in PE, a \in$ `nondom`$(A \setminus x)$ **loop**
    if $(x \cup a \in X)$ $PE \leftarrow PE \cup (x \cup a)$
  end loop
  $PE \leftarrow$ `nondom`(PE)
end while

`nondom`(S): **procedure** (S)
`nondom`$(S) \leftarrow \{x \in S : \nexists x' \in S : f(x') \leq f(x)\}$

Such a constructive algorithm (without the filter for non-dominated vectors after the end of the for loop) has been used for the minimum spanning tree problem by Corley (1985). It performs reasonably well on problems whose single objective version can be solved optimally by the greedy algorithm. However, it does not necessarily generate all efficient solutions even for such problems. Serafini (1986) showed that for each efficient solution there is a topological order, such that if the greedy algorithm uses that topological order all efficient solutions of matroid problems are found. A greedy heuristic is also used by Rosenblatt and Sinuany-Stern (1989).

To describe the multiobjective local search heuristic we shall assume that a neighbourhood definition is given and that for any $x \in X$, $\mathcal{N}(x)$

defines the neighbourhood of x (Algorithm 2).

Algorithm 2 Multiobjective Local Search

`MOLocalSearch`: **procedure** $(x, \mathcal{N},\ PE)$

input : x, feasible solution
output : PE, the set of locally potentially efficient solutions

$PE \leftarrow x$, $S \leftarrow x$
repeat
  for all $\hat{x} \in S$ **loop**
    for all $x \in \mathcal{N}(\hat{x})$ **loop**
      if $(x \not\succ \widehat{x}$ and $\widehat{x} \not\succ x)$ $S \leftarrow S \cup x$
      if $(x \succ \hat{x}) S \leftarrow S \cup x \setminus \hat{x}$
    end loop
    $S \leftarrow$ `nondom`(S)
  end loop
until $S = PE$

Andersen et al. (1996) give a local search heuristic based on the exchange of edges for the bicriteria spanning tree problem. Hamacher and Ruhe (1994) use a local search procedure in a two-phase algorithm for the bicriteria spanning tree problem after first calculating supported efficient solutions exactly.

The greedy and local search heuristics can be combined. The idea is to first construct solutions using the greedy approach and use local search to improve the solutions or generate more potentially efficient solutions. Such a combination is used by Sigal (1994) for the TSP. Both the multiobjective greedy algorithm and the multiobjective local search algorithm are of course general templates that can be applied to a wide variety of problems.

We shall now discuss heuristic methods for the multicriteria TSP as an example of how single objective heuristics can be adapted to the multicriteria case. Two of the best known heuristics for the TSP are the tree and Christofides' heuristic, Lawler et al. (1985). Both procedures begin by first constructing a minimal spanning tree. The tree heuristic then duplicates each edge of the spanning tree, thus obtaining a graph where all nodes have even degree. This graph is therefore Eulerian and has a tour that traverses each edge exactly once. A Hamiltonian cycle can then be determined by elimination of nodes that are repeatedly visited.

Christofides' heuristic improves upon that by finding a minimal weight perfect matching of those nodes which have odd degree. The union of the spanning tree and the minimal weight matching is again Eulerian. From there the heuristic proceeds as the tree heuristic.

How can this be modified for the multiobjective TSP? Ehrgott (2000) replaces the steps of finding a minimal spanning tree and matching by finding a spanning tree and matching with minimal norm of the vector valued weights. He gives a number of results on the approximation properties of these algorithms according to the definitions of r_1- and r_2-performance ratios mentioned in Section 3.2.

5 Multiobjective Metaheuristics

The adaptation of metaheuristic techniques for the solution of multiobjective optimization problems has mushroomed over the last ten years, giving birth to multiobjective metaheuristic (MOMH). From a historical perspective, the pioneer approximation methods for multiobjective problems have appeared since 1984, in the following order: Genetic Algorithms (GA, Schaffer (1984)), Artificial Neural Networks (ANN, Malakooti et al. (1990)), Simulated Annealing (SA, Serafini (1992)), and Tabu Search (TS, Gandibleux et al. (1997)). The pioneer methods have two characteristics. First, they are inspired either by *Evolutionary Algorithms (EA)*, or by *Neighbourhood Search Algorithms (NSA)*. Second, the early methods are direct derivations of single objective optimization metaheuristics, incorporating small adaptations to integrate the concept of efficient solution for optimizing multiple objectives.

5.1 Evolutionary Algorithms versus Neighbourhood Search Algorithms

Evolutionary Algorithms manage a solution population $\mathcal{P}$ rather than a single feasible solution. In general, they start with an initial population and combine principles of self adaptation, i.e. independent evolution (as in the mutation strategy in genetic algorithms), and cooperation, i.e. the exchange of information between individuals (as in the pheromones handled in ant colonies), to improve approximation quality. Because the whole population contributes to the evolutionary process, the generation mechanism is parallel along the frontier, and thus these methods are also called *global convergence-based methods*. This characteristic makes population-based methods very attractive for solving multiobjective problems.

In *Neighbourhood Search Algorithms*, generation relies upon one individual, a current solution x_n, and its neighbours $\{x\} \subseteq \mathcal{N}(x_n)$. Using a local aggregation mechanism for the objectives (often based on a weighted sum), a weight vector $\lambda \in \Lambda$, and an initial solution x_0, the procedure iteratively projects the neighbours into the objective space in a search direction λ, by optimizing the corresponding parametric single objective problem. A local approximation of the non-dominated frontier is obtained using archives of the successive potentially efficient solutions detected. This generation mechanism is sequential along the frontier, producing a local convergence to the non-dominated frontier, and so such methods are called *local convergence-based methods*. The principle is repeated for diversified search directions to completely approximate the non-dominated frontier. NSAs present an aggressive convergence because the search is less dispersed, but they require more effort in diversification in order to cover the efficient frontier completely. All NSA methods use a current solution x_n and a neighbourhood $\mathcal{N}(x_n)$ to generate a new solution $x \in \mathcal{N}(x_n)$. The comparison of x and x_n according to p objectives $f_k(x)$, $k = 1 \ldots p$ raises three possible situations (Figure 1 for a biobjective maximization problem) if $\Delta f_k = f_k(x) - f_k(x_n)$ is the difference between solutions x and x_n with respect to objective k.

(Case Ca) $\forall k \, \Delta f_k \geq 0$
All the objectives are improved for solution x. x (weakly) dominates the current solution x_n and is always accepted. If $\forall k \, \Delta f_k = 0$, the two solutions are equivalent. x is always accepted, except if only a *minimal complete set* X_{E_m} of efficient solutions (i.e. a subset of X_E that contains no equivalent solutions, and for any $x \in X_E$ there exists $x' \in X_{E_m}$ such that x and x' are equivalent) is computed.

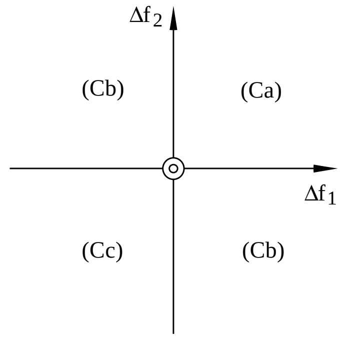

Figure 1: *Three situations can occur when comparing solutions x and x_n according to multiple objectives (here, in the case of two maximized objectives).*

(Case Cb) $\exists k, k' : \Delta f_k < 0$ and $\Delta f_{k'} > 0$
An improvement and a deterioration occur simultaneously for different criteria. Both solutions x and x_n are potentially efficient.

(Case Cc) $\forall k \, \Delta f_k \leq 0$
All objectives are deteriorated with at least one strict inequality. Solution x is dominated by x_n.

For the two last cases a scalarizing function $s(f(x), \lambda)$ is often used to project the multidimensional objective space into a monodimensional one using a weight vector $\lambda \in \mathbb{R}^p_{\geq}$. Such functions are well-known in the context of interactive procedures for multiobjective optimization. The scalarizing function allows to produce a "local aggregation" of the objectives in order to compute the "weighted distance" $\Delta s = s(f(x), \lambda) - s(f(x_n), \lambda)$ between $f(x)$ and $f(x_n)$.

Today, the multiobjective metaheuristics are often hybridized. For example, some NSA-based methods also handle a population of solutions, see Czyzak and Jaszkiewicz (1996) and Hansen (1998). This coupling aims to break the independence aspect of the search process, inherent in the sequential generation principle, by exploiting information available in the population. One idea is to use the population as generator for a diversification strategy.

5.2 False Multiobjective Methods

Metaheuristics dealing with multiple objective optimization problems are sometimes wrongly presented as MOMH. This occurs when multiobjective problems are solved with a single objective strategy, looking for a unique compromise solution. In this case the original multiobjective problem is transformed or managed as the optimization of one or several single objective problems. The two following examples are representative of this class of methods. They are presented in Osman and Laporte (1996) as the first papers describing the use of TS as a technique for solving multiobjective problems.

A sequence of single objective problems. In Hertz et al. (1994) the method consists in solving a sequence of single objective problems considering in turn each objective f_j associated with a penalty term:

$$\begin{array}{lrcll} \min & f_k(x) & & & \\ \text{s.t.} & f_j(x) & \leq & \bar{f}_j & j = 1 \ldots k-1 \\ & x & \in & X. & \end{array}$$

A parameterized value function. In Dahl et al. (1995) a family of scalarized problems (2.1) are solved to generate an approximation of a subset of X_E.

$$\begin{array}{ll} \min & \lambda_1 f_1(x) + \lambda_2 f_2(x) + \lambda_3 f_3(x) \\ \text{s.t.} & x \in X. \end{array}$$

We designate such approaches as “false multiobjective methods” and do not consider them in the framework of this synthesis.

6 The Multiobjective Evolutionary Algorithms Wave

The first to introduce a multiobjective metaheuristic was Schaffer (Schaffer (1984) and Schaffer (1985)). He developed a multiobjective evolutionary algorithm (MOEA), called *Vector Evaluated Genetic Algorithm (VEGA)*, which was an extension of Grefenstette’s GENESIS programme (Grefenstette (1984)) to include multiple objective functions. The vector extension concerns only the selection procedure.

6.1 Vector Evaluated Genetic Algorithm by Schaffer (1984)

For each generation in VEGA, three stages are performed (Algorithm 3). The selection procedure is performed independently for each objective. In the first stage, the population is divided into p subpopulations S^k according to their performance in objective k (routine `pickIndividuals`). Each subpopulation is entrusted with the optimization of a single objective. In the next stage, subpopulations are shuffled to create a mixed population (routine `shuffle`). In the final stage, genetic operators, such as mutation and crossover, are applied to produce new potentially efficient individuals (routine `evolution`). This process is repeated for N_{gen} iterations.

Algorithm 3 VEGA, Vector Evaluated Genetic Algorithm

input :	*pop*,	the population size
	N_{gen},	the limits of generations
	parameters,	the crossover probability and mutation rate
output :	PE,	the set of potentially efficient solutions

begin `VEGA`

--| Generate an initial population of *pop* individuals
  $\mathcal{P}_0 \leftarrow$ `initialization`(pop)

--| Generation process
  for n **in** $1, \ldots, N_{gen}$ **loop**

  --| 1. Elaborate p sub-population of size pop/p using each objective k
  --| in turn
    $S^k \leftarrow$ `pickIndividuals`$(\ pop/p\ ,\ k\ ,\ \mathcal{P}_{n-1}),\ \forall k = 1, \ldots, p$

  --| 2. Set a population of size *pop* in shuffling together the p
  --| sub-populations S^k
    $S \ \leftarrow$ `shuffle`$(\cup_{k=1,\ldots,p} S^k)$

  --| 3. Apply genetic operators
    $\mathcal{P}_n \leftarrow$ `evolution`$(S, parameters)$

  endLoop
  $PE \leftarrow \mathcal{P}_{N_{gen}}$

end `VEGA`

Because VEGA selects individuals who excel in one performance dimension without looking at the other dimensions, the speciation problem can arise with that method. This implies that individuals with a balanced performance on all objectives will not survive under this selection mechanism. Speciation is undesirable because it is opposed to the generation of compromise solutions. Due to this characteristic VEGA is labeled as a non-Pareto approach, Coello (1999). Additional heuristics were developed (like crossbreeding among the species) and studied to overcome this tendency.

6.2 Modern MOEAs

Although, it has some serious drawbacks, VEGA has had a strong influence up to now, and was at the origin of the Multiobjective Evolutionary Algorithm (MOEA) wave. Since VEGA many Multiobjective Evolutionary Algorithms have been developed. Significant progress concerns corrections of shortcomings observed in the first algorithms introduced and propositions of new algorithmic primitives to generate a better approximation of X_E. These MOEAs are characterized according to population structure, archiving, selection/elitism mechanism, and fitness function. Laumanns et al. (2001) give an overview of the techniques that are applied in most MOEAs (Figure 2).

Two central questions motivate the research about MOEA:

(1) how to accomplish both fitness assignment and selection in order to guide the search toward the efficient frontier?

(2) how to maintain a diversified population in order to avoid premature convergence and find a uniform distribution of solutions along the efficient frontier?

For the first question MOEAs are distinguished by the way the performance of individuals is evaluated in the selection. Firstly, when *the objectives are considered separately*, the selection of individuals is performed by considering each objective independently (Schaffer (1995)); or the selection is based on a comparison procedure according to a predefined (or random) order on the objectives (Fourman (1985)); or the selection takes into account probabilities assigned to each objective in order to determine a predominant objective (Kursawe (1992)). Secondly, when *the objectives are*

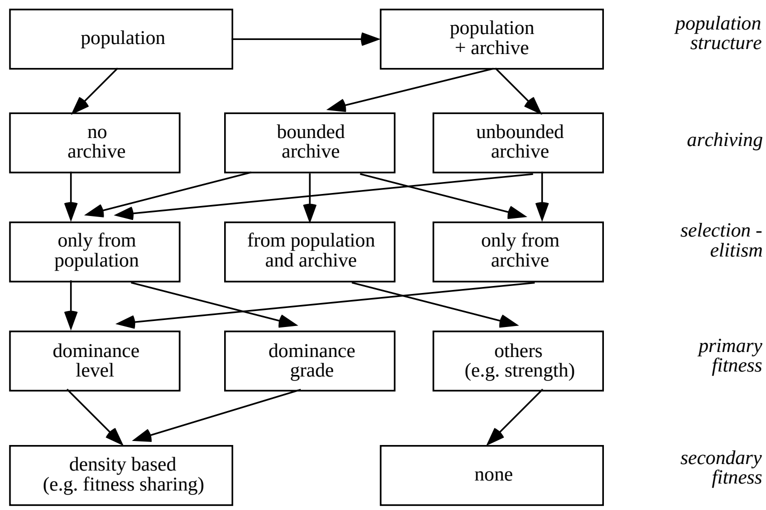

Figure 2: *A taxonomy of techniques in MOEAs (Laumanns et al. (2001)). The arrows indicate existing combinations.*

aggregated into a single parameterized objective function, the parameters of the function are systematically updated during the same runs (at random or using a particular weight combination) taking advantage of information collected on the population of individuals (Hajela and Lin (1992) and Murata and Ishibuchi (1995)). Each aggregation defines a search direction in the objective space and the idea is to optimize in multiple directions simultaneously. Thirdly, when *the concept of efficiency is directly used* (non-domination ranking) the fitness of an individual (i.e. a solution) is calculated on the basis of the dominance definition. The idea is to take advantage of information carried by the population of solutions using the notion of domination for selection. This is the most common approach and has led to several Pareto-based fitness assignment schemes, see Fonseca and Fleming (1993), Horn et al. (1994), Srinivas and Deb (1994), Zitzler and Thiele (1998), etc.

Goldberg (1989) has suggested the use of non-domination ranking according to the following principle (Figure 3). All non-dominated individuals are assigned rank 1. They are then temporarily withdrawn from the population. Then the remaining non-dominated individuals take rank 2 and so forth. The rank of an individual determines its fitness value. It is a concept of fitness related to the whole population. This evaluation scheme is implemented in NSGA by Srinivas and Deb (1994). Other evaluations of the rank have been proposed. For Fonseca (Fonseca and Fleming (1993)) the rank of an individual is equal to the number of solutions which dominate this individual (implemented in MOGA93). For Zitzler and Thiele (1998) a strength value $s \in [0, 1)$ is calculated for each potentially efficient solution. For all other individuals in the population, the strengths of all potentially efficient solutions by which it is covered (dominated) are summed up.

The majority of the other components of a MOEA deal with the second question. A fitness sharing based on a principle of niches is the most frequently used technique. Most MOEAs are implementing fitness sharing, e.g., Fonseca and Fleming (1993), Horn et al. (1994), Srinivas and Deb (1994), Zitzler and Thiele (1998). *Niches* are solution neighbourhoods in objective space centered on candidate solutions and with a radius σ_{sh}. Based on the number N of solutions in these niches, the selection of individuals can be influenced to generate more in areas where niches are sparsely populated, with the goal of greater distribution uniformity along the non-dominated frontier (see Figure 4). A *sharing function*, which measures the distance $d(i,j)$ between a candidate solution i and a neighbour j, is defined by

$$\phi(d(i,j)) = 1 - \left(\frac{d(i,j)}{\sigma_{sh}}\right)^{\alpha}$$

if $d(i,j) < \sigma_{sh}$ and 0 otherwise. The parameter α amplifies ($\alpha > 1$) or attenuates ($\alpha < 1$) the sharing value computed. Thus the shared fitness of candidate i is

$$f_{s_i} = \frac{f_i}{\sum_{j=1}^{N} \phi(d(i,j))}$$

such that the fitness of candidates increases if ϕ values are small, i.e. the distance of neighbours from i is close to σ_{sh}.

Among the elements playing a significant role in a MOEA, the importance of the elite solutions to improve MOEAs is underlined by Laumanns et al. (2001). The results of numerical experiments show that the use of elitism must be accompanied by a strong rate of mutation in order to prevent the population from specializing too fast. The contribution of elite solutions in the generation of the non-dominated frontier in MOCO problems has been investigated by Gandibleux et al. (2001) and Morita et al. (2001). For the knapsack problem, using greedy solutions, or supported efficient solutions, in the population make the algorithm more apt to generate

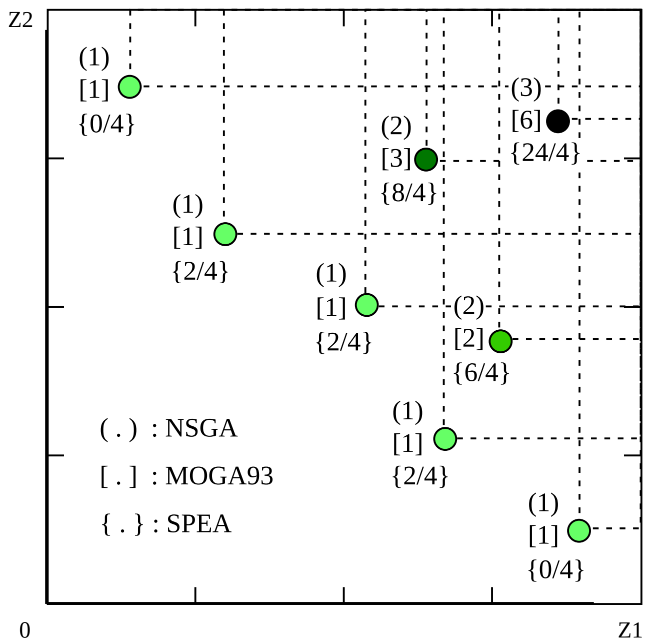

Figure 3: *Computation of rank in NSGA (Srinivas and Deb (1994)), MOGA93 (Fonseca and Fleming (1993)) and SPEA (Zitzler and Thiele (1998)).*

efficient solutions (Figure 5). Also, using elite solutions in combination with a generic path-relinking operator has shown impressive results in knapsack and assignment problems, Gandibleux et al. (2004).

