

 Open access • Journal Article • DOI:10.1007/S10479-007-0230-0

On local optima in multiobjective combinatorial optimization problems

— [Source link](#) 

Luís Paquete, Tommaso Schiavinotto, Thomas Stützle

Institutions: University of the Algarve, Scuola Normale Superiore di Pisa, Université libre de Bruxelles

Published on: 04 Aug 2007 - Annals of Operations Research (Springer US)

Topics: Metaheuristic, Combinatorial optimization, Optimization problem, Combinatorial search and Tabu search

Related papers:

- [Performance assessment of multiobjective optimizers: an analysis and review](#)
- [A fast and elitist multiobjective genetic algorithm: NSGA-II](#)
- [Pareto Local Optimum Sets in the Biobjective Traveling Salesman Problem: An Experimental Study](#)
- [Two-phase Pareto local search for the biobjective traveling salesman problem](#)
- [Multicriteria Optimization](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/on-local-optima-in-multiobjective-combinatorial-optimization-2yni4yumtd>

On Local Optima in Multiobjective Combinatorial Optimization Problems

Luis Paquete¹, Tommaso Schiavinotto² and Thomas Stützle³

¹ Universidade do Algarve, Faculdade de Economia, Centro de Sistemas Inteligentes
Campus de Gambelas, 8005 Faro, Portugal

² Signum - Scuola Normale Superiore di Pisa
Piazza dei Cavalieri, 7, 56126 Pisa, Italy

³ Université Libre de Bruxelles, CoDE, IRIDIA,
CP 194/6, Av. F. Roosevelt 50, 1050 Brussels

Abstract. In this article, local optimality in multiobjective combinatorial optimization is used as a baseline for the design and analysis of two iterative improvement algorithms. Both algorithms search in a neighborhood that is defined on a collection of sets of feasible solutions and their acceptance criterion is based on outperformance relations. Proofs of the soundness and completeness of these algorithms are given.

1. Introduction

Stochastic local search (SLS) algorithms have been shown to provide high quality solutions for many hard combinatorial problems (Aarts and Lenstra, 1997; Hoos and Stützle, 2004). A central aspect in the development of these methods is the notion of local optimum, that is, a solution that cannot be improved with respect to its neighborhood. In fact, many general-purpose SLS methods (also called metaheuristics) have been designed with the explicit goal to overcome the problems associated with local optimality.

SLS algorithms have also been applied with considerable success to multiobjective combinatorial optimization (MOCO) problems (Ehrgott and Gandibleux, 2004). However, different from the single objective counterpart, there is not a widely accepted notion of local optimality in MOCO. This may be a reason why the existing SLS approaches to MOCO problems appear to have been developed often in a more ad-hoc and less directed manner.

The notion of local optimum for the single objective case can be extended to MOCO problems by considering sets of solutions. We call the resulting set of local optima a Pareto local optimum set. We first relate the concept Pareto local optimum set to aspects of global and local optima for the single objective case. A negative result is that SLS algorithms for single objective problems are not appropriate for obtaining a Pareto local optimum set. In a next step, we propose two iterative improvement algorithms that search according to a neighborhood that is defined on a collection of sets of solutions, and whose acceptance criterion is based on the *better* relation defined in (Hansen and Jaszkiwicz, 1998; Zitzler et al., 2003).

The first iterative improvement algorithm has, however, the disadvantage of visiting an exponential number of *sets* of solutions at each iteration of the iterative improvement algorithm. A more restricted version, called *Pareto*



local search (PLS), initially proposed and experimentally tested in (Paquete et al., 2002; Paquete et al., 2004), is shown to search for a more *interesting* and restricted set of solutions. We prove that this algorithm terminates in a specific type of a Pareto local optimum set. In addition, since there can be an exponential number of solutions to be visited, alternatives that restrict the acceptance criterion by using measures for the spread of solutions are discussed. However, we show that some caution must be taken on the type of restrictions to be used since for some choices iterative improvement algorithms can cycle.

The article is organized as follows. In Section 2, we introduce notation and discuss the definitions of local optimality we use in this article. Section 3 discusses the relation of these notions to the solutions returned by solving each objective separately. In Section 4, the iterative improvement algorithms are proposed, and Section 5 analyzes PLS with respect to its soundness and completeness. Further extensions and future work are discussed in Section 6.

2. Notation and Definitions

2.1. MULTIOBJECTIVE OPTIMIZATION

The following orders in the Euclidean space \mathbb{R}^K are considered. Let u and v be vectors in \mathbb{R}^K . The *strict component-wise* order is denoted by $u < v$, where $u_i < v_i$, $i = 1, \dots, n$; *component-wise* order by $u \leq v$, where $u \neq v$ and $u_i \leq v_i$, $i = 1, \dots, n$; and *weak component-wise* order by $u \leq v$, where $u_i \leq v_i$, $i = 1, \dots, n$. Note that if we have that $u \leq v$ and $v \leq u$, then $u = v$.

We consider here the general multiobjective program

$$\min \begin{cases} q_1 = F_1(s) \\ \vdots \\ q_K = F_K(s), \end{cases} \quad (1)$$

where s belongs to the set S of feasible solutions and \min is understood in terms of Pareto optimality. $F(s) = (F_1(s), \dots, F_K(s))$ is the *objective function vector* of s and K is the number of objectives. Without loss of generality, only minimization is considered. The objective function of a single objective optimization problem is denoted by $f(s)$. We restrict ourselves to combinatorial optimization problems.

In the context of optimization, the relation between objective function vectors of two feasible solutions s and s' is denoted as follows: if $F(s) < F(s')$, then $F(s)$ *strictly dominates* $F(s')$; if $F(s) \leq F(s')$, then $F(s)$ *dominates* $F(s')$; if $F(s) \leq F(s')$, then $F(s)$ *weakly dominates* $F(s')$. Finally, $F(s)$ and $F(s')$ are (pairwise) *nondominated* if $F(s) \not\leq F(s')$ and $F(s') \not\leq F(s)$, and are (pairwise) *non weakly dominated* if $F(s) \not\leq F(s')$ and $F(s') \not\leq F(s)$. Note that the latter is more restrictive since it implies that $F(s) \neq F(s')$. The same notation is applied when referring to solutions, if the relation holds between their objective function vectors.

