Journal of Heuristics, 6: 295–310 (2000)


# Performance of the MOSA Method for the Bicriteria Assignment Problem

D. TUYTTENS, J. TEGHEM, PH. FORTEMPS AND K. VAN NIEUWENHUYZE
*Laboratory of Mathematics and Operational Research, Faculté Polytechnique de Mons, 9, rue de Houdain, B-7000 Mons, Belgium*
*email: tuyttens@mathro.fpms.ac.be; teghem@mathro.fpms.ac.be; fortemps@mathro.fpms.ac.be*

*Abstract*

The classical linear Assignment problem is considered with two objectives. The aim is to generate the set of efficient solutions. An exact method is first developed based on the two-phase approach. In the second phase a new upper bound is proposed so that larger instances can be solved exactly. The so-called MOSA (Multi-Objective Simulated Annealing) is then recalled; its efficiency is improved by initialization with a greedy approach. Its results are compared to those obtained with the exact method. Extensive numerical experiments have been realized to measure the performance of the MOSA method.

**Key Words:** multi-objective programming, assignment problem, simulated annealing

## 1. Introduction

Until recently, multi-objective combinatorial optimization (MOCO) did not receive much attention in spite of its potential applications. The reason is probably due to specific difficulties of MOCO models as pointed out in a recent survey (Ulungu and Teghem, 1994) concerning this field.

The aim of the present paper is to develop and to compare two procedures, an exact method and a heuristic one, to generate the set of efficient solutions of a particular bi-objective MOCO problem, the Assignment problem.

This is a basic well known combinatorial optimization problem, important for applications and as a sub-problem of more complicated ones, like transportation problem, distribution problem or travelling salesman problem. Moreover, its mathematical structure is very simple and there exist efficient algorithms to solve it in the single objective case, like the Hungarian method (Teghem, 1996).

In a bi-objective framework, the Assignment problem can be formulated as

$$\begin{cases} \text{“min”} z_k(X) = \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}^k x_{ij} \quad k = 1, 2 \\ \displaystyle\sum_{j=1}^{n} x_{ij} = 1 \quad i = 1, \ldots, n \\ \displaystyle\sum_{i=1}^{n} x_{ij} = 1 \quad j = 1, \ldots, n \\ x_{ij} \in \{0, 1\} \end{cases} \tag{P}$$

where $c_{ij}^k$ are non negative integers and $X = (x_{11}, \ldots, x_{nn})$.

A solution $X^*$ of problem $(P)$ is *efficient* if there does not exist any other feasible solution $X$ such that $z_k(X) \leq z_k(X^*),\ k = 1, 2$, with at least one strict inequality. We will denote by $E(P)$ the set of efficient solutions of problem $(P)$.

In multi-objective integer linear programming, it is necessary to distinguish two kinds of efficient solutions (cf. Ulungu and Teghem, 1994):

- the set $SE(P)$ of *supported* efficient solutions which are optimal solutions of the parameterized single objective problem

$$\begin{cases} \min z_\lambda(X) = \lambda_1 z_1(X) + \lambda_2 z_2(X) \\ \sum_{j=1}^{n} x_{ij} = 1 \quad i = 1, \ldots, n \\ \sum_{i=1}^{n} x_{ij} = 1 \quad j = 1, \ldots, n \\ x_{ij} \in \{0, 1\} \\ \lambda_1 > 0,\ \lambda_2 > 0 \end{cases} \tag{$P_\lambda$}$$

- the set $NSE(P) = E(P) \backslash SE(P)$ of *non-supported efficient solutions* which cannot be found by optimization of problem $(P_\lambda)$. These non-supported efficient solutions are necessarily located in the triangles $\triangle Z_r Z_s$ generated in the objective space by two successive supported efficient solutions, as represented in figure 1.