6.3 Bibliographic Notes

A number of important implementations of MOEA have been published in recent years. There are even a number of surveys on the topic (see Coello (1999), Coello (2000), Fonseca and Fleming (1995), and Jones et al. (2002)). The most outstanding among the pioneer MOEAs are mentioned briefly below:

Vector Evaluated Genetic Algorithm (VEGA) by Schaffer (1984).

Multiple Objective Genetic Algorithm (MOGA93) by Fonseca and Fleming (1993). MOGA93 uses a ranking procedure in which the rank

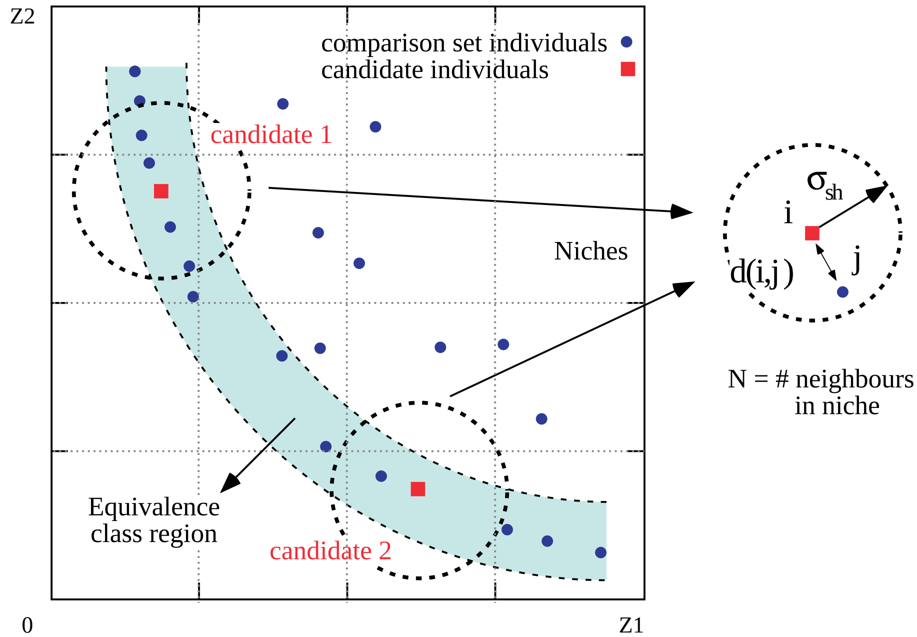

Figure 4: *Niches are solution neighbourhoods in the objective space.*

of an individual is equal to the number of solutions which dominate this individual.

Non-dominated Sorting Genetic Algorithm (NSGA) by Srinivas and Deb (1994). NSGA implements Goldberg's ranking idea in which the rank of an individual is equal to its domination layer, computed by ranking the population on the basis of domination.

Niched Pareto Genetic Algorithm (NPGA) by Horn, Nafpliotis and Goldberg, 1994 (Horn et al. (1994)). NPGA combines the Pareto dominance principle and a Pareto tournament selection where two competing individuals and a set of individuals are compared to determine the winner of the tournament.

For a long time, the problems investigated with these methods were often unconstrained bi-objective problems with continuous variables and non-linear functions. EA are appreciated by the engineering community which

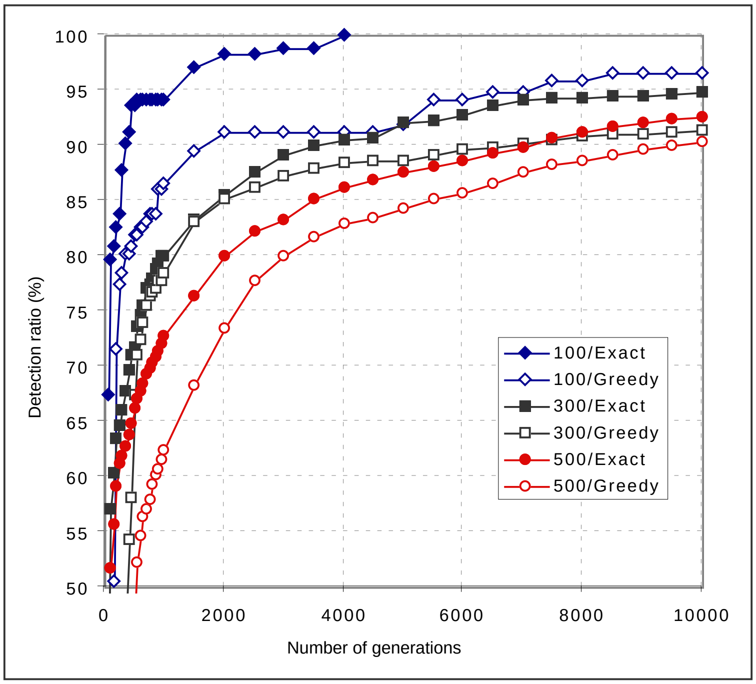

Figure 5: *Influence of the quality of elite solutions (supported solutions and greedy solutions) on the detection ratio (number of exact solutions in the approximation set) in the case of bi-objective knapsack problems (100, 300 and 500 variables).*

could explain the large number of MOEA applications for solving real world problems (in mechanical design or electronics, for example). Surprisingly, few MOEA have been applied to solve MOCO problems. A survey of the literature shows Gen and Li (1998a), Gen and Li (1998b) (Transportation Problems), Zhou and Gen (1999) (Spanning Tree Problems), Jaszkiewicz (1998) (Travelling Salesperson Problems), Ben Abdelaziz et al. (1999), Gandibleux et al. (2001) (Knapsack Problems), Zitzler and Thiele (1999) (Multi-constraint Knapsack Problems), Liepins et al. (1990) (Set Covering Problems), Todd and Sen (1997) (Containership Loading Design), Morita et al. (2001), Pamuk and Köksalan (2001), Tamaki et al. (1994) (Scheduling Problems). The methods which have been used for MOCO are described below:

Multiple Objective Genetic Algorithm (MOGA) by Murata and Ishibuchi (1995). This method is not based on the Pareto ranking principle but on a weighted sum of objective functions, combining them into a scalar fitness function that uses randomly generated weight values in each iteration. Later, the authors coupled a local search with a genetic algorithm, introducing the memetic algorithm principle for multiobjective problems.

Method of Morita et al. (MGK) by Morita, Gandibleux and Katoh, 1998 (Gandibleux et al. (1998)). Seeding solutions, either greedy or supported efficient, are put in the initial population in order to initialise the algorithm with good genetic information. The biobjective knapsack problem is used to validate the principle. This method becomes a memetic algorithm when a local search has been performed on each new potentially efficient solution, Gandibleux et al. (2001).

Strength Pareto Evolutionary Algorithm (SPEA) by Zitzler and Thiele (1998). SPEA takes the best features of previous MOEAs and combines them to create a single algorithm. The multiobjective multi-constraint knapsack problem has been used as a benchmark to evaluate the method (Zitzler and Thiele (1999)).

Pareto Archived Evolution Strategy (PAES) by Knowles and Corne (1999). PAES is an evolutionary strategy that employs local search to generate new candidate solutions and a reference archive to compute solution quality.

Multiple Objective Genetic Local Search (MO-GLS) by Jaszkiewicz (2001b). This method hybridizes recombination operators with local improvement heuristics. A scalarizing function is drawn at random for selecting solutions, which are then recombined, and the offspring of the recombination are improved using heuristics.

Multiple Objective Genetic Tabu Search (MOGTS) by Barichard and Hao (2002). This is another hybrid method in which a genetic algorithm is coupled with a tabu search. MOGTS has been evaluated on the multi-constraint knapsack problem.

7 The Simulated Annealing Wave

Serafini (1992) was the first to use simulated annealing as a technique for multiobjective optimization problems. All multiobjective simulated annealing-based methods developed since then are still closely related to the original single objective method. They extend the single objective algorithm to cope with the notion of efficiency.

Often the authors tested various forms and definitions of acceptance rules. A lot of them have been suggested and discussed by Serafini (1992). Among those, four rules are frequently employed. If T_n is the current temperature value at iteration n the acceptance probability is:

- Rule C (Chebyshev rule):

$$p_n \leftarrow \min \left\{ 1 \; ; \exp \left(\max_{k=1\ldots p} \left\{ -\frac{\lambda_j \; (f_k(x) - f_k(x_n))}{T_n} \right\} \right) \right\} \tag{7.1}$$

- Rule SL (scalar linear rule):

$$p_n \leftarrow \min \left\{ 1 \; ; \exp \left(\sum_{k=1}^{p} \left\{ -\frac{\lambda_j \; (f_k(x) - f_k(x_n))}{T_n} \right\} \right) \right\} \tag{7.2}$$

- Rule W (weak rule):

$$p_n \leftarrow \min \left\{ 1 \; ; \exp \left(\min_{k=1\ldots p} \left\{ -\frac{\lambda_j \; (f_k(x) - f_k(x_n))}{T_n} \right\} \right) \right\} \tag{7.3}$$

- Rule P (product rule):

$$p_n \leftarrow \prod_{k=1\ldots p} \min \left\{ 1 \; ; \exp \left(-\frac{\lambda_j \; (f_k(x) - f_k(x_n))}{T_n} \right) \right\}. \tag{7.4}$$

Three methods are reported chronologically in this section, Czyzak and Jaszkiewicz (1996), Parks and Suppapitnarm (1999), and Ulungu and Teghem (1992). It is important to emphasize that these methods have been elaborated independently from each other. The first method by Ulungu and Teghem (1992) uses a predefined set of weights $\lambda \in \mathbb{R}^p_+$. An independent SA process is then executed for each weight value. The second method by

Czyzak and Jaszkieweicz (1996) introduces the use of a sample of solutions which are simultaneously optimized toward the efficient frontier while they are dispersed over the whole frontier. The last method by Parks and Suppapitnarm (1999) is not based on a principle of search directions, and it does not need an aggregation mechanism for the objectives. Each objective is considered separately. Advanced strategies using the population of potentially efficient solutions drive the approximation mechanism ensuring the detection of the whole efficient frontier.

7.1 Multiobjective Simulated Annealing by Ulungu (1992)

In 1992 (EURO XII conference, Helsinki), Ulungu introduced MOSA, one of the most popular simulated annealing methods for multiobjective optimization, see also Ulungu (1993). MOSA is a direct derivation of the simulated annealing principle for handling multiple objectives. Starting from an initial, randomly generated solution $x_{n=0}$ and a neighbourhood structure $\mathcal{N}(x_n)$, MOSA computes a neighbour $x \in \mathcal{N}(x_n)$ using a set of weights Λ that define search directions $\lambda \in \Lambda$. The comparison of x with x_n according to p objectives $f_k(x)$, $k = 1, \ldots, p$ gives rise to three possible cases. If $\Delta f_k = f_k(x) - f_k(x_n)$ is the difference between solution x and x_n in the objective k (Figures 6 and 7 and Algorithm 4):

(Ca) $\forall k\, \Delta f_k \leq 0$: All the objectives are improved for solution x. x_n (weakly) dominates x.

(Cb) $\exists k, k'\, \Delta f_k < 0$ and $\Delta f_{k'} > 0$: Improvement and deterioration occur simultaneously for different criteria. Both solutions x and x_n are potentially efficient.

(Cc) $\forall k\, \Delta f_k \geq 0$: All objectives are deteriorated with at least one strict inequality. Solution x is dominated by x_n.

A neighbour x is always accepted if it dominates x_n (Ca). When x is dominated (Cc), it can be accepted according to a decreasing probability, depending on the current "temperature" of the cooling schedule (Routine `isAccepted`). In the initial version of MOSA, a neighbour belonging to the intermediate situation (Cb) was also always accepted (Routine `isBetter` and Figure 6). This acceptance principle has been revised in a recent version of the method to include the search direction in the decision (Figure 7).

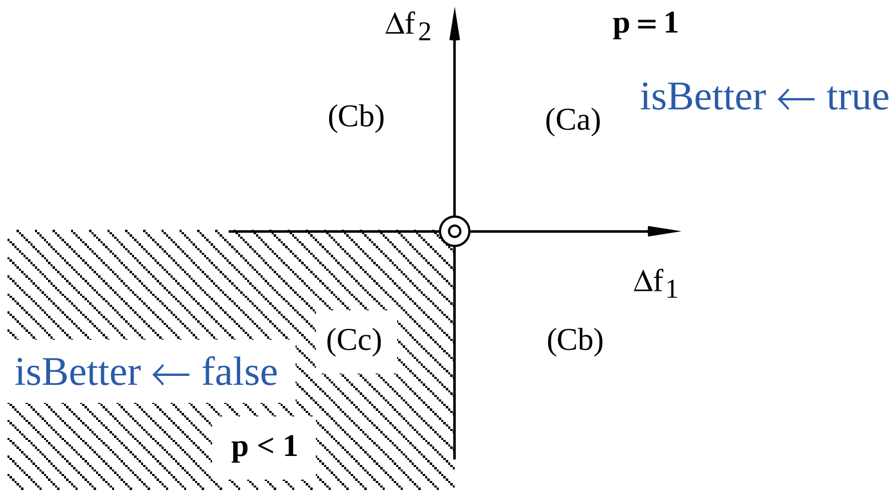

Figure 6: *Acceptance principle in MOSA.*

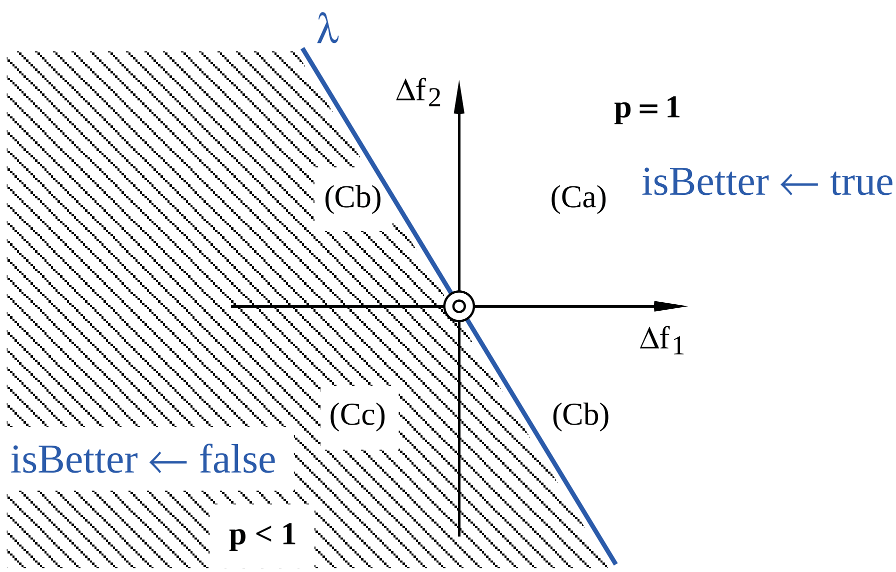

Figure 7: *Revised acceptance principle in MOSA.*

To measure the degradation in the routine `isAccepted`, the values are aggregated in a scalar using a scalarizing function $S(f(x), \lambda)$. Such a function makes a "local aggregation" of the objectives which allows the computation of the "weighted distance" $\Delta s = S(f(x), \lambda) - S(f(x_n), \lambda)$ between $f(x)$ and $f(x_n)$.

When a neighbour is accepted, the set of potentially efficient solutions PE_λ in direction λ is updated. The search stops after a certain number of iterations, or when a predetermined temperature is reached (Routine `isFinished`). At the end, MOSA combines the sequential processes in the objective space Y in a unique set PE by merging the sets PE_λ (Routine `merge`). The outline of MOSA for maximizing objectives is given in Algorithm 4.

The methods that follow are different primarily on four points: (1) the rule for acceptance of a new solution with some probability depending on the temperature; (2) the scheme for decreasing the temperature; (3) the mechanism that guides the browsing of the non-dominated frontier; (4) the use of information drawn from a population of individuals. Various forms and definitions of the acceptance rules have been tested by the method's designers; many of these have been suggested and discussed in Serafini (1992). The most recent SA based methods include dynamic diversification mechanisms that exploit the set of potentially efficient solutions to drive the approximation process, Czyzak and Jaszkiewicz (1999).

7.2 Bibliographic Notes

The following methods based on simulated annealing have been published in the literature:

Multiobjective Simulated Annealing (MOSA) by Ulungu (1993). The principle of MOSA is described in Ulungu's PhD thesis, 1993. Numerical experiments are reported for three bi-objective combinatorial problems (assignment, transportation and knapsack problems). The method is presented later, Ulungu et al. (1995), in a systematic framework. Complete numerical experiments for the bi-objective knapsack problem are reported and compared with solutions obtained with an exact method. An interactive version of MOSA is derived in Teghem et al. (2000) and Ulungu et al. (1998). In Teghem et al.

Algorithm 4 MOSA, MultiObjective Simulated Annealing

input :	Λ,	set of weights
	$T, \alpha, N_{step}, T_{stop}, N_{stop}$,	SA parameters
output :	PE,	set of potentially efficient solutions

begin `MOSA`

$PE \leftarrow \emptyset$
for all $\lambda \in \Lambda$ **loop**
  $T_0 \leftarrow T$; $N_{count} \leftarrow 0$; $n \leftarrow 0$
  randomly draw $x_n \in X$; $PE_\lambda \leftarrow \{x_n\}$
  repeat randomly draw $x \in \mathcal{N}(x_n)$
    if `isBetter`(x, x_n) **or else** `isAccepted`$(x, x_n, n, T_n, \lambda)$ **then**
      $PE_\lambda \leftarrow$ `archive`(PE_λ, x); $x_{n+1} \leftarrow x$; $N_{count} \leftarrow 0$
    else
      $x_{n+1} \leftarrow x_n$; $N_{count}++$
    endIf
    $n++$; `updateParameters`(α, n, T_n)
  until `isFinished`(N_{count}, T_n)
endLoop
$PE \leftarrow$ `merge`(PE_λ)

end `MOSA`

With :

- `isBetter`, a predicate that computes $\Delta f_k \leftarrow f_k(x) - f_k(x_n)$ and then returns $\neg(\forall k : \Delta f_k \leq 0)$.
- `isAccepted`, a predicate that computes the acceptance probability p_n for current iteration as $\exp\left(-\sum_{k=1}^{p} \lambda_k \ (f_k(x) - f_k(x_n))/T_n\right)$, draws p randomly and uniformly distributed in $[0, 1]$ and returns $(p < p_n)$.
- `isFinished`, a predicate that checks whether either a predefined number of iterations N_{stop} or a limit temperature T_{stop} in the cooling schedule has been reached.
- `merge`, a function that builds the union $PE \cup PE_\lambda, \lambda \in \Lambda$ eliminating dominated solutions: $\text{argmin}(PE \cup PE_\lambda)$

(2000), a simulation with a fictitious decision maker is reported for the knapsack problem with four objectives and the assignment problem with three objectives. In Ulungu et al. (1998) the interactive version is used for solving a real situation: the problem of homogeneous grouping of nuclear fuel. In Ulungu et al. (1999), different options in the implementation of MOSA are illustrated by the use of extensive experiments. The bi-objective knapsack problem is used as support and discussions are given in comparison with exact solutions. Tuyttens et al. (2000) is devoted to the use of MOSA for solving the bi-objective assignment problem. The improvement over the initial solution is discussed. Numerical results are reported and compared with solutions obtained with an exact method. The solution of a variety of scheduling problems with MOSA has been experimented in Loukil Moalla et al. (2000b). Numerical results for bi-criteria single machine, parallel machine, and permutation flowshop problems are reported. Some remarks on the number of potentially efficient solutions are mentioned. Recently a multiple objective Vehicle Routing Problem with time windows has been solved with MOSA, El-Sherbeny (2001). Three categories of objectives are discussed: concerning the vehicles used (number of vehicles, number of covered/uncovered vehicles), concerning time (total duration of the routes, the homogeneity of the duration of the routes, working time not used, total waiting times due to time-window constraints), and concerning the flexible duration of the routes (longer duration of the routes is preferred).