A solution to Program (1) is one that is globally nondominated, that is, it is not dominated by any other feasible solution. Such a solution is known in

the literature as *Pareto global optimum*. However, there is usually not only one Pareto global optimum but many, and the solution to Program (1) becomes a set of Pareto global optima. A set is a *Pareto global optimum set* if, and only if, it contains only and all Pareto global optima. Its image in the objective space is called *efficient set*.

Note that several Pareto global optima may have the same objective function vector. Hence, rather than finding the Pareto global optimum set, one may be only interested in one solution for each element of the efficient set. In that case, the *strict* version of Pareto global optimum set is more appropriate. A set is a *strict Pareto global optimum set*, if, and only if, it contains only non weakly dominated Pareto global optima and its image in the objective space coincides with the efficient set. A strict Pareto global optimum set is called *minimal complete set* by Hansen (1979), *complete set of alternatives* by Emelichev and Perepelista (1992) and *strictly Pareto optimal set* by Ehrgott (2000).

2.2. LOCAL OPTIMALITY

An *iterative improvement* algorithm starts from an arbitrary feasible solution and searches neighbors for better solutions to replace the current one. This neighborhood search is repeated until no improvement is found anymore and the algorithm stops in a local optimum. More formally, let $N : S \mapsto 2^S$ be a *neighborhood* function that associates a set of feasible solutions $N(s)$ to every feasible solution s . A feasible solution $s' \in N(s)$ is also called a *neighbor* of s . The solution s is a *local optimum* with respect to N if, and only if, there is no $s' \in N(s)$ such that $f(s') < f(s)$. Note that different iterative improvement algorithms can be defined in dependence of the criterion used for searching an improving neighbor.

The notion of local optimum can be easily extended to MOCO in terms of Pareto optimality by adapting the notion of Pareto global optimum. In (Paquete et al., 2004), a feasible solution s is called a *Pareto local optimum* with respect to a neighborhood N , if, and only if, there is no $s' \in N(s)$ such that $F(s') \leq F(s)$. For the same reasons presented above for defining a Pareto global optimum set, Paquete et al. (2004) defined a set $S' \subseteq S$ to be a *Pareto local optimum set* with respect to a neighborhood N if, and only if, all solutions in S' are Pareto local optima with respect to N . (An analogous definition of Pareto local optimum set is given in (Deb, 1999).) Note, however, that such a set may contain dominated solutions and therefore, if nothing else is said, we require the Pareto local optimum set to have only solutions that are pairwise nondominated. Finally, an extension of this definition to non weakly dominated solutions was introduced in (Paquete and Stützle, 2006). Fig. 1 illustrates the image of a Pareto local optimum set with non weakly dominated solutions in the objective space (two minimizing objectives) by black points, where neighbors are connected by edges. It can be seen that no neighbor (represented by gray points) dominates any of the black points and that the black points are pairwise non weakly dominated.

Note that a Pareto global optimum is also a Pareto local optimum, independent of the neighborhood N . A Pareto global optimum set is a Pareto local

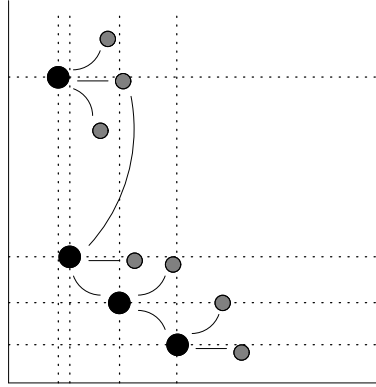


Figure 1. The set of black points forms a non weakly dominated Pareto local optimum set. The gray points are neighbors to each element of that set.

optimum set with nondominated solutions and a strict Pareto global optimum set is a Pareto local optimum set with non weakly dominated solutions. In this sense, these notions are analogous to the relationship of the notions of local optimum and global optimum in the single objective case. Putting this analogy further to the level of algorithms, it is natural to require an iterative improvement algorithm for MOCO problems to terminate once a Pareto local optimum set with only non weakly dominated solutions is found.

Remark: In the remainder of this paper, we will, for conciseness, refer to a Pareto local optimum set with only non weakly dominated solutions simply as a *Pareto local optimum set*.

3. Local Optima and Pareto Local Optima

Given the analogies between local optima for the single- and multiobjective case, a first question is whether applying iterative improvement algorithms to each objective of a MOCO problem separately can be used for obtaining a Pareto local optimum set. In general, the answer is no and we will illustrate these results using the MULTIOBJECTIVE TRAVELING SALESMAN PROBLEM (MOTSP). In the MOTSP are given n cities and distance vectors in \mathbb{N}^K for each pair of cities. The goal is to find a set of Hamiltonian tours that is minimal with respect to the notion of Pareto optimality. This problem is known to be NP-hard since its single objective version is NP-hard (Serafini, 1986).

The first negative result states that there may be cases in which a Pareto global optimum is not a local optimum for any single objective of a MOCO problem. This is shown by the following example.

EXAMPLE 3.1. Consider the MOTSP instance with 5 cities that is taken from (Emelichev and Perepelitsa, 1991).¹ In Figure 2 is given a biobjective MOTSP

¹ Emelichev and Perepelitsa (1991) proposed a generalization of this instance as an example of an MOTSP instance that has *unsupported* Pareto global optima.