It is important to underline that such a distinction is still necessary even if the constraints of the problem satisfy the so-called "totally unimodular" property or "integrality" property (see Teghem, 1996); when this property is verified, the integrality constraints of the single

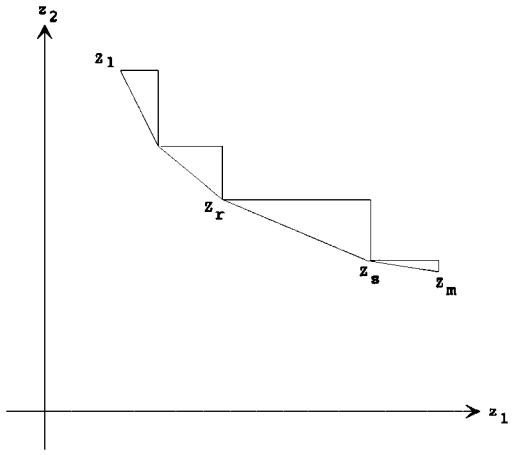


*Figure 1.* Supported efficient solutions $Z_1, \ldots, Z_m$ and potential regions $\triangle Z_r Z_s$ of non supported efficient solutions.

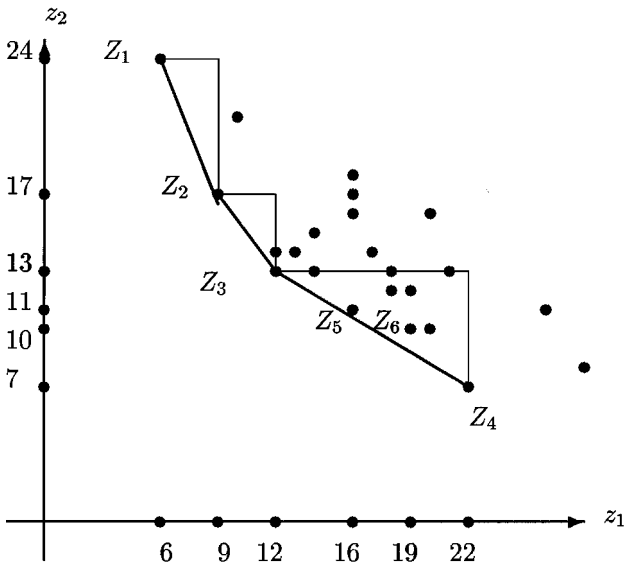


*Figure 2.* The feasible points in the objective space for the didactic example.

objective problem can be relaxed without any deterioration of the objective function, i.e. the optimal values of the variables are integer even if only the linear relaxation of the problem is solved.

It is well known that the single objective Assignment problem—and thus problem $P_\lambda$—satisfies this integrality property. Nevertheless, in the multi-objective framework, there exist non-supported efficient solutions as indicated by the following didactic example:

$$C^{(1)} = \begin{bmatrix} 5 & 1 & 4 & 7 \\ 6 & 2 & 2 & 6 \\ 2 & 8 & 4 & 4 \\ 3 & 5 & 7 & 1 \end{bmatrix} \quad \text{and} \quad C^{(2)} = \begin{bmatrix} 3 & 6 & 4 & 2 \\ 1 & 3 & 8 & 3 \\ 5 & 2 & 2 & 3 \\ 4 & 2 & 3 & 5 \end{bmatrix};$$

The values of the 24 feasible solutions are represented in the objective space in figure 2.

There are four supported efficient solutions, corresponding to points $Z_1$, $Z_2$, $Z_3$ and $Z_4$; two non-supported efficient solutions corresponding to points $Z_5$ and $Z_6$; the eighteen other solutions are non efficient.

Since a few years, a larger attention has been devoted to MOCO problems. In Ulungu (1993), and Ulungu and Teghem (1994, 1995) the authors have proposed some exact methods to determine the whole efficient set $E(P)$ of bicriteria combinatorial problems. They focus on the Assignment and Knapsack problems. For the latter problem, such exact method has been recently improved and implemented for the first time (Visée et al., 1998). However, as pointed out, it is unrealistic to extend these methods to MOCO problems with more than two criteria or with a large amount of variables.

It makes sense, in a pragmatic spirit, to consider "approximate" methods like metaheuristics—Simulated Annealing (S.A.), Tabu Search (T.S.), Genetic Algorithms (G.A.), ... (Pirlot, 1996)—since they provide, for the single objective problem, excellent solutions in a reasonable time. On the other hand, their implementation is rather independent of the problem mathematical structure.

The idea of adapting these methods in a multiobjective framework has been exploited independently by Ulungu (1993) and Ulungu, Teghem, and Fortemps (1995) and by Serafini (1992). The latter only examines the notion of acceptance probability, whilst in Ulungu, Teghem, and Fortemps (1995) a complete Multi-Objective Simulated Annealing (MOSA) algorithm has been designed. The MOSA method has been improved and extensively tested in Ulungu et al. (1998) on the Knapsack problem by comparison with the exact results obtained in Visée et al. (1998).