Pareto Simulated Annealing (PSA) by Czyzak and Jaszkiewicz (1996), Czyzak and Jaszkiewicz (1997), and Czyzak and Jaszkiewicz (1998). PSA combines simulated annealing principles with ideas coming from genetic algorithms. The main differences to MOSA concern the management of weights and the consideration of a set of current solutions. A sample $S \subset X$ of $\#S$ solutions is determined and used as initial solutions. Each solution in this set is "optimized" iteratively following the same mechanisms explained above, i.e. by generating neighbouring solutions that may be accepted according to a probabilistic strategy. The authors suggest one of the C (7.1), SL (7.2), or W (7.3) rules for the acceptance probability $P(x, x_n, T_n, \lambda)$. For a given solution $\bar{x} \in S$ the weights are changed in order to increase the probability of moving it away from its closest neighbour in S denoted by $\bar{x}'$. Solutions in S play the role of agents working almost independently but exchanging information about their positions in the objective space. Thus, the interaction between solutions guides the generation process through the values of λ. This exploration principle will hopefully lead to an approximation spread uniformly along the non-dominated frontier. On the basis of the results of numerical experiments the way of recomputing the weights has been redefined in later papers by Jaszkiewicz (2001a) and Jaszkiewicz (2001c). An additive formulation has shown best performances and replaces the initial multiplicative form.

The PSA method is published for the first time in 1996 (Czyzak and Jaszkiewicz (1996)), but it has been described in technical reports (from the Institute of Computing Science, Poznan University of Technology) in 1994 and also mentioned in Jaszkiewicz's PhD thesis 1995. Czyzak and Jaszkiewicz (1996) and Czyzak and Jaszkiewicz (1998) are methodological papers where the PSA method is completely described. Numerical experiments on the multiobjective knapsack problem are reported. The method is also extensively described in Jaszkiewicz (2001c). Using the multiobjective knapsack problem, Jaszkiewicz (2001a) presents a comparative experiment of various metaheuristics including the PSA method. The instances have 2, 3, and 4 objectives functions. Hapke et al. (1996), Hapke et al. (1998a) and Jaszkiewicz (1997) describe the use of the PSA method for solving practical problems: project scheduling problems, nurse scheduling, optimization of complex manufacturing systems. In Jaszkiewicz and Ferhat (1999), PSA is incorporated in a method for multiple criteria choice problems. PSA has been coupled with an interactive procedure (light beam search) in order to organize an interactive search in Hapke et al. (1998b). The PSA method has been adapted for solving fuzzy multiobjective combinatorial optimization problems, Hapke et al. (1997), Hapke et al. (2000a) and Hapke et al. (2000b). These papers report applications on fuzzy project scheduling problems with multiple objectives.

Engrand's Method by Engrand (1997), Engrand and Mouney (1998). The method is introduced during the ICONE conference and revised by Parks and Suppapitnarm (1999), Suppapitnarm and Parks (1999), Suppapitnarm et al. (2000). This method is a hybridization of simulated annealing principles with genetic algorithms. The method has been originally applied to a nuclear fuel management problem.

Engrand (1997) shows that the method does find the trade-off surface and gives performances comparable with a multiobjective GA method on some test cases. Parks and Suppapitnarm (1999) present the revision of Engrand's method (denoted here by MOSA99) and its application to the pressurized water reactor reload core design optimization problem. The main characteristic of this revised version is its ability to work without search directions, using a population of individuals to ensure the exploration of the complete trade-off surface. Each objective is considered separately. The method uses only the non-domination definition to select potentially efficient solutions, thus avoiding the management of search directions and aggregation mechanisms. Advanced strategies use the population of potentially efficient solutions to drive the approximation mechanism, thus ensuring the detection of the whole efficient frontier.

Recommendations are given about the tuning of the parameters in Suppapitnarm et al. (2000). A comparative presentation of results obtained with MOSA99 and the NSGA for an example problem given by Schaffer is reported in Suppapitnarm and Parks (1999). The MOSA99 method is described in detail and refinements are discussed in Suppapitnarm et al. (2000). These concern the solution acceptance rule, annealing schedule, constraints, and a variety of return to base strategies. The method is experimentally verified on three case studies with two and three objectives: a simple problem, a practical problem concerning the deployment system for a rigid panel on a spacecraft, and a problem of optimizing the performance of a ten-bar cantilever truss.

The Trip Planning Problem (Godart (2001)). In his PhD thesis Godart introduces the "Trip Planning Problem", as a preference-based multiobjective travelling salesman problem with activity and lodging selection. The complete formulation contains five objectives: min transportation cost, min activity cost, min lodging cost, max activity attractivity and max lodging attractivity. A simulated annealing based method, close to the MOSA method, is used to solve this combinatorial problem. Two operational modes are reported: "unidirectional" mode where only one compromise solution is computed and the "omni-directional" mode where several compromise solutions are computed.

Interactive Method for 0-1 Multiobjective Problems by Alves and

Climaco (2000). This is a general interactive method for solving 0-1 multiobjective problems where simulated annealing and tabu search work as two alternative and complementary computing procedures. It is a progressive and selective search of potentially efficient solutions by focusing the search on a sub-region delimited by information specified by the decision-maker for the objective function values. Computational results for multiple-constraint knapsack problems with two objectives are reported.

Bicriteria Scheduling Problems on a Single Machine (Koktener and Köksalan (2000)). An SA algorithm is developed and tested for bicriteria scheduling problems on a single machine using total flow-time with maximum earliness and number of tardy jobs as criteria. Various problem sets (ranges of due date, problem sizes, processing time) are used for the experiment phase. A comparison with previous known results is given.

Other Simulated Annealing-based Methods. Nam and Park (2000) method. This is another simulated annealing-based method. The authors report good results in comparison with MOEA. Applications include aircrew rostering problems, Lučić and Teodorović (1999), assembly line balancing problems with parallel workstations, McMullen and Frazier (1999), and analogue filter tuning, Thompson (2001).

8 The Tabu Search Wave

8.1 Multiobjective Tabu Search by Gandibleux et al. (1996)

In 1996 (MOPGP 96 conference, Torremolinos), Gandibleux et al. introduced a method called MOTS for MultiObjective Tabu Search (Gandibleux et al. (1997)). It is the first TS-based method designed to compute a set of potentially efficient solutions. Using a scalarizing function and a reference point, the method performs a series of tabu processes guided automatically in the objective space by the current approximation of the non-dominated frontier. Intensification, diversification and tabu daemon (usually called aspiration criteria) are designed for the multiobjective case. Two tabu memories are used, one on the decision space $TmemX$, the second on the objective space $TmemY$. The former is an attribute-based tabu list pre-

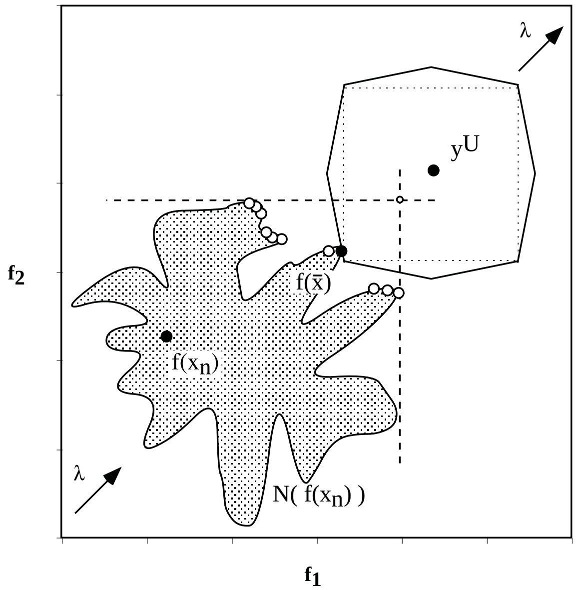

Figure 8: *Illustration of the search strategy in MOTS for the bi-objective case. Points ∘ are the potentially efficient solutions generated during an iteration.*

venting a return to already visited solutions during a tabu process. The latter is connected to the objectives and based on an improvement measure of each objective. It is used for updating weights between two consecutive tabu processes. The method is shown in Algorithm 5.

The before MOTS search strategy is encapsulated in *a tabu process*, which is composed of a series of iterations. Let us consider, at the n^{th} iteration, the current solution x_n and its (sub)neighbourhood $\mathcal{N}(x_n)$ obtained according to a suitable move $x_n \to x$ defined according to the structure of the feasible domain X (routine `exploreNeighbourhood`). The successor $\bar{x}$ of x_n for the next iteration is selected from the list of neighbour solutions $\mathcal{L} = \{x \in \mathcal{N}(x_n)\}$ as the best according to a weighted scalarizing function $S(f(x), y^U, \lambda)$:

$$S(f(x), y^U, \lambda) = \max_{1 \le k \le p} \left\{ \lambda_k \left(y_k^U - f_k(x) \right) \right\} + \rho \sum_{k=1}^{p} \lambda_k \left(y_k^U - f_k(x) \right), \rho > 0$$

The number of candidates in list $\mathcal{L}$ is limited to K solutions. The value of this parameter depends on the neighbourhood size ($1 \le K \le$

size of $\mathcal{N}(x_n)$). The reference point y^U in the scalarizing function is the locally determined utopian point $y^U = (y_1^U, \ldots, y_p^U)$ over $\mathcal{L}$, where $y_k^U > \sup\{f_k(x)|x \in \mathcal{L}\}$ (routine `utopianPoint`). This point dominates the ideal point given by the highest objective function value on each objective among the solutions in the neighbourhood of the current solution. The tabu list $TmemX$ is used to avoid cycling (routines `isTabu` and `updateTabuMemoryX`). The selected solution $\bar{x} \in \mathcal{L}$, which minimizes $S(f(x), y^U, \lambda)$ over $\mathcal{L}$ such that the move $x_n \rightarrow \bar{x}$ is not tabu, becomes the new current solution x_{n+1}, as depicted in Figure 8. The *tabu daemon* overides the tabu status of a solution $x' \in \mathcal{N}(x_n)$, when $s(f(x'), \lambda) \leq s(f(\bar{x}), \lambda) - \Delta$, with Δ being a static or dynamic threshold value (routine `TabuDaemon`). As $\mathcal{L}$ is generally a finite enumerable subset of X, the successor solution x_{n+1} can be found easily. However, the time complexity depends on the size of the neighbourhood $\mathcal{N}(x_n)$. Each iteration ends with the identification of the potentially efficient solutions *pe* in $\mathcal{L}$, which represents a local approximation of the non-dominated frontier (routine `archive`).

One tabu process is an intensification mechanism in which the effort spent is dynamically adjusted according to *pe*, the approximation obtained, in comparison with the global approximation PE. The first adjustment is based on a strategy promoting a "promising candidate". If $\bar{x} \in pe$ seems to be a promising solution, then the current search is increased. The promising characterisation of $\bar{x}$ is evaluated by regarding the number of solutions of PE that were dominated by $\bar{x}$ (routine `isPromising`). The second adjustment is based on the dectection of a "sterile search". When *pe* contains no potentially efficient solution according to PE (routine `isSterile`), the current neighbourhood is said to be sterile. This means that the neighbourhood is not promising in terms of efficient solutions, and the interest in the current search process is decreased.

The scalarazing function leads to the exploration of the non-dominated frontier in the direction given by the weight vector λ. The diversification strategy updates the weights periodically and automatically so that the importance of objectives that have been significantly improved is decreased (routine `setSearchDirection`). To perform this diversification strategy, a pseudo-criterion preference model is suggested. Using threshold parameters for each objective, which measure the level of improvement $f_k(\bar{x}) - f_k(x)$ according to the objective k, one of the following four scenarios is determined: 1) the performance improvement for objective k is worse (D, dominated); 2) improvement is too weak to be interesting (I, indifferent); 3) improvement is significant (S, weak improvement) or 4) improvement is strong (P, strong improvement). When the objective k is significantly or strongly improved, the weight λ_k is either slightly or strongly decreased, whereas the weights related to the others cases are increased. To ensure that all the areas of the non-dominated frontier are visited, a tabu list with respect to the objectives is introduced ($TmemY$). In the case S (respectively case P), the objective k is declared tabu for a predetermined number of iterations. Thus, its weight cannot be increased as long as its tabu status is true. Obviously, the diversification scheme is simpler in the case of biobjective problems. A deterministic and myopic design of weights $\lambda \in \Lambda$ can be performed, which provides a suitable description of the search direction.

MOTS is a generic method, rather than a ready-to-use technique. All of its primitives need to be stated in a suitable manner, according to the MOCO problem to be solved.

8.2 Bibliographic Notes

The extension of TS to multiobjective programming is recent in comparison with the other metaheuristics. Hybrid TS-based methods have been proposed, in an effort to improve solution diversification along the non-dominated frontier. Some ideas come from MOEA, like the use of a population (Hansen (2000)), or a combination of tabu search with genetic algorithms (Ben Abdelaziz et al. (1999)). Multiobjective tabu search procedures have been applied mainly to MOCO problems, particularly the knapsack problem. The following MOMH, based on TS, can be found in the literature.

Multiobjective Tabu Search (MOTS) by Gandibleux, Mezdaoui and Fréville, 1996 (Gandibleux et al. (1997)). This method was first instantiated on an unconstrained permutation problem, and then later on the biobjective knapsack problem, Gandibleux and Fréville (2000), in combination with bounds to reduce the search space.

Sun's Method, Sun (1997). An interactive procedure using tabu search for general multiple objective combinatorial optimization problems, the procedure is similar to the Combined Tchebycheff/Aspiration Criterion Vector Method, Steuer et al. (1993). The tabu search is used to solve subproblems in order to find potentially efficient solutions.

Algorithm 5 MOTS, MultiObjective Tabu Search

input :	TTX,	tabu tenure on decision space
	TTY, PSI,	tabu tenure and thresholds on objective space
	Δ, K,	daemon activation parameter and size of candidate list
	$IterInit$,	initial intensity level for a tabu process
	$stopConditions$,	conditions to stop the search
output :	PE,	the set of potentially efficient solutions

```
begin MOTS
  TmemX, TmemY ← ∅
  n ← 1;   select x_n ∈ X;   PE ← {x_n}

  --| Diversification in Z
  repeat  λ ← setSearchDirection( TmemY, PE )
          Iter ← IterInit

          --| Tabu process
          repeat  --| Local exploration of the non-dominated frontier
                  L ← exploreNeighbourhood( x_n )
                  z̃ ← utopianPoint( L )

                  --| Selection of the successor solution
                  x̄ ← arg min{s(f(x), y^U, λ) | x ∈ L,
                          ¬isTabu ( move(x_n → x), TmemX )}
                  x* ← TabuDaemon( x̄, L, TmemX )
                  if x* = ∅ then x_{n++} ← x̄ else x_{n++} ← x* endif
                  TmemX ←
                    updateTabuMemoryX( move(x_{n−1} → x_n), TmemX )

                  --| Dynamic adjustment of the intensification
                  --| and update of the approximation
                  pe ← {x_n}; ∀x ∈ L : pe ← archive( pe, x )
                  if isPromising( x_n, PE ) then increase Iter endIf
                  if isSterile( pe, PE ) then decrease Iter endIf
                  PE ← merge( PE, pe ) ;  Iter ← Iter − 1

          until Iter = 0   --| iterations are performed
  until stopConditions are fulfilled
end MOTS
```

The principles used for designing the TS search strategy are similar to those defined for MOTS. This method has been used for facility location planning, Agrell et al. (1999).

Multiobjective Tabu Search (MOTS97) by Hansen (2000). This method uses a set of "generation solutions", each with its own tabu list. These solutions are dispersed throughout the objective space in order to allow searches in different areas of the non-dominated frontier. Weights are defined for each solution with the aim of forcing the search into a certain direction of the non-dominated frontier and away from other current solutions that are efficient with respect to it. Diversification is ensured by a set of generation solutions and a drift criterion. Results for the knapsack problem are available, and also for the resource constrained project scheduling problem in Viana and Sousa (2000).

Ben Abdelaziz, Chaouachi and Krichen's Hybrid Method, Ben Abdelaziz et al. (1999). The authors present a multiobjective hybrid heuristic for the knapsack problem. The method is a mix of a tabu search and a genetic algorithm.

Baykasoglu, Owen and Gindy's Method, Baykasoglu et al. (1999). A candidate list provides an opportunity to diversify the search. The method is designed to handle any type of variable (integer, zero-one, continuous and mixed), and has been used for goal programming problems, Baykasoglu (2001a) and Baykasoglu (2001b).

Other tabu search-based methods have been developed for scheduling problems (Loukil Moalla et al. (2000a)) and the trip planning problem, Godart (2001). A hybrid and interactive solution process based on SA and TS is proposed in Alves and Climaco (2000).

9 The Ant Colony Optimization Wave

Recently the Ant Colony System (ACS) paradigm, a population based algorithm, has been adapted to multiobjective optimization. Ant colony optimization techniques are inspired by the behavior of real ants foraging for food. The key to the effectiveness of a colony of ants in finding short paths to a food source is a chemical substance called pheromone. It provides the ants with the ability to communicate. Ants will initially move along random directions depositing pheromone along their paths. When an ant finds a food source it returns to the nest. An ant on a shortest path will return to the nest first, thus more pheromone will be deposited on the shortest paths. Because a moving ant will choose a path with a probability that depends on the amount of pheromone detected, paths that are more frequently travelled become more attractive, and over time the shortest paths will be used most often. Also, the pheromone evaporates over time, so that pheromone trails of infrequently travelled paths become weaker and those of frequently travelled shortest paths are reinforced.

The use of artificial ant colony systems has first been proposed by Dorigo and co-authors. Details of single objective optimisation techniques based on ant colonies can be found, e.g. in Dorigo (1992), Dorigo et al. (1996), and Dorigo et al. (1997). In recent years (since 1999) ant colony optimization technique have also been proposed for multiobjective combinatorial optimization. Because of the way ant colony systems work, their application is most attractive for problems for which solutions can be constructed sequentially, and all applications in the MOCO area to date are of such a type: TSP, VRP, sequencing and scheduling problems, and portfolio selection problems.