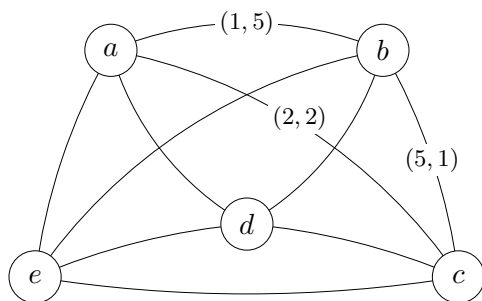


Figure 2. Graph of the MOTSP instance for Example 3.1. All the edge weight vectors are $(5, 5)$ except for the ones labeled explicitly.

instance (the edges without label have weight vectors $(5, 5)$). The Pareto global optimum set (S_g) and the corresponding image in the objective space (O_g) are presented in the following table:

solution	S_g	O_g
s_1	(a, b, d, e, c)	$(18, 22)$
s_2	(a, b, e, d, c)	$(18, 22)$
s_3	(a, b, c, e, d)	$(21, 21)$
s_4	(a, b, c, d, e)	$(21, 21)$
s_5	(a, c, b, d, e)	$(22, 18)$
s_6	(a, c, b, e, d)	$(22, 18)$

Consider the 2-exchange neighborhood, where for a given solution s , all neighbors can be obtained by replacing two edges from s with two different edges not present in s . For this neighborhood, any solution with objective function vector $(21, 21)$ is not a local optimum for any component of the objective function vector since there is a better neighbor for each objective. For example, the solution s_4 has two better neighbors: solution s_2 with respect to the first objective (by removing the edges (b, c) and (e, a) and adding the edges (b, e) and (c, a) in s_4) and s_5 with respect to the second objective (by removing the edges (a, b) and (c, d) and adding the edges (a, c) and (b, d) in s_4).

The following example shows that a locally optimal solution with respect to at least one objective of the MOCO problem may not be a Pareto local optimum.

EXAMPLE 3.2. Figure 3a gives a biobjective MOTSP instance (the edges that are not present have each distance vector component larger than 2). The Pareto global optimum set is $\{(a, g, e, b, d, c, f)\}$, the tour indicated by bold edges, and the efficient set is $\{(7, 8)\}$. Under the same 2-exchange neighborhood from the previous example, the sequence $s = (a, b, c, d, e, f, g)$, which is indicated in Figure 3b with straight lines, has an objective value vector $(8, 12)$ and it is a locally

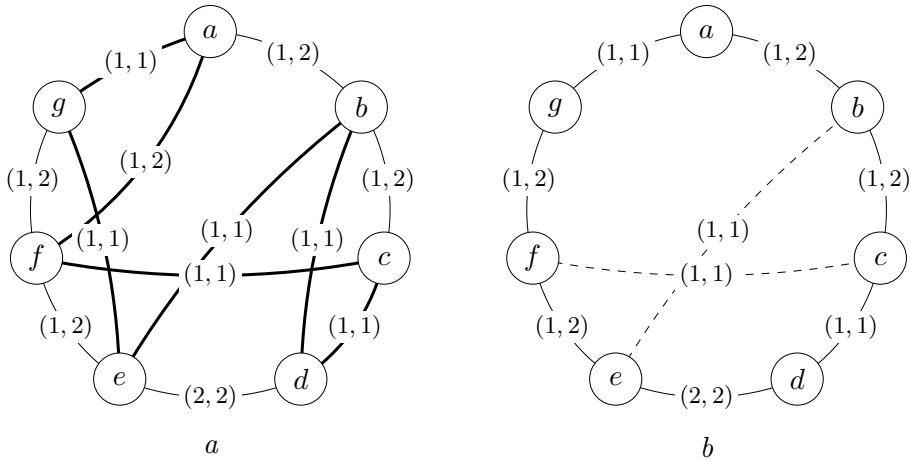


Figure 3. An MOTSP instance (a) and a local optimum (b) for Example 3.2.

optimal tour for the first component. However, this sequence is not a Pareto local optimum since it is weakly dominated by the sequence (a, b, e, d, c, f, g) obtained by removing the edges (b, c) and (e, f) from s , and adding the edges (b, e) and (c, f) (the latter two edges are dashed in Figure 3b) with an objective value vector $(8, 10)$.

These two examples give a negative answer to the initial question posed in this section, that is, solutions returned by applying an iterative improvement algorithm to each objective may not result in a Pareto local optimum set. However, an acceptance criterion that accepts *sets of solutions* can, in fact, identify Pareto local optima. Two variants are discussed in the next two sections.

4. A basic iterative improvement algorithm for MOCO

One common approach to solve MOCO problems is to use the (weak) component-wise ordering when comparing neighboring solutions (Knowles and Corne, 2000; Angel et al., 2004b; Paquete et al., 2004). In that case, the acceptance criterion is to accept a neighbor if it is non (weakly) dominated; since in the local search process we can expect to have more than one solution for which this holds, an additional data structure, which is usually called *archive*, is used to maintain the current set of solutions.

How do these two features, component-wise acceptance criterion and the usage of an archive, relate to our notions of Pareto local optimum set? To answer this question we have to define an analogy to iterative improvement algorithms for the single objective case. In fact, in the literature there are almost no iterative improvement algorithms for MOCO problems that stop in a set of solutions that cannot be improved locally, an exception being the Pareto local search (PLS) algorithm proposed in (Paquete et al., 2004). In this section, we define a basic iterative improvement algorithm for MOCO problems that is based on a general neighborhood between sets of solutions and a dominance relation between sets of points. In the next section, we discuss the relationship

of the outcomes of PLS to the outcomes of this basic iterative improvement algorithm.

Certainly, a suitable stopping criterion for a basic iterative improvement algorithm for MOCO problems is to terminate once a Pareto local optimum set is found. Hence, such an algorithm should, at each iteration, select a set of non weakly dominated solutions and work on a set of solutions. We define $U_{S'} = \bigcup_{s \in S'} N(s)$ for a given arbitrary neighborhood function N and a set S' of feasible solutions. Then, a natural neighborhood of S' , identified by $\Upsilon(S')$, is the power set of $S' \cup U_{S'}$, that is,

$$\Upsilon(S') = 2^{S' \cup U_{S'}}.$$

The next step is to define an acceptance criterion for sets of solutions that are neighbors with respect to Υ . For this task, the outperformance relations proposed in (Hansen and Jaszkiewicz, 1998) and later extended in (Zitzler et al., 2003) are useful. Given two arbitrary point sets A and B in a K -dimensional objective space, the following relations are introduced:

1. $A \overset{\circ}{<} B$ if every b in B is strictly dominated by at least one a in A , i.e., A *strictly dominates* B ;
2. $A \overset{\circ}{\leq} B$ if every b in B is dominated by at least one a in A , i.e., A *dominates* B ;
3. $A \overset{\circ}{\triangleleft} B$ if every b in B is weakly dominated by at least one a in A and $A \neq B$, i.e., A *is better than* B ;
4. $A \overset{\circ}{\leq\leq} B$ if every b in B is weakly dominated by at least one a in A , i.e., A *weakly dominates* B ;
5. $A \overset{\circ}{\parallel} B$ if neither $A \overset{\circ}{\leq\leq} B$ nor $B \overset{\circ}{\leq\leq} A$, i.e., A and B are *incomparable*.