Recently, the multi-objective Knapsack problem has been tackled by other research teams: Czyzak, Hapke, and Jaszkiewicz (1994) and Czyzak and Jaszkiewicz (1998) proposed another way to adapt S.A. to a multi-objective framework; independently, Hansen (1997), Gandibleux, Mezdaoui, and Fréville (1996), and Ben Abdelaziz, Chaouachi, and Krichen (1997) did the same with T.S., the latter combining also T.S. and G.A.

Let us note that in Ulungu, Teghem, and Ost (1998), an interactive version of MOSA has been used to solve an industrial problem.

The aim of this paper is to measure the performance of the MOSA method on the bi-objective Assignment problem.

In Section 2, the two-phase method is presented; the procedure improves on what is done in Ulungu and Teghem (1995), due to a better upper bound of $z_\lambda(X)$ for non supported efficient solutions in triangle $\triangle Z_r Z_s$. The implementation of this exact method authorizes to solve problem till $n = 50$.

The MOSA method is described in Section 3; an improvement is introduced by considering *a greedy step* to define the starting point of S.A.

Extensive numerical results are given in Section 4 in order to compare the two methods and to analyze the performance of the MOSA method.

## 2. The two-phase method

This procedure will exactly generate the set $E(P)$: the two phases will determine respectively $SE(P)$ and $NSE(P)$.

### *2.1. The first phase*

Let $S \cup S'$ be the list of supported efficient solutions already generated: $S$ contains the extreme supported efficient solutions, $S'$ the non extreme ones.

$S$ is initialized with two efficient optimal solutions respectively of objectives $z_1$ and $z_2$, obtained by the Hungarian method (in fact a variant which returns all optimal solutions: see Remark 1 below).

Solution of $S$ are ordered by increasing value of criterion $z_1$; let $X^r$ and $X^s$ be two consecutive solutions in $S$. The single criterion problem $P_\lambda$ is considered with values $\lambda_1 = z_{2r} - z_{2s} > 0$ and $\lambda_2 = z_{1s} - z_{1r} > 0$, and optimized by the Hungarian method.

Let $\{X^{(t)},\ t = 1, \ldots, T\}$ be the set of optimal solutions obtained in this manner and $\{Z_t, t = 1, \ldots, T\}$ their images in the objective space. Two possible cases can arise:

- $\{Z_r, Z_s\} \cap \{Z_t, t = 1, \ldots, T\} = \emptyset$.
  Solutions $X^t$ are new supported efficient solutions; $X^1$ and $X^T$ (provided $T > 1$) are extreme and put in $S$; if $T > 2$, $X^2, \ldots, X^{T-1}$ are put in $S'$. It will be necessary at further steps to consider the pairs $(X^r, X^1)$ and $(X^T, X^s)$.
- $\{Z_r, Z_s\} \subset \{Z_t, t = 1, \ldots, T\}$.
  If $T > 2$, $X^2, \ldots, X^{T-1}$ are new non extreme supported efficient solutions giving the same optimal value as $X^r \equiv X^1$ and $X^s \equiv X^T$ for $z_\lambda(X)$: they are put in $S'$.

The first phase is continued until all pairs $(X^r, X^s)$ of $S$ have been examined without extension of $S$. Finally, we obtain $SE(P) = S \cup S'$.

### *2.2. The second phase using the Hungarian method*

The purpose is to examine each triangle $\triangle Z_r Z_s$ determined by two successive solutions $X^r$ and $X^s$ of $SE(P)$ and to determine the possible non supported solutions whose image lies inside this triangle. As previously, we note

$$z_\lambda(X) = \lambda_1 z_1(X) + \lambda_2 z_2(X)$$

with $\lambda_1 = z_{2r} - z_{2s}$ and $\lambda_2 = z_{1s} - z_{1r}$ and $c_{ij}^{(\lambda)} = \lambda_1 c_{ij}^1 + \lambda_2 c_{ij}^2$.