9.1 Multiobjective Ant Colony Optimization Algorithms

Multiobjective ant colony optimization algorithms have been proposed for problems with a weighted sum of objectives, Doerner et al. (2001b), a hierarchy on the objectives (lexicographic optimization, Gambardella et al. (1999), Gravel et al. (2002), T'Kindt et al. (2002)) and in the Pareto optimality sense, Doerner et al. (2004), Iredi et al. (2001), Mariano and Morales (1999b), McMullen (2001), and Shelokar et al. (2000). They either use a single colony (Doerner et al. (2004), Gravel et al. (2002), Iredi et al. (2001), McMullen (2001), Shelokar et al. (2000), T'Kindt et al. (2002)) or multiple colonies (Doerner et al. (2001b), Gambardella et al. (1999), Iredi et al. (2001), Mariano and Morales (1999b)). Most of the algorithm use both heuristic information, which is problem specific, and pheromone information to calculate the probability for making the next step in the solution construction procedure. As a typical example of a multicolony system for finding efficient solutions we present an algorithm adapted from

Iredi et al. (2001), see Algorithm 6.

Algorithm 6 Multicolony Ant Algorithm by Iredi et al. (2001)

input : m, a, the size of all colonies and one colony
α, β, weight of pheromone and heuristic information
output : PE, the set of potentially efficient solutions

```
begin MCAA

  PE ← ∅
  repeat   for colony l = 1, ..., m/a  loop
               initialisePheromone(τ)
               initialiseHeuristic(η)

               for ant k = 1, ..., a  loop
                   assignWeight(k, l, λ)
                   constructSolution(k, l, p(λ, α, β), τ, η, x_k^l)
                   localPheromoneUpdate(τ, x_k^l)
               end for

           end for
           PE ← nonDominated(PE, ∪_{k,l}{x_k^l})
           globalPheromoneUpdate(τ, ∪_{k,l}{x_k^l}, PE)

  until stoppingCriterion

end MCAA
```

The details of the algorithm are problem dependent. In the scheduling problem discussed in Iredi et al. (2001), pheromone and heuristic information is stored in square matrices, where the indices of η_{ij} and τ_{ij} refer to having job j in position i in the sequence. Heuristic information is computed according to the objectives and remains unchanged during the algorithm. Initially, $\tau_{ij} = \tau_0$. Let m be the total number of ants and a be the number of ants per colony.

Several methods for the choice of weights λ ant k of colony l uses for the first (of two) objectives are proposed.

$$\begin{aligned}
\texttt{assignweight}(k,l,\lambda) &= \frac{k-1}{(m/a)-1} \\
\texttt{assignweight}(k,l,\lambda) &= \frac{l-1}{a} + \frac{k}{m} \\
\texttt{assignweight}(k,l,\lambda) &= \frac{i-1}{a+1} + \frac{2(k-1)}{(a+1)((m/a)-1}.
\end{aligned}$$

In the first case, the ants in each colony use the same weights, in the second the weight intervals for the colonies have only the boundaries in common, wheras in the last the weight interval of colony l intersects with those of colony $l-1$ and $l+1$.

The probability by which job j is chosen to be in position i is given by

$$p_{ij} = \frac{{\tau^1}_{ij}^{\lambda\alpha}{\tau^2}_{ij}^{(1-\lambda)\alpha}{\eta^1}_{ij}^{\lambda\beta}{\eta^2}_{ij}^{(1-\lambda)\beta}}{\sum_{h\in S}{\tau^1}_{ih}^{\lambda\alpha}{\tau^2}_{ih}^{(1-\lambda)\alpha}{\eta^1}_{ih}^{\lambda\beta}{\eta^2}_{ih}^{(1-\lambda)\beta}},$$

where α and β are weights of pheromone and heuristic information, respectively.

All ants that generated a solution in PE in the current iteration contribute to the pheromone update. First τ_{ij} is set to $(1-\rho)\tau_{ij}$ for all (i,j) (evaporation). Then $\tau_{ij}^h \leftarrow (1-\rho)\tau_{ij}^h + 1/r$, for all (i,j) appearing in the r updating ants. Since each colony maintains its own pheromone matrices, the update can either be done by colony or by region in PE. No local pheromone update is done.

The algorithm of Doerner et al. (2004) for portfolio selection also fits in this framework. Here $p=1$, pheromone is initialised to $\tau_0 > 0$, heuristic information $\eta_i(x)$ is based on feasibility, depends on the partial solution constructed so far, and is updated during the algorithm. The authors use a random assignment of weights to cope with the many objectives.

Local pheromone update is $\tau_i^k \leftarrow (1-\rho)\tau_i^k + \rho\tau_0$ for objective k and project i, if project i has been added to a portfolio by an ant. The ants who contributed a solution to PE in the current iteration update the pheromone globally using $\tau_i^k \leftarrow (1-\rho)\tau_i^k + \rho\Delta_i^k$, where Δ_i^k is 10, if project i appears in the best solution for objective k and 5 if it is in the second best solution.

The solution is constructed by adding project i to the portfolio if a uniformly distributed random number q is not bigger than a parameter q_0 and $\sum_{k=1}^{p}(\lambda_k \tau_i^k)^\alpha)(\eta_i(x))^\beta$ is maximal among all projects that can be added to the partial solution x and have $\eta_i(x) > 0$. Otherwise the project to be added is chosen randomly with probability

$$p_i = \frac{\sum_{k=1}^{p}(\lambda_k \tau_i^k)^\alpha \eta_i(x)^\beta}{\sum h \in S(x) \sum_{k=1}^{p}(\lambda_k \tau_h^k)^\alpha \eta_h(x)^\beta}.$$

9.2 Bibliographic Notes

Shelokar et al. (2000), Shelokar et al. (2002), Shelokar et al. (2003): In Shelokar et al. (2000) an ant algorithm for multiobjective continuous optimization is proposed. An interesting feature is that it combines the ant system methodology with the strength Pareto fitness assignment of Zitzler and Thiele (1998) and clustering methods. The algorithm is applied to reliability engineering problems in Shelokar et al. (2002) and to the optimization of reactor regenerator systems in Shelokar et al. (2003).

McMullen (2001): In this paper a just-in-time sequencing problem with the objective of minimizing the number of setups and minimizing the usage rates of raw materials is addressed. The problem is reformulated as a TSP by spatialising the data and applying a standard single objective ant colony algorithm.

Gambardella et al. (1999): Here, a bicriteria vehicle routing problem with time windows with the lexicographically sorted objectives of minimizing the number of vehicles and minimizing total travel distance is solved by a multicolony ant system called MACS. The algorithm uses one colony for each objective, local and global pheromone update. The colonies cooperate through the use of the global best solution for the global pheromone update. Local search is applied to improve the quality of each solution found. Numerical tests show that the algorithm improves some of the best known solutions of test problems from the literature.

Mariano and Morales (1999a), Mariano and Morales (1999b): The algorithm MOAQ uses one ant colony for each objective function. All colonies have the same number of ants. (Partial) solutions of each colony are used in the next colony. The algorithm is applied to two literature problems (and compared with VEGA (Schaffer (1984)). The research is motivated by the real world problem of designing a water distribution network for irrigation to minimise network cost and maximise profit.

Iredi et al. (2001): The authors propose a number of ant colony optimization algorithms to solve bicriteria combinatorial problems, including ones with single and multiple colonies. Various methods for pheromone update and weight assignment (in order to browse the whole non-dominated frontier) are proposed and tested on a single machine scheduling problem to minimise total tardiness and changeover cost. Numerical tests are presented.

Doerner et al. (2001a), Doerner et al. (2002), Doerner et al. (2003), Doerner et al. (2004): The problem framework is that of a multiobjective portfolio selection problem. The authors consider a rather large number of objectives $p = (B + R)T$, where B is the number of benefit categories, R is the number of resources, and T is the planning period. They use one colony and a random selection of weights of objectives for each ant. The global update considers the best and second best solutions for each objective found in the current iteration. The results on test problems are compared with NSGA (Srinivas and Deb (1994)), PSA (Czyzak and Jaszkiewicz (1996)), and the true efficient set (for small problems).

Doerner et al. (2001b): The authors solve a special case of the pickup and delivery problem with the linearly combined objectives of total number of vehicles and empty vehicle movements by a multicolony approach. The colonies use different heuristic information, and their sizes change during the algorithm.

T'Kindt et al. (2002): A (single) ant colony optimization approach is proposed for a two machine bicriteria flowshop problem to minimize makespan and total flowtime in a lexicographic sense. The solutions produced by the ants are improved by local search.

Gravel et al. (2002): The problem of sequencing orders for the casting of aluminium provides the background in this paper. Four objectives are considered in a lexicographic sense. A distance function based on penalties for bad performance is used to translate the problem into a TSP setting. Global pheromone update considers only the primary objective.

10 Other Approaches and the Trend

In addition to the previously discussed multiobjective versions of the now classic metaheuristics (GA, SA, and TS), and the now very popular ant colony systems, there are other MOMH. Several of them have been published most recently or have until now been discussed only during international conferences:

- The first works using Artificial Neural Networks (ANN) to solve MOP have been published (Malakooti et al. (1990), Sun et al. (1996), Sun et al. (2000)) at the beginning of the Nineties. However, the ANN approach remains marginal.
- Adaptations of metaheuristics, such as the Greedy Randomized Adaptive Search Procedure GRASP, Gandibleux et al. (1998), and Scatter Search, Beausoleil (2001), have been presented at recent international conferences.
- Papers concerning a dedicated heuristic, Köksalan (1999), a stochastic search method (Sysoev and Dolgui (1999)) and a comparison of neighbourhood search techniques for MOP, Marett and Wright (1996), are also worthy of mention.

Combinatorial optimisation is rich in term of results. It is natural to bridge between the known theoretical results for single objective combinatorial problems and MOCO. Managed inside a MOMH, this information can be used advantageously to enforce the aptitude of approximation methods. For example, adding an additional constraint to the unidimensional biobjective knapsack problem will help to reduce the search space, Gandibleux and Fréville (2000).

Another promising way is to design MOMH components that take the specific aspect of MOCO into account. In Gandibleux et al. (2004), similarities in solutions and subsets of exact solutions are used advantageously by the components of an evolutionary method. Here, interesting performance results are measured with a *path relinking* operator, Glover and Laguna (1997), given a subset of optimal solutions (or approximations) in the initial population. Path-relinking generates new solutions by exploring the trajectories that connect good solutions. A path-relinking operation starts by randomly selecting I_A (the initiating solution) and I_B (the guiding solution), two individuals from the current population (Figure 9). The path-relinking operation generates a path $I_A(= I_0), I_1, \ldots, I_B$ through the neighbourhood space, such that the distance between I_i and I_B decreases monotonically in i, where the distance is defined as the number of positions for which different values are assigned in I_i and I_B.

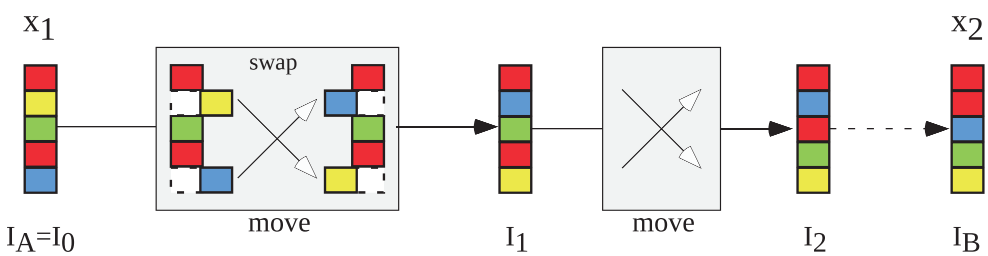

Figure 9: *Path-relinking operator principle.*

Although many such paths may possibly exist, one path is chosen using, for example, random moves based on a swap operator. Such randomness introduces a form of diversity to the solutions generated along the path. For every intermediate solution I_i, a single solution is generated in the neighbourhood (Figure 10). This principle has been successfully implemented for computing the complete non-dominated frontier of assignment problems with two objectives. Figure 11 provides a sample of these results. Using the same numerical instances, this population-based method based on specific operators outperforms MOSA (Tuyttens et al. (2000)).

MOCO problems rarely tackled hitherto are now investigated with MOMH. These include timetabling problems (Paquete and Fonseca (2001)), space allocation problems (Pires et al. (2001)), multi-period distribution management problems (Ribeiro and Lourenço (2001)), and vehicle routing problems (Geiger (2001) and Rahoual et al. (2001)), to name a few.

The last aspect of research concerns issues related to computer implementation. Efficient data structures, such as the quad-tree, have proven

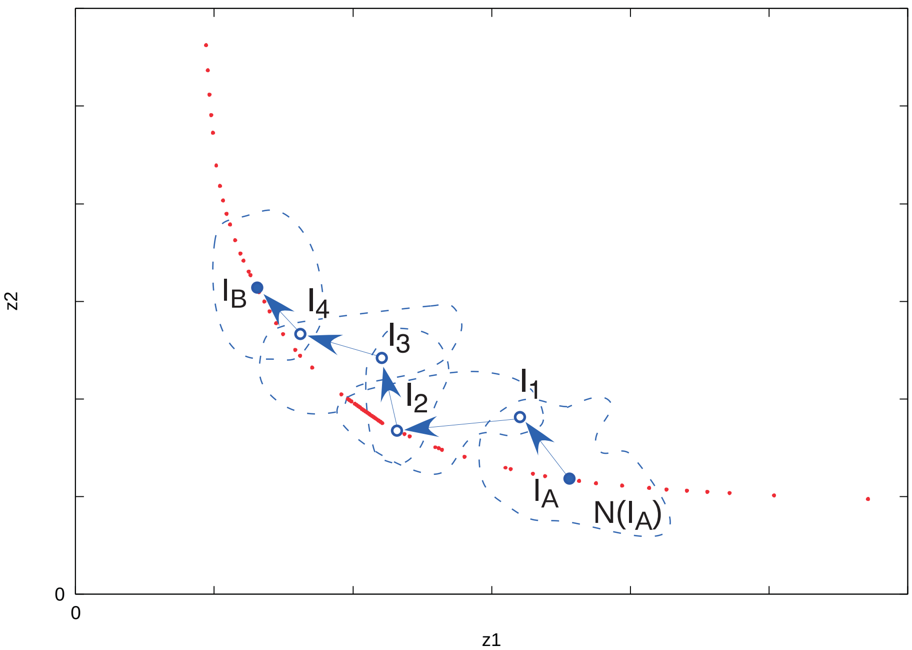

Figure 10: *Illustration of a possible path construction. I_A and I_B are two individuals ramdomly selected from the current population (small bullets). I_A is the initiating solution, I_B is the guiding solution. $\mathcal{N}(I_A)$ is the feasible neighbourhood according to the move defined. $I_A - I_1 - I_2 - I_3 - I_4 - I_B$ is the path that is built.*

their efficiency for managing non-dominated criterion vectors, Habenicht (1983) and Sun and Steuer (1996). Reusable software, such as object-oriented frameworks for multiobjective local search, is under development, Claro and Sousa (2001).

These new trends promise many future papers.

References

Agrell P., Sun M. and Stam A. (1997). A tabu search multi-criteria decision model for facility location planning. Proceedings of the 1997 DSI Annual Meeting, San Diego, California, volume 2, 908–910. Decision Sciences Institute, Atlanta.

Alves M.J. and Climaco J. (2000). An interactive method for 0-1 multiobjective

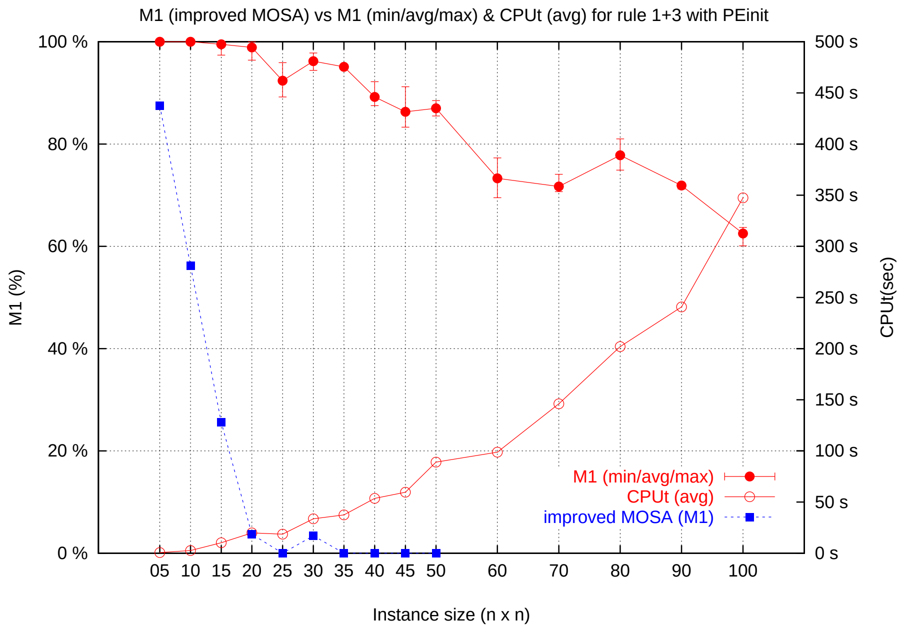

Figure 11: *Approximations obtained with the MOSA method and a population-based method for the assignment problem with two objectives. The comparison is based on the performance measure M_1, Ulungu (1993), which measures the percentage of exact solutions included in the final approximation set.*

problems using simulated annealing and tabu search. *Journal of Heuristics* 6, 385–403.

Andersen K.A., Jørnsten K. and Lind M. (1996). On bicriterion minimal spanning trees: An approximation. *Computers and Operations Research* 23, 1171–1182.

Ausiello G., Crescenzi P., Gambosi G., Kann V., Marchetti-Spaccamela A. and Protasi M. (1999). *Complexity and Approximation – Combinatorial Optimization Problems and Their Approximability Properties.* Springer Verlag.

Bagchi T.P. (1999). *Multiobjective Scheduling by Genetic Algorithms.* Kluwer Academic Publishers.

Barichard V. and Hao J.K. (2002). Un algorithme hybride pour le problème de sac à dos multi-objectifs. JNPC'2002 Proceedings: Huitièmes Journées Nationales sur la Résolution Pratique de Problèmes NP-Complets, Nice, France, 27–29 May 2002, 19–30.

Baykasoglu A. (2001a). Goal programming using the multiple objective tabu search. *Journal of the Operational Research Society* 52, 1359–1369.

Baykasoglu A. (2001b). MOAPPS 1.0: Aggregate production planning using the multiple objective tabu search. *International Journal of Production Research* 39, 3685–3702.

Baykasoglu A., Owen S. and Gindy N. (1999). A taboo search based approach to find the Pareto optimal set in multiple objective optimisation. *Journal of Engineering Optimization* 31, 731–748.

Beausoleil R. (2001). Multiple criteria scatter search. In: Sousa J.P. de (ed.), *MIC'2001 Proceedings of the 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 2, 539–543.

Ben Abdelaziz F., Chaouachi J. and Krichen S. (1999). A hybrid heuristic for multiobjective knapsack problems. In: Voss S., Martello S., Osman I. and Roucairol C. (eds.), *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*. Kluwer Academic Publishers, 205–212.

Burkard R.E., Rote G., Ruhe G. and Sieber N. (1989). Algorithmische Untersuchungen zu bikriteriellen kostenminimalen Flüssen in Netzwerken. *Wissenschaftliche Zeitung der Technischen Hochschule Leipzig* 13, 333–341.

Claro J. and Sousa J.P. de (2001). An object-oriented framework for multiobjective local search. In: Sousa J.P. de (ed.), *MIC'2001 Proceedings of the 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 1, 231–236.

Coello C.A., Van Veldhuizen D. and Lamont G. (2002). *Evolutionary Algorithms for solving multi-objective problems*. Kluwer Academic Publishers.