Moreover, $A \overset{\circ}{<} B \Rightarrow A \overset{\circ}{\leq} B \Rightarrow A \overset{\circ}{\triangleleft} B \Rightarrow A \overset{\circ}{\leq\leq} B$ (Zitzler et al., 2003). Figure 4 illustrates these relations between sets of points in the objective space with two minimizing objectives in some particular cases. In Plot 1, any black point is strictly dominated by a white one. Plot 2 shows that there is a black point (the minimum with respect to F_1) that is dominated (but not strictly) by another two white points. The remaining black points are strictly dominated by some white point. Plot 3 shows that all black points are weakly dominated by white points, although not all black points (the gray points belong to both sets) are dominated, and hence, the two sets are different. In Plot 4, both sets are equal and the white points weakly dominate the black points and vice-versa. Finally, the plot on the bottom shows two sets that are incomparable, i.e., one can neither say that all black points dominate the white ones nor vice-versa.

Based on the neighborhood Υ and on any of the outperformance relations above, local search algorithms can now be defined. In fact, depending on the particular relation chosen, different outcomes can be expected. Consider first

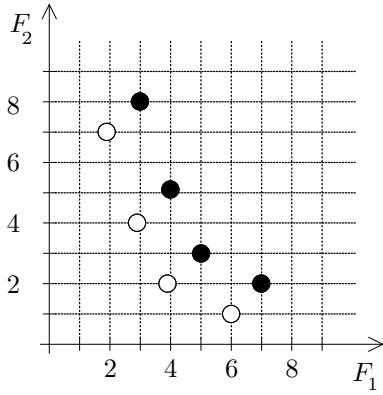
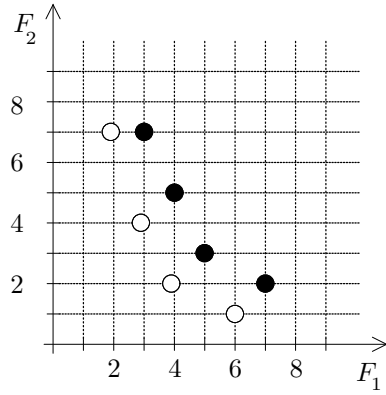
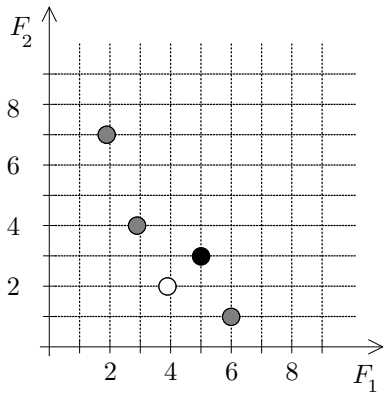
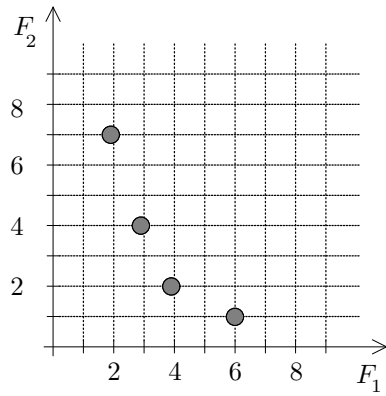
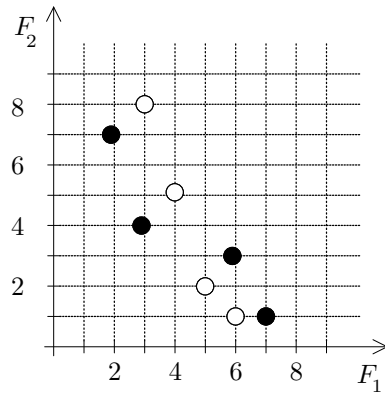
1. $A \overset{\circ}{<} B$ 2. $A \overset{\circ}{\leq} B$ 3. $A \overset{\circ}{<} B$ 4. $A \overset{\circ}{\leq} B$ (and $B \overset{\circ}{\leq} A$)5. $A \overset{\circ}{\parallel} B$

Figure 4. Illustration of relations between a set A (white points) and a set B (black points) of nondominated objective value vectors. A gray point belongs to both sets. Note that each plot shows a special case that cannot be classified w.r.t. the preceding relation. For example, $A \overset{\circ}{\leq} B$ is true for plots 1 to 4, but the special case in the plot 4 cannot be classified with respect to the relations $\overset{\circ}{<}$, $\overset{\circ}{\leq}$, and $\overset{\circ}{<}$. See the text for further details.

Algorithm 1 Basic Iterative Improvement (BII) for MOCOs

```

1: Input:  $(F, \Upsilon, S)$ 
2: while  $\exists S' \in \Upsilon(S) \wedge F(S') \overset{\circ}{\triangleleft} F(S)$  do
3:    $S = S'$ 
4: return  $S$ 

```

the case of the relations $\overset{\circ}{\triangleleft}$ and $\overset{\circ}{\trianglelefteq}$; it is easy to see that once a single Pareto global optimum is found, no additional Pareto globally optimal solution is attainable, since the sets that contain any such additional Pareto global optima are not dominated. Also the relation $\overset{\circ}{\trianglelefteq}$ appears not to be appropriate. Consider, for example, two neighbor sets of solutions having the same objective function vectors; since both weakly dominate each other, a cycle occurs and such an iterative improvement algorithm would never terminate. Therefore, the relation $\overset{\circ}{\triangleleft}$ is the most appealing to be exploited by an iterative improvement algorithm.

In its simplest form, the algorithm starts from an arbitrary, initial solution set, and it iteratively searches neighbors to find *better* solution sets until a Pareto local optimum set is found. An outline of such a basic iterative improvement algorithm (BII) is given by Algorithm 1. Interestingly, the final set cannot be extended by any solution that is a neighbor to any of its elements, because any such set would contain a (weakly) dominated solution. Such a set is called here a *maximal Pareto local optimum set*. In other words, $S' \subseteq S$ is a *maximal Pareto local optimum set* with respect to a neighborhood N if, and only if,

$$(\forall s' \in U_{S'}) (\exists s \in S') F(s) \leq F(s').$$

An equivalent definition is to say that S' is a maximal Pareto local optimum set if there is no other S^* in $\Upsilon(S')$ such that S^* is *better than* S' . Finally, note that a strict Pareto global optimum set is also a maximal Pareto local optimum set.

A main drawback of BII is the exponential size of the neighborhood $\Upsilon(S')$ with respect to the number of elements in S' ($2^{O(|S'|)}$ with $N(s)$ having polynomial size) and, hence, it may be time consuming to search the full neighborhood already for a small sized set. To reduce the time required for finding an improving neighbor, we need to find an iterative improvement algorithm that searches an interesting subset of the neighborhood in polynomial time with respect to the number of elements in S' .²

² Note that S' may, in the worst case, be of exponential size with respect to the instance size for many MOCO problems, such as the multiobjective versions of the SHORTEST PATH PROBLEM (Hansen, 1979), MINIMUM SPANNING TREE PROBLEM (Hamacher and Ruhe, 1994) and the TRAVELING SALESMAN PROBLEM (Emelichev and Perepelitsa, 1992). However, often the neighborhood of a solution is of polynomial size.

5. Soundness and Completeness of Pareto Local Search

Pareto Local Search (PLS) is a simple iterative improvement algorithm for MOCO problems that was first proposed and experimentally tested in (Paquete et al., 2004) for the MOTSP and later in (Paquete and Stützle, 2006) for the MULTIOBJECTIVE QUADRATIC ASSIGNMENT PROBLEM. PLS resembles some other approaches to MOCO such as BLS (Angel et al., 2004b), SEMO and FEMO (Laumanns et al., 2002), PAES (Knowles and Corne, 1999), the local search algorithm mentioned in (Ehrgott and Gandibleux, 2004), as well as the post-processing steps applied in (Hamacher and Ruhe, 1994; Jörnsten et al., 1996; Talbi, 2003). This section shows that PLS examines a polynomial size subset of neighbors at each iteration and that it outputs a maximal Pareto local optimum set.

The essential idea of PLS is to extend the current set of solutions S' (maintained in an archive) by searching subsets of $\Upsilon(S')$ considering one solution in S' after another. Dominated solutions are eliminated as soon as possible. Note that if a set S^* is obtained by eliminating dominated solutions from S' , then $S^* \overset{\circ}{\triangleleft} S'$. The outline of PLS is given in Algorithm 2. Its pseudo-code is presented in a slightly different way as in (Paquete et al., 2004) to ease the proofs of completeness and soundness.

PLS splits the set of solutions S' in the archive into V_T and V_F , where V_T is the subset of solutions in S' for which the complete neighborhood was already evaluated (solutions marked as *visited*) and V_F contains the remaining solutions in S' (solutions marked as *not visited*). PLS starts from some initial solution and flags it as not visited (that is, includes it in V_F); V_T is initially empty. Then, at each iteration through the repeat–until loop, PLS evaluates all neighbors of one solution s in V_F . For each $s' \in N(s)$, it checks whether there is another solution in $N(s) \cup V_F \cup V_T$ that weakly dominates it (line 10). If this is not the case, s' is added to W (line 11) and it will be chosen in a following iteration if it is still nondominated with respect to $N(s) \cup V_F \cup V_T$ or with respect to V_T (line 12). For convenience, W^i denotes the set W and V^i the set $V_T \cup V_F$ at the end of iteration i (line 14).

Note that PLS removes dominated solutions of V_T and V_F at each iteration and V_T and V_F contain only solutions that are nondominated by any solution in $V_F \cup V_T$. The following proposition shows that PLS is indeed an iterative improvement algorithm.

PROPOSITION 5.1. *PLS is an iterative improvement algorithm according to the neighborhood Υ and the relation $\overset{\circ}{\triangleleft}$.*

Proof. For proving Proposition 5.1, it is sufficient to show the two following assertions:

1. At any iteration i , no neighbor is accepted at all or a neighbor in $\Upsilon(V^{i-1})$ is accepted;
2. If a neighbor is accepted, it improves the current set of solutions according to $\overset{\circ}{\triangleleft}$.

Algorithm 2 Pareto Local Search (PLS)

```

1: Input:  $(F, N, s)$ 
2:
3:  $V_F \leftarrow \{s\}$ 
4:  $V_T$  is empty
5: repeat
6:   choose  $s$  in  $V_F$ 
7:    $V_T \leftarrow V_T \cup \{s\}$ 
8:    $W$  is empty
9:   for all  $s'$  in  $N(s)$  do
10:    if  $(\nexists s'' \in W \cup V_T \cup V_F) F(s'') \leq F(s')$  then
11:       $W \leftarrow W \cup \{s'\}$ 
12:     $V_T \leftarrow \{s' \mid s' \in V_T : (\nexists s'' \in W) F(s'') \leq F(s')\}$ 
13:     $V_F \leftarrow (V_F \setminus \{s\}) \cup W$ 
14:     $V_F \leftarrow \{s' \mid s' \in V_F : (\nexists s'' \in W) F(s'') \leq F(s')\}$ 
15:  until  $V_F$  is empty
16: return  $V_T$ 

```

For the first assertion, a subset of $\Upsilon(V^{i-1})$ is examined until an improving set of solutions is found. At the end of each iteration, the two following cases can occur:

- if W^i is not empty, then $W^i \subseteq N(s)$.
- if W^i is empty, then no neighbor solution is accepted (thus, $V^i = V^{i-1}$);

Since in the first case it holds that $V^i \subseteq V^{i-1} \cup W^i \subseteq V^{i-1} \cup N(s)$, then, given the definition of the neighborhood Υ , $V^i \in \Upsilon(V^{i-1})$. In the second case, the next iteration of the repeat–until loop is invoked. If V_F is empty, then the algorithm stops.