Coello C.A. (1996). An Empirical Study of Evolutionary Techniques for Multiobjective Optimization in Engineering Designe. Ph.D. Dissertation, Tulane University.

Coello C.A. (1999). A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information Systems* 1, 269–308.

Coello C.A. (2000). An updated survey of GA-based multiobjective optimization techniques. *ACM Computing Surveys* 32, 109–143.

Coello C.A. (2004). List of references on evolutionary multiobjective optimization. http://www.lania.mx/~ccoello/EMOO/.

Corley H.W. (1985). Efficient spanning trees. *Journal of Optimization Theory and Applications* 45, 481–485.

Czyzak P. and Jaszkiewicz A. (1996). A multiobjective metaheuristic approach to the localization of a chain of petrol stations by the capital budgeting model. *Control and Cybernetics* 25, 177–187.

Czyzak P. and Jaszkiewicz A. (1997). Pareto simulated annealing. In: Fandel G.

and Gal T. (eds.), *Multiple Criteria Decision Making*, Proceedings of the XIIth International Conference, Hagen (Germany), Lecture Notes in Economics and Mathematical Systems 448, 297–307.

Czyzak P. and Jaszkiewicz A. (1998). Pareto simulated annealing – A metaheuristic technique for multiple objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis* 7, 34–47.

Dahl G., Jörnsten K. and Lokketangen A. (1995). A tabu search approach to the channel minimization problem. Proceedings of the International Conference on Optimization Techniques and Applications (ICOTA'95), 5-8 July 1995, Chengdu, China, 369–377.

Mira de Fonseca C. M. (1995). Multiobjective Genetic Algorithms with Applications to Control Engineering Problems. Ph.D. Dissertation, University of Sheffield.

Deb K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley.

Delorme X., Gandibleux X. and Rodriguez J. (2003). Résolution d'un problème d'évaluation de capacité d'infrastructure ferroviaire. Actes du colloque sur l'innovation technologique pour les transports terrestres (TILT) 2, 647–654. GRRT, Lille, France.

Doerner K., Gutjahr W.J., Hartl R.F., Strauss C. and Stummer C. (2001a). Ant colony optimization in multiobjective portfolio selection. In: Sousa J.P. de (ed.), *MIC'2001 Proceedings of the 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 1, 243–248.

Doerner K., Hartl R.F. and Reimann M. (2001b). Are COMPETants more competent for problem solving? The case of a multiple objective transportation problem. Technical Report 50, Department for Production and Operations Management, University of Vienna.

Doerner K., Gutjahr W.J., Hartl R.F., Strauss C. and Stummer C. (2002). Investitionsentscheidungen bei mehrfachen Zielsetzungen und künstliche Ameisen. In: Chamoni P., Leisten R., Martin A., Minnemann J. and Stadtler H. (eds.), *Operations Research Proceedings 2001, Selected Papers of OR 2001*. Springer Verlag, 355–362.

Doerner K., Gutjahr W.J., Hartl R.F., Strauss C. and Stummer C. (2003). Pareto ant colony optimization in multiobjective portfolio selection with LP preprocessing. Technical Report 2003-06, Center for Business Studies, University of Vienna.

Doerner K., Gutjahr W.J., Hartl R.F., Strauss C. and Stummer C. (2004). Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection. *Annals of Operations Research* (to appear).

Dorigo (1992). *Optimization, Learning and Natural Algorithms*. Ph.D. Dissertation, Politecnico di Milano. (in Italian).

Dorigo M., Colorni A. and Maniezzo V. (1996). The ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics Part B* 26, 29–41.

Dorigo M., Di Caro G. and Gambardella L.M. (1997). Ant algorithms for discrete optimization. *Artificial Life* 5, 137–172.

Ehrgott M. (1999). Integer solutions of multicriteria network flow problems. *Investigação Operacional* 19, 229–243.

Ehrgott M. (2000). Approximation algorithms for combinatorial multicriteria optimization problems. *International Transactions in Operational Research* 7, 5–31.

Ehrgott M. and Gandibleux X. (2000). A survey and annotated bibliography of multiobjective combinatorial optimization. *OR Spektrum* 22, 425–460.

Ehrgott M. and Gandibleux X. (2001). Bounds and bound sets for biobjective combinatorial optimization problems. In: Koksalan M. and Zionts S. (eds.), *Multiple Criteria Decision Making in the New Millennium*, Lecture Notes in Economics and Mathematical Systems 507, 241–253. Springer Verlag.

Ehrgott M. and Gandibleux X. (2004). Bound sets for biobjective combinatorial optimization problems. Technical report, Department of Engineering Science, The University of Auckland.

Ehrgott M., Klamroth K. and Schwehm S. (2004). An MCDM approach to portfolio optimization. *European Journal of Operational Research* 155, 752–770.

Ehrgott M. and Ryan D.M. (2002). Constructing robust crew schedules with bicriteria optimization. *Journal of Multi-Criteria Decision Analysis* 11, 139–150.

Ehrgott M. and Tenfelde-Podehl D. (2003). Computation of ideal and nadir values and implications for their use in MCDM methods. *European Journal of Operational Research* 151, 119–131.

Ehrgott M. and Wiecek M.M. (2004). Multiobjective Programming. In: Figueira J., Greco S. and Ehrgott M. (eds.), *Multiple Criteria Decision Analysis: State of the Art Surveys*. Kluwer Academic Publishers.

El-Sherbeny N. (2001). Resolution of a vehicle routing problem with a multiobjective simulated annealing method. Ph.D. Dissertation, Université de Mons-Hainaut.

Engrand P. (1997). A multi-objective approach based on simulated annealing and its application to nuclear fuel management. Proceedings of the 5th ASME/

SFEN/JSME International Conference on Nuclear Engineering. Icone 5, Nice, France, 416–423.

Engrand P. and Mouney X. (1998). Une méthode originale d'optimisation multi-objectif. Technical Report 98NJ00005, EDF-DER Clamart, France.

Erlebach T., Kellerer H. and Pferschy U. (2001). Approximating multi-objective knapsack problems. In: Dehne F., Sack J.R. and Tamassia R. (eds.) *Algorithms and Data Structures. 7th International Workshop, Providence, RI, August 8-10, 2001*, Lecture Notes in Computer Science 2125. Springer Verlag, 210–221.

Erlebach T., Kellerer H. and Pferschy U. (2002). Approximating multiobjective knapsack problems. *Management Science* 48, 1603–1612.

Fernández E. and Puerto J. (2000). Multiobjective solution of the uncapacitated plant location problem. *European Journal of Operational Research* 145, 509–529.

Fonseca C.M., Fleming P.J., Zitzler E., Deb K. and Thiele L. (2003). *Evolutionary Multi-Criterion Optimization. EMO 2003, Second International Conference, Faro, Portugal, April 2003 Proceedings*, Lecture Notes in Computer Sciences 2632. Springer Verlag.

Fonseca C.M. and Fleming P.J. (1993). Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In: Forrest S. (ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms, San Mateo, California, 1993. University of Illinois at Urbana-Champaign.* Morgan Kaufman, 416–423.

Fonseca C.M. and Fleming P.J. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary Computation* 3, 1–16.

Fourman M.P. (1985). Compaction of Symbolic Layout using Genetic Algorithms. In: Grefenstette J.J. (ed.), *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms.* Lawrence Erlbaum, 141–153.

Fruhwirth B., Burkard R.E. and Rote G. (1989). Approximation of convex curves with application to the bicriterial minimum cost flow problem. *European Journal of Operational Research* 42, 326–388.

Gambardella L.M., Taillard E. and Agazzi G. (1999). MACS-VRPTW: A multiple ant colony system for vehicle routing problems with time windows. In: Corne D., Dorigo M. and Glover F. (eds.), *New Ideas in Optimization.* McGraw-Hill, 63–76.

Gandibleux X. and Fréville A. (2000). Tabu search based procedure for solving the 0/1 multiobjective knapsack problem: The two objective case. *Journal of*

Heuristics 6, 361–383.

Gandibleux X., Mezdaoui N. and Fréville A. (1997). A tabu search procedure to solve multiobjective combinatorial optimization problems. In: Caballero R., Ruiz F. and Steuer R. (eds.), *Advances in Multiple Objective and Goal Programming*, Lecture Notes in Economics and Mathematical Systems 455. Springer Verlag, 291–300.

Gandibleux X., Morita H. and Katoh N. (1998). A genetic algorithm for 0-1 multiobjective knapsack problem. Proceedings of the International Conference on Nonlinear Analysis and Convex Analysis (NACA98), July 28-31, 1998, Niigata, Japan.

Gandibleux X., Morita H. and Katoh N. (2001). The supported solutions used as a genetic information in a population heuristic. In: Zitzler E., Deb K., Thiele L., Coello Coello C.A. and Corne D. (eds.), *First International Conference on Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science 1993. Springer Verlag, 429–442.

Gandibleux X., Morita H. and Katoh N. (2004). A population-based metaheuristic for solving assignment problems with two objectives. *Journal of Mathematical Modelling and Algorithms* (to appear).

Gandibleux X., Vancoppenolle D. and Tuyttens D. (1998). A first making use of GRASP for solving MOCO problems. Technical report, University of Valenciennes, France, 1998.

Geiger M. (2001). Genetic algorithms for multiple objective vehicle routing. Sousa J.P. de (ed.), *MIC'2001 Proceedings of the 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 1, 349–353.

Gen M. and Li Y.Z. (1998a). Solving multi-objective transportation problems by spanning tree-based genetic algorithm. In: Parmee I.C. (ed.), *Adaptive Computing in Design and Manufacture: The Integration of Evolutionary and Adaptive Computing Technologies with Product/System Design and Realisation*. Springer Verlag, 95–108.

Gen M. and Li Y.Z. (1998b). Spanning tree based genetic algorithm for bicriteria transportation problem. *Computers and Industrial Engineering* 35, 531–534.

Glover F. and Laguna M. (1997). *Tabu Search*. Kluwer Academic Publishers.

Godart J.M. (2001). Problèmes d'optimisation combinatoire à caractère économique dans le secteur du tourisme (organisation de voyages). Ph.D. Dissertation, Université de Mons-Hainaut.

Goldberg D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Co.

Gravel M., Price W.L. and Gagné C. (2002). Scheduling continuous casting of aluminium using a multiple objective ant colony optimization metaheuristic. *European Journal of Operational Research* 143, 218–229.

Grefenstette J.J. (1984). GENESIS: A system for using genetic search procedures. Proceedings of the 1984 Conference on Intelligent Systems and Machines, 161–165.

Habenicht W. (1983). Quad trees – A datastructure for discrete vector optimization problems. In: Hansen P. (ed.), *Essays and Surveys on Multiple Criteria Decision Making*, Lecture Notes in Economics and Mathematical Systems 209. Springer Verlag, 136–145.

Hajela P. and Lin C.Y. (1992). Genetic search strategies in multicriterion optimal design. *Structural Optimization* 4, 99–107.

Hamacher H.W. and Ruhe G. (1994). On spanning tree problems with multiple objectives. *Annals of Operations Research* 52, 209–230.

Hansen M.P. (1998). Metaheuristics for multiple objective combinatorial optimization. Ph.D. Dissertation, Technical University of Denmark, Report IMM-PHD-1998-45.

Hansen M.P. (2000). Tabu search for multiobjective combinatorial optimization: TAMOCO. *Control and Cybernetics* 29, 799–818.

Hansen P. (1979). Bicriterion path problems. In: Fandel G. and Gal T. (eds.), *Multiple Criteria Decision Making Theory and Application*, Lecture Notes in Economics and Mathematical Systems 177. Springer Verlag, 109–127.

Hapke M., Jaszkiewicz A. and Slowinski R. (1996). Interactive analysis of multiple-criteria project scheduling problems. Proceedings of the The Fifth International Workshop on Project Management and Scheduling – EURO PMS'96, Poznan, Poland, 107–110.

Hapke M., Jaszkiewicz A. and Slowinski R. (1997). Fuzzy project scheduling with multiple criteria. Proceedings of Sixth IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'97, July 1-5, Barcelona, Spain, 1277–1282.

Hapke M., Jaszkiewicz A. and Slowinski R. (1998a). Fuzzy multi-mode resource-constrained project scheduling with multiple objectives. In: Weglarz J. (ed.) *Recent Advances in Project Scheduling.* Kluwer Academic Publishers, 355–382.

Hapke M., Jaszkiewicz A. and Slowinski R. (1998b). Interactive analysis of multiple-criteria project scheduling problems. *European Journal of Operational Research* 107, 315–324.

Hapke M., Jaszkiewicz A. and Slowinski R. (2000a). Pareto simulated annealing for fuzzy multi-objective combinatorial optimization. *Journal of Heuristics* 6, 329–354.

Hapke M., Kominek P., Jaszkiewicz A. and Slowinski R. (2000b). Integrated tools for software project scheduling under uncertainty. In: Brucker P., Heitmann S., Hurink J. and Knust S. (eds.), *Proceedings of the 7th International Workhop on Project Management and Scheduling PMS'2000, Osnabrück, Germany, April 17-19*, 149–151.

Hertz A., Jaumard B., Ribeiro C. and Formosinho Filho W. (1994). A multi-criteria tabu search approach to cell formation problems in group technology with multiple objectives. *RAIRO – Recherche Opérationnelle* 28, 303–328.

Horn J., Nafpliotis N. and Goldberg D.E. (1994). A niched Pareto genetic algorithm for multiobjective optimization. Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, Orlando, FL, 29 June - 1 July 1994, volume 1, 82–87.

Iredi S., Merkle D. and Middendorf M. (2001). Bi-criterion optimization with multi colony ant algorithms. In: Zitzler E., Deb K., Thiele L., Coello Coello C.A. and Corne D. (eds.), *First International Conference on Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science 1993. Springer Verlag, 359–372.

Jaszkiewicz A. (1997). A metaheuristic approach to multiple objective nurse scheduling. *Foundations of Computing and Decision Sciences Journal* 22, 169–184.

Jaszkiewicz A. (1998). Genetic local search for multiple objective combinatorial optimization. Working paper RA-014/98, Institute of Computing Science, Poznan University of Technology, Poland.

Jaszkiewicz A. (2001a). Comparison of local search-based metaheuristics on the multiple objective knapsack problem. *Foundations of Computing and Decision Sciences Journal* 26, 99–120.

Jaszkiewicz A. (2001b). Multiple objective genetic local search algorithm. In: Köksalan M. and Zionts S. (eds.) *Multiple Criteria Decision Making in the New Millennium*, Lecture Notes in Economics and Mathematical Systems 507, 231–240.

Jaszkiewicz A. (2001c). Multiple objective metaheuristic algorithms for combinatorial optimization. Ph.D. Dissertation, Poznan University of Technology.

Jaszkiewicz A. and Ferhat A.B. (1999). Solving multiple criteria choice problems by interactive trichotomy segmentation. *European Journal of Operational Research* 113, 271–280.

Jones D., Mirrazavi S.K. and Tamiz M. (2002). Multi-objective meta-heuristics: An overview of the current state-of-the-art. *European Journal of Operational Research* 137, 1–9.

Kim B., Gel E.S., Carlyle W.M. and Fowler J.W. (2001). A new technique to compare algorithms for bi-criteria combinatorial optimization problems. In: Köksalan M. and Zionts S. (eds.), *Multiple Criteria Decision Making in the New Millenium*, Lecture Notes in Economics and Mathematical Systems 507. Springer Verlag, 113–123.

Knowles J.D. and Corne D.W. (1999). The Pareto archived evolution strategy: A new baseline algorithm for multiobjective optimisation. Proceedings of the 1999 Congress on Evolutionary Computation, Washington, D.C., 98–105.

Köksalan M. (1999). A heuristic approach to bicriteria scheduling. *Naval Research Logistics* 46, 777–789.

Koktener E. and Köksalan M. (2000). A simulated annealing approach to bicriteria scheduling problems on a single machine. *Journal of Heuristics* 6, 311–327.

Küfer K.H., Scherrer A., Monz M., Alonso F., Trinkaus H., Bortfeld T. and Thieke C. (2003). Intensity-modulated radiotherapy – A large scale multi-criteria programming problem. *OR Spectrum* 25, 223–249.

Kursawe F. (1992). Evolution strategies for vector optimization. Proceedings of the 10th International Conference on Multiple Criteria Decision Making, Taipei-Taiwan, 187–193.

Laumanns M., Zitzler E. and Thiele L. (2001). On the effect of archiving, elitism, and density based selection in evolutionary multi-objective optimization. Proceedings of the Evolutionary Multi-Criteria Optimization. First International Conference, EMO 2001. Zürich, Switzerland, March 7–9, 2001, Lecture Notes in Computer Science 1993. Springer Verlag, 181–196.

Lawler E.L., Lenstra J.K., Rinooy Kan A.G. and Shmooys D.B. (1985). *The traveling salesman problem: A guided tour of combinatorial optimization*. Wiley Interscience Series in Pure and Applied Mathematics. John Wiley.

Lee H. and Pulat P.S. (1993). Bicriteria network flow problems: Integer case. *European Journal of Operational Research* 66, 148–157.

Liepins G.E., Hilliard M.R., Richardson J. and Palmer M. (1990). Genetic algorithms application to set covering and travelling salesman problems. In: Brown D.E. and White C.C. (eds.), *Operations Research and Artificial Intelligence: The Integration of Problem-solving Strategies*. Kluwer Academic Publishers, 29–57.

Loukil Moalla T., Teghem J. and Fortemps P. (2000a). Solving multiobjective scheduling problems with tabu search. Workshop on Production Planning and Control, 2-4 October 2000, Facultés Universitaires Catholiques de Mons, Mons, Belgium.

Loukil Moalla T., Teghem J. and Tuyttens D. (2000b). Solving multiobjective scheduling problems with the MOSA method. Workshop on Production Planning and Control, 2-4 October 2000, Facultés Universitaires Catholiques de Mons, Mons, Belgium.

Lučić P. and Teodorović D. (1999). Simulated annealing for the multi-objective aircrew rostering problem. *Transportation Research A: Policy and Practice* 33, 19–45.

Malakooti B., Wang J. and Tandler E.C. (1990). A sensor-based accelerated approach for multi-attribute machinability and tool life evaluation. *International Journal of Production Research* 28, 23–73.

Marett R. and Wright M. (1996). A comparison of neighborhood search techniques for multi-objective combinatorial problems. *Computers and Operations Research* 23, 465–483.

Mariano C.E. and Morales E. (1999a). MOAQ and ant-Q algorithm for multiple objective optimization problems. In: Banzhaf W., Daida J., Eiben A.E., Garzon M.H., Honavar V., Jakiela M. and Smith R.E. (eds.), *Proceeding of the Genetic and Evolutionary Computation Conference, Orlando, Florida, USA, 13-17 July 1999*, volume 1, 894–901.

Mariano C.E. and Morales E. (1999b). A multiple objective ant-q algorithm for the design of water distribution irrigation networks. Technical report, Instituto Mexicano de Tecnología del Agua, México.

McMullen P.R. (2001). An ant colony optimization approach to addressing a JIT sequencing problem with multiple objectives. *Artificial Intelligence in Engineering* 15, 309–317.

McMullen P.R. and Frazier G.V. (1999). Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel workstations. *International Journal of Production Research* 36, 2717–2741.

Miettinen K. (1999). *Nonlinear Multiobjective Optimization*. International Series in Operations Research and Management Science 12. Kluwer Academic Publishers.