For the second assertion, we have that W^i is not empty. Then,

$$V^i = (V^{i-1} \cup W^i) \setminus Z$$

where

$$Z = \{s \mid s \in V^{i-1} \cup W^i : (\exists s' \in V^{i-1} \cup W^i) F(s') \leq F(s)\}.$$

Given the definition of the relation $\overset{\circ}{\triangleleft}$ and that $V^{i-1} \cup W^i \overset{\circ}{\leq} Z$, it follows that $V^i \overset{\circ}{\triangleleft} V^{i-1} \cup W^i$. Note that if Z is empty, then V^i is set to $V^{i-1} \cup W^i$. Given the relation $\overset{\circ}{\triangleleft}$ and that W^i contains at least one solution not dominated by any solution in V^{i-1} , it follows that $V^i \overset{\circ}{\triangleleft} V^{i-1}$. \blacksquare

Both termination and soundness (the solution returned is a local optimum) of PLS can be proved. Before proceeding, it is convenient to introduce the following property, which says that if a solution is not dominated by any solution in V^i , then no solution in V^j can have dominated that solution in any previous iteration $j < i$.

LEMMA 5.2. *Given a solution s , for any iteration i it holds that*

$$(\nexists s' \in V^i) F(s') \leq F(s) \Rightarrow (\forall j < i) (\nexists s' \in V^j) F(s') \leq F(s).$$

Proof. This Lemma is proved by induction: For $i = 0$ this is trivially true. Suppose that this property holds for V^{i-1} , then it also holds for V^i :

$$\begin{aligned} & (\nexists s' \in V^i) F(s') \leq F(s) && \Rightarrow \\ \left[\text{Since } V^i = (V^{i-1} \cup W^i) \setminus Z, (Z \text{ defined as in proof to Proposition 5.1}) \right] & & & \\ \Rightarrow & (\nexists s' \in V^{i-1} \cup W^i) F(s') \leq F(s) && \Rightarrow \\ \Rightarrow & (\nexists s' \in V^{i-1}) F(s') \leq F(s) && \Rightarrow \\ \left[\text{Inductive hypothesis} \right] & & & \\ \Rightarrow & (\forall j < i) (\nexists s' \in V^j) F(s') \leq F(s) && \end{aligned}$$

■

The two following results can now be proved.

THEOREM 5.3. *The algorithm PLS always terminates*

Proof. Given that the set of feasible solutions is finite, it suffices to show that a solution s is at most chosen once in line 6 of Algorithm 2. This means that s will not be inserted in V_F after having been inserted in V_T (line 7). Considering any iteration, a solution is inserted in V_F only if it is included previously in W . The goal is to show that, once a solution s is inserted in V_T , it will never be included again to W : Suppose that the same solution s is inserted in W^i (in the i -th iteration). This means that there does not exist any other solution s' in V^{i-1} such that $F(s') \leq F(s)$. But since s has been inserted in V_F and, hence, in V^{i-1} in a previous step (and because $F(s) \leq F(s)$) this would be a contradiction to Lemma 5.2. ■

THEOREM 5.4. *The algorithm PLS returns a maximal Pareto local optimum set.*

Proof. The soundness of PLS is shown by contradiction and by using Lemma 5.2. Suppose that the archive at the end of Algorithm 2 (V^h) is not a maximal Pareto local optimum set. This can only be the case if there exists a solution $s \notin V^h$ such that $\nexists s'' \in V^h . F(s'') \leq F(s)$ and there is a solution s' in V^h such that s belongs to $N(s')$. Hence, s' must have been inserted into V_T at some step i . At that step also $N(s')$ is evaluated, and the following two scenarios can occur: (i) s is inserted in W^i or (ii) a solution s^* in $N(s')$ is inserted in W^i for which it holds that $F(s^*) \leq F(s)$. In either of the two cases, s or s^* would be accepted to V^i . If s is accepted to V^i and it is not in V^h , then this can only happen because in some iteration $i < j \leq h$ a solution was added to V^j that

dominated s . Because of Lemma 5.2, this would be a contradiction. If, however, s^* is accepted to V^i , this would again contradict Lemma 5.2, since at iteration i a solution that dominates s is added (and hence, from iteration i on there must be a solution in V that dominates s). ■

PLS takes $O(|N(s) \cup V_T \cup V_F|^2)$ steps at each iteration.³ Consequently, it visits a polynomial-size subset of neighbors of Υ . Furthermore, if $N(s)$ and $V_T \cup V_F$ are of polynomial size, PLS runs in polynomial time with respect to the number of solutions in the archive per iteration. However, the size of the archive may not be bounded by a polynomial in the instance size. In that case, the archive of PLS (and, hence, the complexity of PLS) would grow exponentially.

Thus, a desired feature of PLS may be to have a polynomially *bounded* archive where only the most *interesting* solutions are kept. Bounded archives are used frequently in *multiobjective evolutionary algorithms* (Fonseca et al., 2003; Zitzler et al., 2001), where often only nondominated solutions that contribute towards having a *spread* set of solutions are accepted (see (Knowles and Corne, 2004) for a study on the desired properties of an archive). However, a maximal Pareto local optimum set may not be obtained, since the size of the archive is bounded. Moreover, certain notions of spread between solutions can lead to an undesired cycling behavior as shown in the following example.

EXAMPLE 5.5. *Assume that we have an algorithm with an archive restricted to m solutions, the acceptance criterion follows the rules given in PLS, and the spacing measure M (Schott, 1995) is used to evaluate the spread of solutions. The spacing measure is defined as*

$$M(\{s_1, \dots, s_m\}) = \sqrt{\frac{1}{m-1} \sum_{i=1}^m (\bar{d} - d(s_i))^2} \quad (2)$$

where

$$d(s_i) = \min_{1 \leq j \neq i \leq m} \sum_{k=1}^K \left(|f_k(s_i) - f_k(s_j)| \right), \quad (3)$$

Given a set of solutions $\{s_1, \dots, s_m\}$ in the archive, \bar{d} is the average value of $d(s_1), \dots, d(s_m)$. The lower the value of the spacing measure, the more spread are the solutions in the objective space (Schott, 1995). Moreover, solutions that are minimum to each of the objectives are maintained (note that this is a desired feature of an archive (Knowles and Corne, 2004)). Once the archive is full, the modified PLS would replace a solution by another one if either it leads to a better set, or, if the two sets are incomparable, the spacing measure is decreased. Figure 5 gives a sequence of iterations in which this algorithm cycles with $m = 3$. Let $\{s_1, s_2, s_3\}$ be the set of solutions in the archive at a certain iteration with $M(\{s_1, s_2, s_3\}) = 0$. Since s_4 is neighbor of s_3 and dominates it, s_4 replaces s_3 and $M(\{s_1, s_2, s_4\}) = 12$. Then, $\{s_1, s_2, s_5\}$ replaces $\{s_1, s_2, s_4\}$, since s_5 is a neighbor of s_4 and, despite being nondominated, the spacing measure becomes

³ This time complexity can be reduced if the algorithm of Kung *et al.* (1975) is used for removing dominated solutions for $K < 5$.

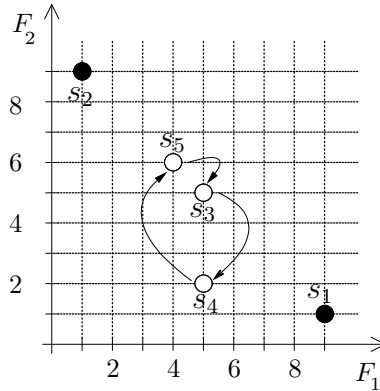


Figure 5. Illustration of a cycle of PLS with an archive of three solutions

lower if solution s_5 replaces s_4 ($M\{s_1, s_2, s_5\} = 5.33$). Finally, given that s_5 is a neighbor of s_3 , $M(\{s_1, s_2, s_3\}) < M\{s_1, s_2, s_5\}$; then s_3 replaces s_5 , resulting in a cycle of length three.

Certainly, also other algorithms that use a similar acceptance criterion may potentially be affected by such a cycling behavior. Thus, this example shows that some caution should be taken when using measures on the spread of solutions and bounded archives.

6. Discussion and Further Work

Recent experimental results show that PLS is a practically viable algorithm and that it can reach, from a solution quality point of view, competitive performance to more complex SLS methods (Paquete et al., 2004; Paquete and Stützle, 2006). Still, a disadvantage of PLS is that the archive may grow very strongly, leading to high computation times. While the usage of bounded archives may be a practical way to go, as shown in the previous section, some care has to be taken with this option.

If we reconsider the fact that PLS stops in a local optimum for MOCO problems, it is clear that it can serve as the basis for general-purpose SLS methods (Hoos and Stützle, 2004), just as it is the case in single-objective optimization problems. An adaptation to several well-known SLS methods could be as follows:

- to *Simulated Annealing* by accepting a worse neighbor set of solutions with a certain probability;
- to *Tabu Search* by using a tabu mechanism based on sets of solutions that cannot be visited again for some number of iterations;
- to *Iterated Local Search* by *perturbing* solutions and possibly by removing some others, once a maximal Pareto local optimum set is found, and to restart PLS from the perturbed solution;

- to *Variable Neighborhood Search* (Hansen and Mladenović, 2002) by changing the definition of neighborhood once a maximal Pareto local optimum set is found;
- to *Memetic Algorithms* (Moscato, 1999) by maintaining a population of maximal Pareto local optimum sets and generating offsprings by set union or set intersection of such local optima.

Considering the possibly high computation times of PLS, for these extensions the usage of bounded archives should possibly be considered. Another possible issue for PLS is the initial solution set. In fact, simplified versions of PLS have already been used successfully by seeding them with solutions returned from scalarizations of MOCO problems. Examples of such successful combinations are (Hamacher and Ruhe, 1994; Jörnsten et al., 1996; Paquete and Stützle, 2003), although further work in this direction is certainly required.

There are various other directions for further research. Although one can find in the literature already many well performing approaches based on local search algorithms for MOCO, there is still a need to explain their success from a more theoretical side. Interesting directions for future research include the following.

- *Approximate Local Search*. Given that there could be an exponential sized Pareto local optimum set in PLS, an interesting idea would be to build local search algorithms that obtain a $(1 + \epsilon)$ -*approximation* to the Pareto global optimum set in polynomial time as in (Erlebach et al., 2002; Papadimitriou and Yannakakis, 2000; Ruhe and Fruhwirth, 1990; Warburton, 1987). Already some results are known for the MOTSP with two objectives where the edges have distances 1 or 2 (Angel et al., 2004a). Further work for other problems is clearly required.
- *Convergence Properties*. A well known result is that Simulated Annealing and some other SLS methods converge to the global optimum in the single objective case. Despite the little relevance for practice, one may ask whether similar results can be obtained for the multiobjective versions.
- *Bounded archiving*. Many approaches in the literature present different ways of bounding the archive. It would be interesting to know if such algorithms may cycle as shown in our example.

Acknowledgments

This work was supported by the Metaheuristics Network, a Research Training Network funded by the Improving Human Potential programme of the CEC, grant HPRN-CT-1999-00106. The information provided is the sole responsibility of the authors and does not reflect the Community’s opinion. The Community is not responsible for any use that might be made of data appearing in this publication. Thomas Stützle acknowledges support of the Belgian FNRS, of which he is a research associate.