Morita H., Gandibleux X. and Katoh N. (2001). Experimental feedback on biobjective permutation scheduling problems solved with a population heuristic. *Foundations of Computing and Decision Sciences Journal* 26, 23–50.

Murata T. and Ishibuchi H. (1995). MOGA: Multi-objective genetic algorithms. Proceedings of the 2nd IEEE International Conference on Evolutionary Computing, Perth, Australia, 289–294.

Nam D. and Park C.H. (2000). Multiobjective simulated annealing: A comparative study to evolutionary algorithms. *International Journal of Fuzzy Systems* 2, 87–97.

Osman I. and Laporte G. (1996). Metaheuristics: A bibliography. *Annals of Operations Research* 63, 513–623.

Osyczka A. (2001). *Evolutionary Algorithms for Single and Multicriteria Design Optimization*, Studies in Fuzziness and Soft Computing 79. Physica Verlag.

Pamuk S. and Köksalan M. (2001). An interactive genetic algorithm applied to the multiobjective knapsack problem. In: Köksalan M. and Zionts S. (eds.), *Multiple Criteria Decision Making in the New Millennium*, Lecture Notes in Economics and Mathematical Systems 507. Springer Verlag, 265–272.

Papadimitriou C.H. and Yannakakis M. (2000). On the approximability of trade-offs and optimal access to web sources. Proceedings of the 41st Annual Symposium on the Foundation of Computer Science FOCS00, 86–92, Redondo Beach, CA (USA).

Paquete L. and Fonseca C.M. (2001). A study of examination timetabling with multiobjective evolutionary algorithms. In: Sousa J.P. de (ed.), *MIC'2001 Proceedings 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 1, 149–153.

Parks G. and Suppapitnarm A. (1999). Multiobjective optimization of PWR reload core designs using simulated annealing. Proceedings of the International Conference on Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications. Madrid, Spain, September 1999, volume 2, 1435–1444.

Pires D., Henggeler Antunes C. and Gomes Martins A. (2001). A tabu search multiobjective approach to capacitor allocation in radial distribution systems. In: Sousa J.P. de (ed.), *MIC'2001 Proceedings of the 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 1, 169–174.

Rahoual M., Kitoun B., Mabed M., Bachelet V., and Benameur F. (2001). Multicriteria genetic algorithms for the vehicle routing problem with time windows. In: Sousa J.P. de (ed.), *MIC'2001 Proceedings of the 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 2, 527–532.

Ramos R.M., Alonso S., Sicilia J. and González C. (1998). The problem of the optimal biobjective spanning tree. *European Journal of Operational Research* 111, 617–628.

Randriamasy S., Gandibleux X., Figueira J., and Thomin Ph. (2002). Fiche brevet n^{o} 03291744.5-2416 intitulée “Dispositif et procédé de détermination de chemins de routage dans un réseau de communications, en présence d'attributs de sélection”. Fiche déposée le 15 Juillet 2002.

Reeves C. (1995). *Modern Heuristic Techniques for Combinatorial Problems.* McGrawHill.

Ribeiro R. and Lourenço H. (2001). A multi-objective model for a multi-period distribution management problem. In: Sousa J.P. de (ed.), *MIC'2001 Proceedings of the 4th Metaheuristics International Conference, Porto, July 16-20, 2001* 1, 97–101.

Rosenblatt M.J. and Sinuany-Stern Z. (1989). Generating the discrete efficient frontier to the capital budgeting problem. *Operations Research* 37, 384–394.

Ruhe G. and Fruhwirth B. (1990). $\epsilon-$optimality for bicriteria programs and its application to minimum cost flows. *Computing* 44, 21–34.

Safer H.M. (1992). Fast Approximation Schemes for Multi-Criteria Combinatorial Optimization. Ph.D. Dissertation, Sloan School of Management, MIT.

Safer H.M. and Orlin J.B. (1995a). Fast approximation schemes for multi-criteria combinatorial optimization. Working paper 3756-95, Sloan School of Management, MIT, Cambridge, MA, 1995.

Safer H.M. and Orlin J.B. (1995b). Fast approximation schemes for multi-criteria flow, knapsack, and scheduling problems. Working paper 3757-95, Sloan School of Management, MIT, Cambridge, MA, 1995.

Sawaragi Y., Nakayama H. and Tanino T. (1995). *Theory of Multiobjective Optimization*. Academic Press.

Sayin S. (2000). Measuring the quality of discrete representations of efficient sets in multiple objective mathematical programming. *Mathematical Programming* 87, 543–560.

Schaffer J.D. (1984). Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. Ph.D. Dissertation, Vanderbilt University.

Schaffer J.D. (1985). Multiple objective optimization with vector evaluated genetic algorithms. In: Grefenstette J.J. (ed.), *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*. Lawrence Erlbaum, 93–100.

Serafini P. (1986). Some considerations about computational complexity for multi objective combinatorial problems. In: Jahn J. and Krabs W. (eds.), *Recent advances and historical development of vector optimization*, Lecture Notes in Economics and Mathematical Systems, 294. Springer Verlag, 222–232.

Serafini P. (1992). Simulated annealing for multiobjective optimization problems. Proceedings of the 10th International Conference on Multiple Criteria Decision Making, Taipei-Taiwan, 87–96.

Shelokar P.S., Adhikari S., Vakil R., Jayaraman V.K. and Kulkarni B.D. (2000). Multiobjective ant algorithm: Combination of strength Pareto fitness assignment and thermodynamic clustering. *Foundations of Computing and Decision Sciences* 25, 213–230.

Shelokar P.S., Jarayaman V.K. and Kulkarni B.D. (2002). Ant algorithm for single and multiobjective reliability optimization problems. *Quality and Reliability Engineering International* 18, 497–514.

Shelokar P.S., Jayarama V.K. and Kulkarni B.D. (2003). Multiobjective optimization of reactor-regenerator system using ant algorithm. *Petroleum Science and Technology* 21, 1167–1184.

Sigal I.K. (1994). Algorithms for solving the two-criterion large-scale travelling salesman problem. *Computational Mathematics and Mathematical Physics* 34, 33–43.

Srinivas N. and Deb K. (1994). Multiobjective optimization using non-dominated sorting in genetic algorithms. *Evolutionary Computation* 2, 221–248.

Steuer R., Silverman J. and Whisman A. (1993). A combined Tchebycheff/aspiration criterion vector interactive multiobjective programming procedure. *Management Science* 39, 1255–1260.

Sun M. (1997). Applying tabu search to multiple objective combinatorial optimization problems. Proceedings of the 1997 DSI Annual Meeting, San Diego, California, volume 2, 945–947.

Sun M., Stam A. and Steuer R. (1996). Solving multiple objective programming problems using feed-forward artificial neural networks: The interactive FFANN procedure. *Management Science* 42, 835–849.

Sun M., Stam A. and Steuer R. (2000). Interactive multiple objective programming using Tchebycheff programs and artificial neural networks. *Computers and Operations Research* 27, 601–620.

Sun M. and Steuer R. (1996). Quad-trees and linear lists for identifying nondominated criterion vectors. *INFORMS Journal on Computing* 8, 367–375.

Suppapitnarm A. and Parks G. (1999). Simulated annealing: An alternative approach to true multiobjective optimization. In: Wu A.S. (ed.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99). Orlando, Florida*, 406–407.

Suppapitnarm A., Seffen K., Parks G. and Clarkson P. (2000). A simulated annealing algorithm for multiobjective optimization. *Engineering Optimization* 33, 59–85.

Sysoev V. and Dolgui A. (1999). A Pareto optimization approach for manufacturing system design. Proceedings of the International Conference on Industrial Engineering and Production Management (IEPM'99). July 1-15 1999, Glasgow, Scotland, volume 2, 116–125.

Tamaki H., Mori M., Araki M., Mishima Y. and Ogai H. (1994). Multi-criteria optimization by genetic algorithms: A case of scheduling in hot rolling process. Proceedings of the 3rd Conference of the Association of Asian-Pacific Operational Research Societies within IFORS (APORS94), 1994.

Teghem J., Tuyttens D. and Ulungu E.L. (2000). An interactive heuristic method for multi-objective combinatorial optimization. *Computers and Operations Research* 27, 621–634.

Tenfelde-Podehl D. (2002). Facilities Layout Problems: Polyhedral Structure, Multiple Objectives and Robustness. Ph.D. Dissertation, Universität Kaiserslautern.

Thompson M. (2001). Application of multi objective evolutionary algorithms to analogue filter tuning. In: Zitzler E., Deb K., Thiele L., Coello Coello C.A. and Corne D. (eds.), *First International Conference on Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science 1993. Springer Verlag, 546–559.

T'Kindt V., Monmarché N., Tercinet F. and Laügt D. (2002). An ant colony optimization algorithm to solve a 2-machine bicriteria flowshop scheduling problem. *European Journal of Operational Research* 142, 250–257.

Todd D.S. and Sen P. (1997). A multiple criteria genetic algorithm for containership loading. In: Bäck T. (ed.), *Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA97)*.

Tuyttens D., Teghem J., Fortemps P. and Van Nieuwenhuyse K. (2000). Performance of the MOSA method for the bicriteria assignment problem. *Journal of Heuristics* 6, 295–310.

Ulungu E.L. and Teghem J. (1992). Heuristic for multi-objective combinatorial optimization problems with simulated annealing. Presented at the EURO XII Conference, Helsinki.

Ulungu E.L. (1993). Optimisation combinatoire multicritère: Détermination de l'ensemble des solutions efficaces et méthodes interactives. Ph.D. Dissertation, Université de Mons-Hainaut.

Ulungu E.L. and Teghem J. (1994). The two-phases method: An efficient procedure to solve bi-objective combinatorial optimization problems. *Foundations of Computing and Decision Sciences* 20, 149–165.

Ulungu E.L., Teghem J. and Fortemps P.H. (1995). Heuristics for multi-objective combinatorial optimisation problem by simulated annealing. In: Wei Q., Gu J., Chen G. and Wang S. (eds.), *MCDM: Theory and Applications*. SCI-TECH Information Services, 228–238.

Ulungu E.L., Teghem J., Fortemps P.H. and Tuyttens D. (1999). MOSA method: A tool for solving multiobjective combinatorial optimization problems. *Journal of Multi-Criteria Decision Analysis* 8, 221–236.

Ulungu E.L., Teghem J. and Ost C. (1998). Efficiency of interactive multi-objective simulated annealing through a case study. *Journal of the Operational Research Society* 49, 1044–1050.

Viana A. and Sousa P.J. de (2000). Using metaheuristics in multiobjective ressource constrained project scheduling. *European Journal of Operational Research* 120, 359–374.

Visée M., Teghem J., Pirlot M. and Ulungu E.L. (1998). Two-phases method and branch and bound procedures to solve the bi-obective knapsack problem. *Journal of Global Optimization* 12, 139–155.

Warburton A. (1987). Aproximation of Pareto optima in multiple-objective shortest-path problems. *Operations Research* 35, 70 –79.

White D.J. (1986). Epsilon efficiency. *Journal of Optimization Theory and Applications* 49, 319–337.

Zhou G. and Gen M. (1999). Genetic algorithm approach on multi-criteria minimum spanning tree problem. *European Journal of Operational Research* 114, 141–152.

Zitzler E. (1999). Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications. Ph.D. Dissertation, Swiss Federal Institute of Technology.

Zitzler E., Deb K., Thiele L., Coello C. and Corne D. (2001). *Evolutionary Multi-Criterion Optimization*, *Lecture Notes in Computer Sciences*, 1993. Springer Verlag.

Zitzler E. and Thiele L. (1998). An evolutionary algorithm for multiobjective optimization: The strength Pareto approach. Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zürich, Switzerland.

Zitzler E. and Thiele L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation* 3, 257–271.

DISCUSSION

Rafael Caballero(1) **and Carlos Romero**(2)
(1) Universidad de Málaga, Spain
(2) Universidad Politécnica de Madrid, Spain

In the last years, there has been a Kuhnian revolution in the optimisation and computation sciences that basically consists in the so called metaheuristics approaches. Under this name several methods like Genetic Algorithms, Tabu Search, Simulated Annealing, etc, have appeared. All these methods share the common purpose of guiding a heuristic search to explore the feasible set, in order to obtain "good solutions" to complicated mathematical programming problems.

The metaheuristics approaches overcome the main difficulties associated to the multiple objective optimisation problems. In fact, these methods are underpinned by algorithms designed for linear cases (with continuous or integer/binary variables). Hence, when they are applied to complex problems (i.e., huge number of integer/binary variables, non-linear functions, logic constraints, non-convex feasible sets, etc), the determination of a precise optimum is very difficult if not impossible. However, the complexities cited above are quite common in many real-world problems. This situation has generated an important advance in the development and application of metaheuristic approaches as it is extensively shown in the paper that we are commenting. It should also be noted, that the problems formulated and solved with the help of metaheuristics come from very different disciplines such as: artificial intelligence, neural networks, data mining, classification problems, etc. The main and single disadvantage of this type of approach is that global optimality of the solution obtained cannot be guaranteed.

On the other hand, combinatorial optimisation is a relevant approach, among other things, for their potential applications to many real problems such as: capital budgeting, assignment problems, travelling salesman problem, etc. However, in many real world applications, where many decision variables and constraints are involved, finding an optimal solution to such type of problems can become an extremely difficult task. This difficulty is due to that in combinatorial optimisation, the feasible set has not a convex structure, but a lattice of points or a set characterised by disjoint line segments in the close relative mixed-integer programming problems.

Moreover, these computational difficulties achieve almost the level of impossibility, when realistically several objectives are considered within the combinatorial optimisation scenario.

For the reasons above commentated, it is not a surprise the boom of papers that have appeared in around the last ten years, addressing multiobjective combinatorial optimisation (MOCO) problems with the help of metaheuristics procedures. This boom, that is, this period of normal science, using again a Kuhnian language, required a serious work of assessment and systematisation. This task, has been superbly undertaken by Professors Ehrgott and Gandibleux in the survey that we are commenting. In fact, they have not only collected an impressive number of references in the field, but what is much more important, they have been able to put all the work into perspective, analysing with rigour the pros and cons of these methodological approaches. They have also outlined the most promising trends in this area, and they have finished with a sentence that we completely agree and provides a highlight of the future in this discipline "The new trends promise many future papers".

The papers reviewed by Ehrgott and Gandibleux focus in the description and assessment of the existing metaheuristics for the determination of the Pareto-efficient frontier; that is, they basically assess the return of each method with respect to a measurement of the computational effectiveness. Perhaps, it should also be interesting an additional effort analysing the surveyed methods, also from the perspective of the traditional multiple objective programming methods. In this way, the potential links between both families of approaches will be clarified. Thus, it seems promising to establish connections between metaheuristics approaches and several multiple objectives approaches like methods using scalarising functions, aspiration levels, weights, as well as methods that interact with the decision-maker. This orientation will increase clarity and precision in the future dialogue, among researchers and practitioners belonging to both fields of specialisation.

Regarding future trends, we dare to point out two directions still not well explored, where metaheuristics seem specially promising. One line refers to developments and refinements in the use of interactive metaheuristics methods for MOCO problems. In this direction, is advisable a recent paper by Phelps and Köksalan (2003). In this research, the interaction with the decision-maker is made up through "pairwise" comparisons, that they are used to estimate the fitness of the newly generated solutions. It seems to us, that this type of interaction as well as its implication within the heuristic procedure, might be of great interest in the near future. This is specially relevant in real applications, where the computation time of each interaction must be small in order to make the whole interaction process viable.

The other line of research refers to the use of this type of approach not only to the approximation of the efficient set, that we are aware that is a very complex task, but also in the determination of the optimum or best-compromise solution. In fact, within a MOCO context not only the feasible set has a complex structure (e.g., lack of convexity), what requires metaheuristics procedures for its computation, but also the multi-objective preferential function is specially complex (e.g., lack of concavity).

References

Phelps S.P. and Köksalan M. (2003). An Interactive Evolutionary Metaheuristic for Multiobjective Combinatorial Optimization. *Management Science* 49, 1726-1738.

Carlos A. Coello Coello

Centro de Investigación y de Estudios Avanzados
del Instituto Politécnico Nacional, Mexico

The paper by Ehrgott and Gandibleux constitutes, with no doubt, a very valuable source for those interested in the use of metaheuristics for multiobjective combinatorial optimization (MOCO). The paper is very well written and it provides a comprehensive (and well explained) coverage of the most significant works developed in the area.

My first comment is regarding the multiobjective evolutionary algorithms (MOEAs) discussed in Section 6. Although it is true that VEGA, MOGA93, NSGA and NPGA are the most remarkable MOEAs from the early days of the field, it would be nice if the authors could briefly refer to the most remarkable modern MOEAs in current use. From these, the most noticeable absence is the NSGA-II (Deb el al. (2002)) which is one of the most competitive MOEAs designed to date. Another important MOEA that is missing is SPEA2 (Zitzler et al. (2002)), which is a slightly revised version of SPEA that has become increasingly popular. It is also worth mentioning that there is a "memetic" version of PAES that has been used for MOCO (see Knowles and Corne (2000)).

An issue that was recently brought to my attention has to do with the remarkable similarities between the ant colony and GRASP. This is despite the fact that they were both developed at different time periods, with different motivations and within different research communities. It would be nice if the authors could briefly comment on this matter.

I would add to Section 10 the two following metaheuristics that have recently become popular choices for multiobjective optimization:

- **Particle Swarm Optimization:** This metaheuristic was inspired by the choreography of a bird flock and has been found to be very useful in a wide variety of optimization tasks (Kennedy and Eberhart (2001)). Despite its extreme simplicity and ease of implementation (which makes the approach very fast in continuous optimization problems), this approach has triggered a considerable amount of research related to its use in multiobjective optimization, Coello Coello and Salazar Lechuga (2002), Fieldsend (2003), Fieldsend and Singh (2002), Hui and Eberhart (2002), Hui et al. (2000), Li (2003), Mostaghim and Teich (2003), Parsopoulos and Vrahatis (2002), Ray and Liew (2002), Srinivasan and Seow (2003).

- **Artificial Immune Systems:** Also with a biological inspiration, artificial immune systems have become quite popular in the last few years due to their suitability for certain types of applications, mainly related to computer security, Dasgupta (1999), De Castro and Timmis (2002). The first attempt to extend an artificial immune system to deal with multiobjective optimization problems dates back to the work by Yoo and Hajela (1999) in which a simple linear aggregating function is used. However, more recently, several other proposals (from which only a few are based on the concept of *Pareto optimality*) have been presented at international conferences, Anchor et al.

(2002), Coello Coello and Cruz Cortés (2002), Cruz Cortés and Coello Coello (2003), Cui et al. (2001), Kuparti and Azarm (2000).

References

Anhor K.P., Zydallis J.B., Gunsh G.H. and Lamont G.B. (2002). Extending the Computer Defense Immune System: Network Intrusion Detection, with a Multiob jetive Evolutionary Programming Approah. In: Timmis J. and Bentley P.J. (eds.), *First International Conference on Artificial Immune Systems (ICARIS'2002)*, Uninniversity of Kent at Canterbury, UK, September 2002.

Coello Coello C.A. and Cruz Cortés N. (2002). An Approah to Solve Multiobjetive Optimization Problems Based on an Artificial Immune System. In: Timmis J. and Bentley P.J. (eds.), *First International Conference on Artificial Immune Systems (ICARIS'2002)*, University of Kent at Canterbury, UK, September 2002.

Coello Coello C.A. and Salazar Lechuga M. (2002). MOPSO: A Proposal for Multiple Objetive Partile Swarm Optimization. *Congress on Evolutionary Computation (CEC'2002)* 2, 1051-1056. Pisataway, New Jersey, May 2002.

Cruz Cortés N. and Coello Coello C.A. (2003). Multiobjective Optimization Using Ideas from the Clonal Seletion Principle. In: Cantu-Paz E., Foster J.A, Deb K., David L., Rajkumar R. (eds.), *Genetic and Evolutionary Computation–GECCO 2003. Proceedings*, Leture Notes in Computer Science 2723. Springer Verlag, 158-170.

Cui X, Li M. and Fang T.J. (2001). Study of Population Diversity of Multiobjective Evolutionary Algorithm Based on Immune and Entropy Principles. Proceedings of the Congress on Evolutionary Computation 2001 (CEC'2001), pages 1316-1321, Pisataway, New Jersey, May 2001. IEEE Servie Center.

Dasgupta D. (1999). *Artificial Immune Systems and Their Applications*. Springer Verlag.

De Castro L.N. and Timmis J. (2002). *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer Verlag.

Deb K., Pratap A., Agarwal S. and Meyarivan T. (2002). A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transations on Evolutionary Computation* 6, 182-197.

Fieldsend J.E. and Singh S. (2002). A Multi-Objective Algorithm based upon Particle Swarm Optimisation, an Efficient Data Struture and Turbulence. Proeedings of the U.K. Workshop on Computational Intelligence, Birmingham, UK, September 2002.

Fieldsend J.E. (2003). Novel Algorithms for Multi-Objective Search and their Application in Multi-Objective Evolutionary Neural Network Training. Ph.D. Dissertation, University of Exeter.

Hui X. and Eberhart R.C. (2002). Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization. Congress on Evolutionary Computation 1, 1677-1681, Pisataway, New Jersey, July 2000. IEEE Service Center.

Hui X., Eberhart R.C. and Shi Y. (2003). Particle Swarm with Extended Memory for Multiobjective Optimization. IEEE Swarm Intelligence Symposium Proceedings, 193-197, Indianapolis, Indiana, USA. IEEE Service Center.

Kennedy J. and Eberhart R. (2001). *Swarm Intelligence*. Morgan Kaufmann Publishers.

Knowles J. and Corne D. (2000). M-PAES: A Memetic Algorithm for Multiobjective Optimization. Congress on Evolutionary Computation 1, 325-332, Pisataway, New Jersey, USA. IEEE Service Center.

Kurpati A. and Azarm S. (2000). Immune Network Simulation with Multiobjective Genetic Algorithms for Multidisiplinary Design Optimization. *Engineering Optimization* 33, 245-260.

Li X. (2003). A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization. In: Cantu-Paz E., Foster J.A, Deb K., David L., Rajkumar R. (eds.), *Genetic and Evolutionary Computation–GECCO 2003. Proceedings*, Leture Notes in Computer Science 2723. Springer Verlag, 37-48.

Mostaghim S. and Teih J. (2003). Strategies for Finding Good Loal Guides in Multi-objective Particle Swarm Optimization (MOPSO). IEEE Swarm Intelligence Symposium Proceedings, pages 26-33, Indianapolis, Indiana, USA, April 2003. IEEE Service Center.

Parsopoulos K.E. and Vrahatis M.N. (2002). Particle Swarm Optimization Method in Multiobjective Problems. Proceedings of the 2002 ACM Symposium on Applied Computing (SAC'2002), 603-607, Madrid, Spain, 2002. ACM Press.

Ray T. and Liew K.M. (2002). A Swarm Metaphor for Multiobjective Design Optimization. *Engineering Optimization* 34, 141-153.

Srinivasan D. and Seow T.H. (2003). Partile Swarm Inspired Evolutionary Algorithm (PS-EA) for Multiobjective Optimization Problem. Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003) 3, 2292-2297, Canberra, Australia, December 2003. IEEE Press.

Yoo J. and Hajela P. (1999). Immune Network Simulations in Multicriterion Design. *Strutural Optimization* 18, 85-94.

Zitzler E., Laumanns N. and Thiele L. (2002). SPEA2: Improving the Strength

Pareto Evolutionary Algorithm. In: Giannakoglou K., Tsahalis D., Periaux J., Papailou P. and Fogarty T. (eds.), *EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*, Athens, Greece, 2002.

Thomas Hanne
Fraunhofer Institute for Industrial Mathematics, Germany

The paper by Ehrgott and Gandibleux presents a thorough treatment of metaheuristics for solving multiobjective combinatorial optimization (MOCO) problems. An introduction into the field and its basic terminology is provided. Especially helpful is the presentation of many method details (almost as extensive as usually for textbooks) and some pseudo code for various families of multiobjective metaheuristics and single instances of algorithms. Since metaheuristics usually require problem-specific adaptations, even a presentation of pseudo code does not make them immeadiately ready-to-use. For this purpose, the paper discusses a rich amount of relevant literature which provides the reader with deeper insights into the research field and to actually use the methods for his/her specific MOCO problems. Since the paper is quite comprehensive, this nontechnical discussion note aims at revisiting some more or less open issues of the wider research field.

1. What is approximation?

Usually, approximation is defined, as done in the paper, in terms of the relative closeness of the obtained solution(s) to the optimal solution(s) (also see Kall (1986)). For continuous optimization problems, the question of approximation is frequently reformulated as a question of convergence, for which a number of different concepts of formalization do exist. Since many metaheuristics can be characterized as stochastic search routines, the usual concepts of (deterministic) convergence have to be replaced by stochastic ones. Considering the multiobjective nature of an optimization problem, the idea of closeness to the optimum becomes even more complicated. See, e.g., Helbig and Pateva (1994) for various concepts of ϵ-efficiency. As in the case of scalar optimization, an analysis of convergence or stochastic convergence may, in principle, be based on a number of different concepts. For instance, in the case of multiobjective evolutionary algorithms for continuous optimization problems, a stochastic convergence analysis is provided in Hanne (1999).

With respect to MOCO, dealing with a finite set of solutions makes some things easier and others harder. With a finite set of alternatives a convergence proof should also, in principle, be possible. However, there may be metaheuristics which do not allow for convergence even if the available time is not restricted. For instance, the result that canonical genetic algorithms do not converge has stimulated quite a lot of discussion and the insight that small modifications such as an elitist rule which preserves the best solution(s) guarantees convergence (see, e.g., Rudolph (1994)).

Reaching a specific quality of the results (see Section 3 below) within a specific time is more problematic. From a theoretical point of view, we know for some NP-hard problems that it is also difficult to find an approximate solution (see, e.g., Ausiello et al. (1999), Kann and Panconesi (1996), or Wanka (2002) for introductions to a complexity analysis of approximation). For instance, assuming P≠NP, the general travelling salesperson problem (TSP) is not in APX, the language class of problems with a constant relative approximation quality (see Papadimitriou and Yannakakis (1993)). For other problems, such as the metric TSP, we may know algorithms guaranteeing a constant relative approximation quality, but we cannot get significantly better if more time is available. The relationship of time and approximation quality is considered in the classes PAS and FPAS. PAS is the language class of problems which can be approximated in polynomial time, while its subclass FPAS additionally requires that the computation time is polynomial in $1/\varepsilon$. In our case, it is possible to show that the metric TSP is not in PAS under the usual assumption P≠NP (see Crescenzi et al. (1999) for more general results on approximation classes).

In these cases we know that there will be no polynomial time algorithm (metaheuristics) which surely delivers a solution of a specific quality. The usual analysis in complexity theory is, however, some kind of worst case analysis. Average case analysis may take place based on a randomization of the algorithms, an idea which, interestingly, has been taken up by various metaheuristics such as evolutionary algorithms (see, e.g., Jansen and Wegener (2001)).

Up to now, there is, however, few research on such kind of analysis in combinatorial optimization. For multiobjective problems, the situation is even worse as one of the authors already remarks in Ehrgott (2000). Summarizing, let us remark that approximation properties in single and multiple objective optimization and, thus, the possible success of metaheuristics significantly depend on properties of the specific class of considered MOCO problems.

2. What method fits best?

Long time ago there was much discussion on general problem solvers (see, e.g., Newell and Simon (1963)) and optimism was expressed that such methods do exist and should be developed soon. Then there came lots of disillusion due to theoretical (e.g. complexity theory) and practical reasons. Those methods which are today called metaheuristics (most of them are in fact much older than this expression) are close to the original idea of general problem solvers. However, as we know, those methods which are good for a broad range of problems are usually not that good for specific problems which may better be solved by specialized or adapted methods. With NP-hard combinatorial problems, which are frequent in practice, it cannot be expected to find an efficient method (i.e. one with polynomial running time) which surely solves the problem to optimality. However, different methods do perform differently for a given type of problems or for specific problem instances.

Therefore, comparative studies of metaheuristics and other methods applicable to combinatorial problems are necessary. Equally important is research into problem-specific adaptations of general solution concepts (see, e.g., Michalewicz (1998) who emphasizes this issue for evolutionary algorithms). The art of designing and tailoring a metaheuristics for a specific optimization problem has a significant impact on the algorithm's performance. Therefore, comparative studies found in the literature are to be taken with a pinch of salt.

3. How to measure the quality?

A prerequisite for comparing algorithms and judging the quality of their results is the availability of a measure of performance. The authors focus on concepts based on the distance to an ideal solution but also discuss some other approaches. Altogether, quite many concepts for measuring quality can be found in the literature and it is in general not clear whether different measures are compatible or lead to similar results. This problem is emphasized in the work by Zitzler et al. (2002) who show that it is not possible to construct an ideal unary quality measure if there are two or more objective functions.

However, despite this theoretical analysis, practical considerations require a suitable measure. A way out of this dilemma may be to introduce preference-based information which allows some kind of scalarization. Assume that a decision maker is, for instance, likely to operate with a reference point approach (see, e.g., Wierzbicki (1986)) for selecting a compromise solution from the efficient set. Then an appropriate measure for judging the quality of an approximate Pareto set might be similar to those distance-based approaches, with the only difference of utilizing a reference point instead of the ideal point.

4. What if plenty of time is available?

In the case that plenty of time is available, is it necessary to change the method? Does the method with a significant increase in time really increase the quality of approximation? Or may there be some stagnation, some kind of "cycling" or "stalling" (such as the cycling and stalling known from the simplex algorithm (see, e.g., Gal (1994), p. 54); such a cycling is, for instance, possible in standard tabu search). Or does improvement of intermediate solutions become disproportionally slow? In that case, the metaheuristics should not beat an enumerative procedure which - with a possibly exponential running time - surely would find an optimum (or the Pareto set). But if there is some, say, "stochastic cycling" (i.e. repetitions of states are not detected and a stochastic procedure may, therefore, repeat unsuccessful search steps) then the procedure may take much more time than an enumerative procedure, even though some kind of stochastic convergence towards the optimum may be given. (With a finite search space, a proof of stochastic convergence should be possible, see, e.g., Rudolph and Agapie (2000) for some results on multiobjective EAs.)

This question is not an artificial or academic one: Think of the situation that for some kind of optimization problems fast or real-time preliminary solutions (with a possibly poor quality) are required during the day time (e.g. in production planning, scheduling, staff assignment) while over night there is sufficient time to find a (near) optimal/efficient solution. Such situations suggest a two-stage approach with 2 different methods if the metaheuristics does not perform equally competitive for both situations.

5. What to do after having found a solution?

One aspect usually neglected in the area of MOMH is the following: Let us assume that a metaheuristics has found a reasonable approximation of the efficient set within an acceptable amount of time. What happens then? Usually, a decision maker does not require a set of efficient alternatives but rather a single solution. Most approaches in multicriteria decision making (MCDM, see, e.g., Gal et al. (1999)), treat these needs by requesting additional information from the decision maker (especially preference-related information) and using it for determining a compromise solution. With the MOMH methods it is usually assumed implicitly that such a preference-based MCDM method may be applied after having found an approximation of the efficient set. Such a proceeding is, of course, possible. A closer connection of an approximation of the efficient set and a preference-eliciting method may, however, prevent unnecessary computation burdens and/or lead to better results.

For instance, if a decision maker has specific threshold values for the individual objectives, efficient solutions violating these thresholds do not need to be computed. Also, a search into directions leading away from utopia solutions or reference points (see, e.g., Wierzbicki (1986)) can be excluded already within the MOMH progress scheme. In an extreme case, a decision maker may state preference-related information which is used for an a priori definition of a scalarizing function. In that case, using a conventional, single-objective metaheuristic may be fully appropriate (that's what the authors call "false MOMH"). Most frequently, a decision maker may have problems with stating such information a priori. Instead, he or she prefers to have some interaction used for exploring the set of potential solutions while stating preference-based information leading to a (hopefully) unique solution. But also in these cases only a small part of the Pareto set may be interesting for that interaction purpose and, hence, excluding the uninteresting regions is preferable when using the MOMH.

References

Ausiello G., Crescenzi P., Gambosi G., Kann V., Marchetti-Spaccamela A. and Protasi M. (1999). *Complexity and Approximation*. Springer Verlag.

Crescenzi P., Kann V., Silvestri R. and Trevisan L. (1999). Structure in Approximation Classes. *SIAM Journal on Computing* 28, 1759-1782.

Ehrgott M. (2000). Approximation Algorithms in Combinatorial Multicriteria Optimization Problems. *International Transactions in Operational Research* 7, 5-31.

Gal T. (1994). *Postoptimal Analyses, Parametric Programming, and Related Topics* (second edition). De Gruyter.

Gal T., Stewart T.J. and Hanne T. (1999). *Multicriteria Decision Making – Advances in MCDM Models, Algorithms, Theory, and Applications*. Kluwer Academic Publishers.

Hanne T. (1999). On the Convergence of Multiobjective Evolutionary Algorithms. *European Journal of Operational Research* 117, 553-564.

Helbig S. and Pateva D. (1994). Several Concepts for ε-Efficiency. *OR Spektrum* 16, 179-186.

Jansen T. and Wegener I. (2003). On the Analysis of a Dynamic Evolutionary Algorithm. Preprint.

Kall P. (1986). Approximation to Optimization Problems: An Elementary Review. *Mathematics of Operations Research* 11, 9-18.

Kann V. and Panconesi A. (1996). Hardness of Approximation. In: Dell'Amico M., Maffioli F. and Martello S. (eds.), *Annotated Bibliographies in Combinatorial Optimization*. John Wiley, 13-30.

Michalewicz Z. (1998). *Genetic Algorithms + Data Structures = Evolution Programs* (third edition). Springer Verlag.

Newell A. and Simon H. A. (1963). GPS, a Program that Simulates Human Thought. In: Feigenbaum E.A. and Feldman J. (eds.), *Computers and Thought*. McGraw-Hill, 279 - 293.

Papadimitriou C.H. and Yannakakis M. (1993). The Traveling Salesman Problem with Distances One and Two. *Mathematics of Operations Research* 18, 1-11.

Rudolph G. (1994). Convergence Analysis of Canonical Genetic Algorithms. *IEEE Transactions on Neural Networks* 5, 96-101.

Rudolph G. and Agapie A. (2000). Convergence Properties of Some Multi-Objective Evolutionary Algorithms. In: Zalzala A., Fonseca C., Kim J.H., Smith A. and

Yao X. (eds.), *Proceedings of the 2000 Congress on Evolutionary Computation (CEC 2000)* 2. IEEE Press, 1010-1016.

Wanka R. (2002). Approximationsalgorithmen, Version 2.1, Paderborn. available at `http://www.upb.de/cs/ag-madh/vorl/Approx02/`

Wierzbicki A.P. (1986). On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spektrum* 8, 73-87.

Zitzler E., Laumanns M., Thiele L. Fonseca, C. and Grunert da Fonseca V. (2002). Why Quality Assessment of Multiobjective Optimizers is Difficult. In: Cantú-Paz E., Mathias K., Roy R., Davis D., Poli R., Balakrishnan K., Honavar V., Rudolph G., Wegener J., Bull L., Potter M.A., Schultz A.C., Miller J.F., Burke E. and Jonoska N. (eds.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002)*. Morgan Kaufmann Publishers, 666–673.

Julián Molina

Universidad de Málaga, Spain

This paper provides a wide view of a field of great interest in the literature, Multiobjective Combinatorial Optimization (MOCO). Within MOCO we can find a variety of problems presenting two main common characteristics:

- The huge number of real applications where a MOCO problem must be solved (as it is shown in this paper).
- The computational cost to solve them.

This way, when facing Combinatorial Optimization problems the estimated time to be solved can be even more than $2 * 10^8$ centuries, as it can be found in Garey and Johnson (1979). But, the main interest of Combinatorial Optimization is the amount of real applications concerning this field, and consequently there is a large demand on methods to solve MOCO problems within a reasonable computational time. Into this frame, metaheuristics have been shown as a really powerful tool able to obtain a good performance relating the ratio *quality of the solutions / time needed to solve*. As it is shown in this work, in the last two decades lots of papers related with metaheuristics for MOCO problems have been published, different special volumes have focused on MOCO, several reviews have appeared, for example Ehrgott and Gandibleux (2000), and a wide variety of algorithms have been developed.

Specially, Multiobjective Evolutionary Algorithms (MOEAs) presented an intense activity in the nineties. This way, modern MOEAs are shown in this paper: MOGA (Fonseca and Fleming (1993)), NSGA (Srinivas and Deb (1994)), NPGA (Horn et al. (1994)), PAES (Knowles and Corne (1999)) and SPEA (Ziztler and Thiele (1998)).

But after this great activity in the nineties, most of these algorithms have been updated according to recent developments and discussions, offering new versions of most of them, conforming what is called as *second generation* algorithms. This way we have now the NSGA-II (Deb et al. (2000)), that is using elitism and not using niches. Erickson et al. (2001) proposed the NPGA-2, with a Pareto ranking and a different fitness sharing scheme: *continuously updated fitness sharing*. Ziztler et al. (2001) proposed the SPEA-2, where, among other differences, their are not using clustering for the archive.

Also some recent algorithms could have been included according with their good performance with some test problems: PESA, Corne et al. (2001) or the Micro-Genetic Algorithm in Coello and Toscano (2000).

Relating the quality measures, there is a good paper by Ziztler et al. (2002) showing how difficult it is to measure the quality of an approximation of a Pareto frontier, some drawbacks of the most used measures in the literature and some new ideas.

One main point also that should be taken into account when analysing metaheuristics is the number and type of parameters. In fact, from my point of view, it is still one disadvantage of this type of methods, because tuning these parameters is not an easy task for a person not involved in this topic. This way, if we want metaheuristics to be widely used by the operational researchers, we should point out two more aspects of a metaheuristic when evaluating its performance:

- Number of parameters to be tuned for getting a good performance.
- The understanding of these parameters.

For example, in this paper the parameters of VEGA are shown:

pop	the population size.
N_{gen}	the limits of generations.
parameters	the crossover probability and mutation rate.

Then, from the point of view of a person not involved in evolutionary algorithms it can be easy to understand the main scheme with the first two parameters, this is:

pop – The more you increase this parameter the more precision you should get.
–As you increase this parameter the method becomes more and more slow.

N_{gen} –Same scheme

But, what happens in general with the crossover operator and the mutation rate? It is not easy for a person not involved in evolutionary algorithms to understand how changes in these parameters will have an influence in the performance of the algorithm or how to fix values to get good solutions for a given problem. Then, from my point a view, this algorithm will be "better" without these last two parameters or if the algorithm is able to do self-tuning for them.