References

- Aarts, E. H. L. and J. K. Lenstra (eds.): 1997, *Local Search in Combinatorial Optimization*. John Wiley & Sons, Chichester, UK.
- Angel, E., E. Bampis, and L. Gourvés: 2004a, ‘Approximating the Pareto Curve with Local Search for the Bicriteria TSP(1,2) Problem’. *Theoretical Computer Science* **310**, 135–146.
- Angel, E., E. Bampis, and L. Gourvés: 2004b, ‘A dynasearch neighborhood for the bicriteria traveling salesman problem’. In (Gandibleux et al., 2004), pp. 153–176.
- Deb, K.: 1999, ‘Multi-objective genetic algorithms: Problem difficulties and construction of test problems.’. *Evolutionary Computation* **7**(3), 205–230.
- Ehrgott, M.: 2000, *Multicriteria Optimization*, Vol. 491 of *Lecture Notes in Economics and Mathematical Systems*. Heidelberg: Springer Verlag.
- Ehrgott, M. and X. Gandibleux: 2004, ‘Approximative Solution Methods for Combinatorial Multicriteria Optimization’. *TOP* **12**(1), 1–90.
- Emelichev, V. and V. Perepelitsa: 1991, ‘Complexity of Vector Optimization Problems on Graphs’. *Optimization* **6**, 903–918.
- Emelichev, V. and V. Perepelitsa: 1992, ‘On the Cardinality of the Set of Alternatives in Discrete Many-criterion Problems’. *Discrete Mathematics and Applications* **2**(5), 461–471.
- Erlebach, T., H. Kellerer, and U. Pferschy: 2002, ‘Approximating multiobjective knapsack problems’. *Management Science* **48**(12), 1603–1612.
- Fonseca, C., P. Fleming, E. Zitzler, K. Deb, and L. Thiele (eds.): 2003, ‘Evolutionary Multi-criterion Optimization (EMO 2003)’, Vol. 2632 of *LNCS*. Springer Verlag.
- Gandibleux, X., M. Sevaux, K. Sörensen, and V. T’kindt (eds.): 2004, ‘Metaheuristics for Multiobjective Optimisation’, Vol. 535 of *LNEMS*. Springer Verlag.
- Hamacher, H. and G. Ruhe: 1994, ‘On spanning tree problems with multiple objectives’. *Annals of Operations Research* **5**, 209–230.
- Hansen, M. and A. Jaszkiwicz: 1998, ‘Evaluating the quality of approximations to the non-dominated set’. Technical Report IMM-REP-1998-7, Institute of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark.
- Hansen, P.: 1979, ‘Bicriterion path problems’. In: *Multiple Criteria Decision Making Theory and Application*, Vol. 177 of *LNEMS*. pp. 109–127, Springer Verlag.
- Hansen, P. and N. Mladenović: 2002, ‘Variable Neighborhood Search’. In: F. Glover and G. Kochenberger (eds.): *Handbook of Metaheuristics*. Kluwer Academic Publishers, Norwell, MA, pp. 145–184.
- Hoos, H. H. and T. Stützle: 2004, *Stochastic Local Search—Foundations and Applications*. Morgan Kaufmann Publishers, San Francisco, CA.
- Jörnsten, K., K. Andersen, and M. Lind: 1996, ‘On bicriterion minimal spanning trees: An approximation’. *Computers & Operations Research* **23**(12), 1171–1182.
- Knowles, J. and D. Corne: 1999, ‘The Pareto archived evolution strategy: A new base line algorithm for multiobjective optimisation’. In: *Proceedings of the 1999 Congress on Evolutionary Computation (CEC’99)*. Washington, pp. 98–105.
- Knowles, J. and D. Corne: 2000, ‘Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy’. *Evolutionary Computation* **8**(2), 149–172.
- Knowles, J. and D. Corne: 2004, ‘Bounded Pareto Archiving: Theory and Practice’. In (Gandibleux et al., 2004), pp. 39–64.
- Kung, H., F. Luccio, and F. Preparata: 1975, ‘On Finding the Maxima of a Set of Vectors’. *Journal of the Association for Computing Machinery* **22**(4), 469–476.
- Laumanns, M., L. Thiele, E. Zitzler, E. Welzl, and K. Deb: 2002, ‘Running Time Analysis of Multi-objective Evolutionary Algorithms on a Simple Discrete Optimization Problem’. In: J. M. Guervos et al. (eds.): *Proceedings of PPSN-VII*, Vol. 2439 of *LNCS*. pp. 44–53, Springer Verlag.
- Moscato, P.: 1999, ‘Memetic Algorithms: A Short Introduction’. In: D. Corne, M. Dorigo, and F. Glover (eds.): *New Ideas in Optimization*. McGraw Hill, London, UK, pp. 219–234.

- Papadimitriou, C. H. and M. Yannakakis: 2000, 'On the Approximability of Trade-offs and Optimal Access of Web Sources'. In: *Proceedings of the 41st Annual Symposium on Foundations of Computer Science (FOCS 2000)*. pp. 86–92.
- Paquete, L., M. Chiarandini, and T. Stützle: 2002, 'A Study of Local Optima in the Multiobjective Traveling Salesman Problem'. Technical Report AIDA-02-07, Fachgebiet Intellektik, Fachbereich Informatik, Technische Universität Darmstadt. Presented at the Multiobjective Metaheuristics Workshop (MOMH 2002), Paris, November 4–5, 2002.
- Paquete, L., M. Chiarandini, and T. Stützle: 2004, 'Pareto Local Optimum Sets in the Biobjective Traveling Salesman Problem: An Experimental Study'. In (Gandibleux et al., 2004), pp. 177–200.
- Paquete, L. and T. Stützle: 2003, 'A Two-Phase Local Search for the Biobjective Traveling Salesman Problem'. In (Fonseca et al., 2003), pp. 479–493, Springer Verlag.
- Paquete, L. and T. Stützle: 2006, 'A Study of Stochastic Local Search Algorithms for the Biobjective QAP with Correlated Flow Matrices'. *European Journal of Operational Research* **169**(3), 943–959.
- Ruhe, G. and B. Fruhwirth: 1990, ' ϵ -optimality for bicriteria programs and its application to minimum cost flows'. *Computing* **44**(1), 21–34.
- Schott, J. R.: 1995, 'Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization'. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts.
- Serafini, P.: 1986, 'Some Considerations About Computational Complexity for Multiobjective Combinatorial Problems'. In: *Recent Advances and Historical Development of Vector Optimization*, Vol. 294 of *LNEMS*. Springer-Verlag, pp. 222–231.
- Talbi, E. G.: 2003, 'A Hybrid Evolutionary Approach for Multicriteria Optimization Problems: Application to the Flow Shop'. In (Fonseca et al., 2003), pp. 416–428, Springer Verlag.
- Warburton, A.: 1987, 'Approximation of Pareto optima in multi-objective shortest-path problems'. *Operations Research* **35**(1), 70–79.
- Zitzler, E., K. Deb, L. Thiele, C. A. Coello, and D. Corne (eds.): 2001, 'Evolutionary Multi-criterion Optimization (EMO 2001)', Vol. 1993 of *LNCS*. Springer Verlag.
- Zitzler, E., L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Fonseca: 2003, 'Performance Assessment of Multiobjective Optimizers: An Analysis and Review'. *IEEE Transactions on Evolutionary Computation* **7**(2), 117–132.