And this could be also a main point in the discussion "Evolutionary Algorithms versus Neighborhood Search Algorithms", because, generally, maybe it's easier to understand, for a not involved researcher, the parameters of the Neighborhood Search algorithms (number of iterations, tabu tenure, temperature, etc) than the parameters of the Evolutionary algorithms (crossover rate, mutation rate, fitness sharing rate, etc). Then, when evaluating or analysing metaheuristics it could be important to pay more attention to the number of parameters, their influence in the performance and the effort needed to implement a good tuning, because also this could be a main point for the success of this type of tools into the Operational Research community.

References

Coello C. and Toscano G. (2000). A Micro-Genetic Algorithm for Multiobjective Optimization, Lania-RI-2000-06, Laboratorio Nacional de Informática Avanzada, Xalapa, Veracruz, México.

Corne D., Jerram N., Knowles J. and Oates J. (2001). PESA-II: Region-based Selection in Evolutionary Multiobjective Optimization. In: Spector L., Goodman E., Wu A., Langdon W.B., Voigt H.-M., Gen M., Sen S., Dorigo M., Pezeshk S., Garzon M.H., and Burke E. (eds.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001).* Morgan Kaufmann Publishers, 283–290.

Deb K., Agrawal S., Pratab A., and Meyarivan T. (2000). A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. KanGal report 200001, Indian Institute of Technology, Kanpur, India.

Ehrgott M. and Gandibleux X. (2000). A Survey and Annotated Bibliography on Multiobjective Combinatorial Optimization. *OR Spektrum* 22, 425-460.

Erickson M., Mayer A. and Horn J. (2001). The Niched Pareto Genetic Algorithm 2 Applied to the Design of Groundwater Remediation Systems. In: Zitzler E., Deb K., Thiele L., Coello Coello C.A. and Corne D. (eds.), *First International Conference on Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science 1993. Springer Verlag, 681-695.

Fonseca C.M. and Fleming P.J. (1993). Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In: Forrest S. (ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms, San Mateo, California, 1993. University of Illinois at Urbana-Champaign.* Morgan Kaufman, 416–423.

Garey M. and Johnson D. (1979). *Computers and Intractability.* Freeman.

Horn J., Nafpliotis N. and Goldberg D.E. (1994). A Niched Pareto Genetic Algorithm for Multiobjective Optimization. Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, Orlando, USA, 82–87.

Knowles J.D. and Corne D.W. (1999). The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimisation. Proceedings of the 1999 Congress on Evolutionary Computation, Washington, D.C., 98–105.

Srinivas N. and Deb K. (1994). Multiobjective Optimization Using Non-Dominated Sorting in Genetic Algorithms. *Evolutionary Computation* 2, 221–248.

Zitzler E., Laumanns M. and 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.

Zitzler E., Laumanns M., Thiele L. Fonseca, C. and Grunert da Fonseca V. (2002). Why Quality Assessment of Multiobjective Optimizers is Difficult. In: Cantú-Paz E., Mathias K., Roy R., Davis D., Poli R., Balakrishnan K., Honavar

V., Rudolph G., Wegener J., Bull L., Potter M.A., Schultz A.C., Miller J.F., Burke E. and Jonoska N. (eds.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002)*. Morgan Kaufmann Publishers, 666–673.

Zitzler E. and Thiele L. (1998). An Evolutionary Algorithm for Multiobjective Optimization: The strength Pareto Approach. Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zürich, Switzerland.

Justo Puerto
Universidad de Sevilla, Spain

The paper deals with some of the nowadays more important paradigms within Operations Research and Applied Mathematics: multiobjective programming, combinatorial problems and complexity analysis of solution algorithms. Each one of these paradigms would have deserved an isolated discussion paper; needless to say that the joint consideration of the three makes this paper a valuable piece of research that would be well-regarded for the people working in the three areas.

The authors make a outstanding presentation of different approximative algorithms for MultiObjective Combinatorial Optimization (MOCO) problems. Special attention requires the multiobjective extensions of the greedy, evolutionary, neighborhood search, simulated annealing, tabu search and ant colonies techniques. The authors review some of the most significant techniques and papers in each area. Moreover, they do not simply collect the references, but they have put everything into perspective showing the insights of each technique. It is also remarkable the effort made by the authors to include pseudocodes of each technique which links the theoretical concepts to actual computer implementations (absolutely needed in this field).

MOCO problems are usually NP-hard as well as ♯P-hard. It is well-known that even those combinatorial problems that are polynomial in the scalar case (Euler tours, matching, minimum spanning tree problem,...) become NP-hard in the multiobjective counterpart. Therefore, the development of good heuristics solving this family of problems is a challenging and promising avenue of research. In the last 15 years there has been an important advance in this regard. However, the importance of MOCO problems does not simply rely in their mathematical complexity but also in the large number of real-world applications that can only be modelled in this way (see e.g. Ehrgott and Gandibleux (2000)). An important aspect in this research is to provide each heuristic with a measure of the goodness of the approximation found. This is a very difficult task but still important. A recent paper that illustrates the difficulty of finding such a general measure with regards to efficient sets is the one by Ziztler et al. (2002).

Apart from the good and extensive treatment followed by the authors in the paper, there are some minor details that could have been brought to the discussion. One is the application of exact methods to approximate solution sets of MOCO problems. This can be done by truncating the application of the exact method at a given point and taking the incumbent solution set as the approximation of the exact one. This approach can be applied with any search technique as Branch and Bound or Dynamic Programming. This type of approach reinforces the importance of finding general techniques to generate bound sets which simplify any search strategy for these problems. In addition, there are some challenging problems that may have been included in the trends. Among them I would like to mention multiobjective bilevel programming, dynamic MOCO problems and competitive MOCO problems.

In summary, the paper constitutes an excellent overview of approximate solution methods for MOCO problems that will be an obligate reference for any author doing research in this field.

References

Ehrgott M. and Gandibleux X. (2000). A Survey and Annotated Bibliography of Multiobjective Combinatorial Optimization. *OR Spektrum* 22, 425–460.

Zitzler E., Lautmanns M., Thiele L., Fonseca C.M. and Grunet da Fonseca V. (2002). Why Quality Assessment of Multiobjective Optimizers is Difficult. In: Cantú-Paz E., Mathias K., Roy R., Davis D., Poli R., Balakrishnan K., Honavar V., Rudolph G., Wegener J., Bull L., Potter M.A., Schultz A.C., Miller J.F., Burke E. and Jonoska N. (eds.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002).* Morgan Kaufmann Publishers, 666–673.

Christian Stummer
University of Vienna, Austria

Matthias Ehrgott and Xavier Gandibleux have provided an insightful survey of relevant issues in the area of heuristics and meta-heuristics for multiobjective combinatorial optimization (MOCO). I highly appreciate their effort to compile this snapshot of the current state-of-the-art in a prospering field that deals with problems that not only are of theoretical interest but are of high practical relevance as well. The latter is due to an ongoing paradigm shift: Traditionally, decision-makers have been forced to provide extensive a priori preference information that then was used to explicitly formulate an objective function and, thus, to transform the underlying, naturally multiobjective problems to (pseudo-) single-objective ones; goal programming may serve as a prime example for such an approach. Nowadays, however, an increasing number of managers demand interactive decision support that allows them to gradually specify their preferences while analyzing and exploring the solution space and, thus, to participate in and to control the decision process. Accordingly, there is a need to generate (Pareto-) efficient solutions. As this is an NP-hard task for MOCO problems, (meta-) heuristics that usually provide a favorable compromise between the quality of the approximated solution space and the required computational effort come into play.

In their paper Matthias and Xavier lead the readers from the very definition of both MOCO and approximation methods to discovering the various families of (meta-) heuristic approaches that have been proposed in the past decades. By this means, they deliver theoretical insight as well as summarize practical technicalities that should be of value for practitioners and scientists newly interested in this area. In particular, a commendable effort is done in detailing the four major “waves” of multiobjective metaheuristics that have been seen so far while their description also includes a lot of material and references needed to put these algorithms to work. Further, the authors discuss issues related to the quality of the proposed solution space, such as bounds or performance ratios, that may be of concern even for the more experienced readers who have already implemented such multiobjective (meta-) heuristics and now are going to evaluate their performances. And finally, recent algorithmic trends and (supporting) techniques related to the computer implementation are mentioned as well.

While the paper provides a fairly comprehensive overview of relevant aspects for the implementation and application of approximate solution methods for MOCO, I'd like to add two comments:

The first one concerns an issue that may appear as purely "philosophic" matter at first glance, but actually has practical impact: My concern is whether bi-objective approaches should be mixed up with "true" multiobjective approaches or whether they form a separate class of approaches. Steuer and Na (2003), for example, point out that by multiple they mean three or more objectives because difficulty primarily encounters by transiting from two to more than two objectives as the "efficient frontier" is no longer a frontier but becomes a surface (which avoids the application of available parametric solution techniques). I agree with that. Moreover, for the task of investment planning, which constitutes the field of MOCO applications I'm most familiar with, my practical experience shows that many decision-makers will not be satisfied even with three or four objectives but prefer a much more general treatment of objective values over time; e.g., instead of getting just the net present value for the cash flow they prefer to receive information about the annual cash flows for all financial years up to the planning horizon as separate objectives (for a discussion in the context of research and development project selection cf. Ringuest and Graves (1990), or Stummer and Heidenberger (2003)). As a consequence, MOCO applications may have to deal with up to ten or even more objectives. In general, such an increase in the number of objectives results in an increase in the number of efficient decision alternatives and, among other effects, in a considerably higher effort for the administration of the proposed efficient solutions during the computational runs. As the latter affects the performances of the presented approaches to a considerably different extent, I feel that it should be taken into consideration more thoroughly.

My second comment is partly linked to the first one and basically consists of reporting two preliminary results of an ongoing research project that aims at comparing multiobjective versions of Ant Colony Optimization (ACO) and Tabu Search (TS) in the context of portfolio selection; note that Doerner et al. (2004) have shown that ACO outperforms Simulated Annealing and Genetic Algorithms for this class of problems. Firstly, it turned out that ACO outperforms TS particularly for problem instances with a high number of objectives involved. An in-depth analysis reveals that ACO generates far less (but more carefully constructed) solution proposals during its execution and, thus, requires less computing time for checking whether or not they are efficient: for a typical problem instance (with six objectives considered) ACO had to use just 0.68% of the overall runtime while the TS algorithm needed 27.35%. Moreover, there have been several experiments (with ten objectives considered) for which the TS algorithm spent even more than 90% of its calculation time just for solution administration purposes and, consequently, ACO then clearly outperformed TS while this has not been the case in the first-mentioned example. That should serve as an indicator that the number of objectives in fact matters. The second finding highlights that ACO seems to behave more parameter-insensitive than TS. This is of practical relevance in applications for which only limited a priori information about a problem's characteristics is available. However, when reading through studies in the field of MOCO meta-heuristics I receive the impression that most authors have already performed exhaustive parameter tests on the available problem instances before comparing their meta-heuristics and, thus, there is only minor concern with the investigation of the methods' robustness with respect to modified parameter settings or volatile problem characteristics. Since I anticipate an increasing demand for MOCO meta-heuristics for practical applications in the forthcoming years (where one cannot expect an expert always being present to perform appropriate parameter tests), I feel that much more effort should be brought forward to address this property of MOCO meta-heuristics as well.

To conclude this discussion, I would like to stress that there are plenty of opportunities for further research on (meta-) heuristics for MOCO problems. Promising areas include (i) the evaluation of existing approaches with respect to their performances when numerous objectives have to be taken into account as well as with respect to their robustness, (ii) tools that are aimed towards speeding-up the administration of proposed efficient solutions (such as the quad-tree data structure does), (iii) the consideration of stochastic and/or dynamic aspects (for a corresponding single-objective meta-heuristic approach cf. Gutjahr (2004)), and (iv) the development of hybrid algorithms since the concentration on a sole family of heuristics could be rather restrictive. Out of the trends listed by the authors I particularly agree with their expectations in the potentialities of path-relinking.

References

Doerner K., Gutjahr W.J., Hartl R.F., Strauss C. and Stummer C. (2004) Pareto Ant Colony Optimization: A Metaheuristic Approach to Multiobjective Portfolio Selection. *Annals of Operations Research* (to appear).

Gutjahr W.J. (2004) S-ACO: An Ant-Based Approach to Combinatorial Optimization under Uncertainty. Technical Report 2004-04, Dept. of Statistics and Decision Support Systems, University of Vienna.

Ringuest J.L. and Graves S.B. (1990) The Linear R&D Project Selection Problem: An alternative to Net Present Value. *IEEE Transactions on Engineering Management* 37, 143-146.

Steuer R.E. and Na P. (2003). Multiple Criteria Decision Making Combined with Finance: A Categorized Bibliography. *European Journal of Operational Research*, 150, 496-515.

Stummer C. and Heidenberger K. (2003) Interactive R&D Portfolio Analysis with Project Interdependencies and Time Profiles of Multiple Objectives. *IEEE Transactions on Engineering Management* 50, 175-183.

Rejoinder by Matthias Ehrgott and Xavier Gandibleux

The discussants raise a number of important points that we would like to comment on in this rejoinder. Although the focus of each discussant is different, there are a number of recurring themes, so we respond to those collectively, rather than to the individual contributions. References listed below refer to our paper as [EG: xy] or to references in one of the discussions as [Discussant: xy].

1. New developments in evolutionary algorithms. Pioneer MOEA methods, but also the recently proposed methods are continuously improved. Here, we can mention NSGAII, SPEA2, NPGA2, PESA2, μGA2, NPGA2 among others. These have been described in the book of Coello et al. [EG: Coello et al. (2002)]:

   NSGA-II, [Coello: Deb at al. (2002), Molina: Deb at al. (2000)], is a new version of NSGA which is more efficient (computationally speaking), uses elitism and a crowded comparison operator that keeps diversity without specifying any additional parameters.

SPEA2 [Coello: Zitzler et al. (2002), Molina: Zitzler et al. (2001)] has three main differences with SPEA: (1) it incorporates a fine-grained fitness assignment strategy which takes into account for each individual the number of individuals that dominate it and the number of individuals by which it is dominated; (2) it uses a nearest neighbour density estimation technique which guides the search more efficiently, and (3) it has an enhanced archive truncation method that guarantees the preservation of boundary solutions

PESA (Pareto Envelope-based Selection Algorithm) has been proposed by Corne et al. in 2000. This algorithm uses a hyper-grid division of phenotype space to maintain diversity. In PESA II [Molina: Corne et al. (2001)], the notion of region-based selection is used. In region-based selection, the unit selection is a hyperbox rather than an individual.

μ**GA2** by Toscano and Coello [Molina: Coello and Toscano (2000)], is a revised version of μGA algorithm which does not require any parameter fine-tuning and proposes a dynamic selection scheme through which the algorithm decides which is the "best" crossover operator to be used at any given time.

Also, we are continuously noticing new ideas, as recently the **PICPA** algorithm where constraint programming techniques and population of individuals are combined, Barichard and Hao (2003).

We tried to check if these algorithms have been applied on MOCO problems, but according to our knowledge, the reply today is 'no'. However, if the evolutionary algorithms community and the combinatorial optimization community continue to bring together ideas around multiple objective programming problems, it will not be surprising to observe more and more applications of MOEA for solving MOCO problems in years to come. A library of numerical instances such as the MCDMlib (`www.terry.uga.edu/mcdm/`) offers a useful support for this link.

2. Measurement of quality. This is indeed an area of outstanding importance. The number of proposed measures of quality indicates how difficult it is to really judge the quality of a (meta)heuristic algorithm in the multiobjective context. The paper by Zitzler et al. [Coello, Hanne, Molina, Puerto: Zitzler et al. (2002)] formalizes this. But then, in a way it is not surprising that summarizing the quality of approxima-

tion in one number is impossible in multiobjective optimization. In fact, Tenfelde-Podehl in her PhD thesis [EG: Tenfelde-Podehl (2002)] proposes using some of the measures she suggested in common, to compare the quality of approximations based on multiple criteria.

3. Several discussants point out the need for interactive MOMHs in real applications. This seems to be a most attractive area for future research, and very little has been done so far. We are thankful for the reference provided. Two ways can be seen: First, a "direct approach" that constructs an approximation of the efficient frontier concurrently with a search for a best compromise, probably focusing the search in certain areas of the frontier; or a "sequential approach", where a compromise solution is sought among the final approximation. The latter might be attractive, as a whole arsenal of MCDA methods can be applied for the selection of a best compromise from among a finite set of alternatives (the approximation).

4. Combination of optimization with heuristics. We see that as probably the most interesting direction of research in MOCO. Exact methods can be used to solve subproblems within the metaheuristic, as well as heuristics within an exact method (to find bound sets, for example). We have pointed that out earlier (calling such methods "semi-exact" [EG: Ehrgott and Gandibleux (2000)]), and research in that regard is under way.

5. Two versus many objectives. One might indeed consider "multiple" objectives to imply more than two. And the real challenges of the field do start with three criteria (when even the computation of the supported efficient solutions becomes a challenge). But a thorough understanding of biobjective problems can certainly only be beneficial to make progress in general MOCO. And we are probably just seeing the beginning of research in that area, see also Section 7 in [EG: Ehrgott and Gandibleux (2000)].

6. Tuning of metaheuristics. As pointed out by some discussants consideration must be given to the tuning of metaheuristics. Two main reasons are apparent. First, in order to be practically useful to decision makers, the input required from DM's , especially as concerns technical parameters of the method, needs to be kept to a minimum (we have experienced this in a study on airline crew scheduling). This

has to be balanced against the apparent sensitivity of some methods towards tuning of parameters (Why is a tabu list length of 8 fine, whereas length 5 gives bad results?). Therefore, algorithms that are robust with respect to tuning will be important for practical usefulness. The second reason is for comparison of different heuristics: One may ask the provocative question if authors always invest as much effort into tuning the algorithms they compare their new invention to, as they do in tuning theirs.

Few dynamic tuning strategies exist, like "reactive GRASP" proposed for multiobjective optimisation. As mentioned before μGA2 is one example investigating this issue.

7. Other techniques. The very recent developments of Particle Swarm Optimization and Artificial Immune Systems have not been mentioned in our paper. In fact we are not aware of any studies involving MOCO so far. However, we would like to add references to the use of Scatter Search for an extension of the generalized assignment problem with multiple objectives (Lourenço et al. (2000)), and for the biobjective knapsack problem (Figueira et al. (2004)).

As the discussion of our paper shows, we are far away from making a statement as in the urban legend of the patent officer who allegedly resigned because "everything that can be invented has been invented".

Finally, we would like to thank all discussants for their comments, and Thomas Hanne, Justo Puerto, and Christian Stummer, who - apart from discussing the paper - pointed out typos and some inconsistencies in notation of the paper.

References

Barichard V. and Hao J.K. (2003). A Population and Interval Constraint Propagation Algorithm. *Lecture Notes in Computer Science* 2632, 88-101.

Figueira J., Silva C.G. de and Climaco J. (2004). Integrating Partial Optimization with Scatter Search for Solving Bi-Criteria 0-1-Knapsack Problems. Proceedings of MOPGP04 - Multiobjective Programming and Goal Programming, Hammamet, Tunisia, CD-Rom edition.

Lourenço H.R., Martí R. and Laguna M. (2000). Assigning Proctors to Exams with Scatter Search. In: Laguna M. and González J.S. (eds.), *Computing Tools for*

Modeling, Optimization and Simulation. Kluwer Academic Publishers, 215-228

Top

Volume 12, Number 1
June 2004

CONTENTS