We recall that the dual problem of $P_\lambda$ (in fact of its linear relaxation which is equivalent to $P_\lambda$) is

$$\max \sum_{i=1}^{n} u_i + \sum_{j=1}^{n} v_j$$
$$u_i + v_j \leq c_{ij}^{(\lambda)}$$

where $u_i$ and $v_j$ are the dual variables associated respectively to constraints $i$ and $j$ of problem $P_\lambda$.

In the first phase, the objective function $z_\lambda(X)$ has been optimized by the Hungarian method giving

- $\tilde{z}_\lambda = \lambda_1 z_{1r} + \lambda_2 z_{2r} = \lambda_1 z_{1s} + \lambda_2 z_{2s}$, the optimal value of $z_\lambda(X)$;
- the optimal value of the reduced cost $\bar{c}_{ij}^{(\lambda)} = c_{ij}^{(\lambda)} - (u_i + v_j)$

At optimality, due to duality properties, we have (see Teghem, 1996) $\bar{c}_{ij}^{(\lambda)} \geq 0$ and $\tilde{x}_{ij} = 1 \Rightarrow \bar{c}_{ij}^{(\lambda)} = 0$.

***2.2.1. First step.*** We consider $L = \{x_{ij} : \bar{c}_{ij}^{(\lambda)} > 0\}$. To generate non supported efficient solution in triangle $\triangle Z_r Z_s$, each variable $x_{ij} \in L$ is candidate to be fixed to 1. Nevertheless,

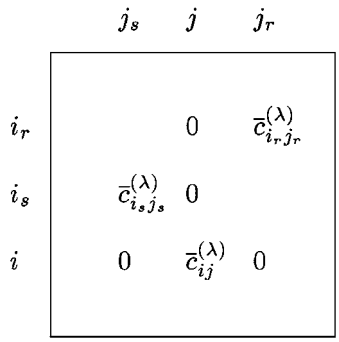


*Figure 3.* Optimal reduced matrix $\bar{c}^{(\lambda)}$ of $P_\lambda$.

a variable can be eliminated if we are sure that the reoptimization of problem $P_\lambda$ will provide a dominated point in the objective space. If $x_{ij} \in L$ is set to 1, a lower bound $l_{ij}$ of the increasing of $\tilde{z}_\lambda$ is given by

$$l_{ij} = \bar{c}^{(\lambda)}_{ij} + \min\left(\bar{c}^{(\lambda)}_{i_r j_r}; \min_{k\neq j} \bar{c}^{(\lambda)}_{i_r k} + \min_{l\neq i} \bar{c}^{(\lambda)}_{l j_r}; \bar{c}^{(\lambda)}_{i_s j_s}; \min_{k\neq j} \bar{c}^{(\lambda)}_{i_s k} + \min_{l\neq i} \bar{c}^{(\lambda)}_{l j_s}\right)$$

where indices $i_r$ and $j_r$ ($i_s$ and $j_s$) are such that in solution $X^r$ ($X^s$) we have

$$x_{i_r j} = x_{i j_r} = 1 \ (x_{i_s j} = x_{i j_s} = 1)$$

Effectively, to reoptimize problem $P_\lambda$ with $x_{ij} = 1$, in regard with its optimal solution $X^r$ ($X^s$), it is necessary to determine—at least—a new assignment in the line $i_r$ ($i_s$) and a new one in the column $j_r$ ($j_s$)(see figure 3).

- If these two assignments are identical, i.e. $i_r j_r$ ($i_s j_s$), the corresponding cost is $\bar{c}^{(\lambda)}_{i_r j_r}$ ($\bar{c}^{(\lambda)}_{i_s j_s}$)
- If there are different, i.e. $i_r k$ and $l j_r$ ($i_s k$ and $l j_s$), the corresponding cost is $\bar{c}^{(\lambda)}_{i_r k} + \bar{c}^{(\lambda)}_{l j_r}$ ($\bar{c}^{(\lambda)}_{i_s k} + \bar{c}^{(\lambda)}_{l j_s}$)

But clearly, to be inside the triangle $\triangle Z_r Z_s$, we must have (see figure 4)

$$\tilde{z}_\lambda + l_{ij} < \lambda_1 z_{1s} + \lambda_2 z_{2r}$$

Consequently, we obtain the following fathoming test:

*Test 1*

$x_{ij} \in L$ can be eliminated if $\tilde{z}_\lambda + l_{ij} \geq \lambda_1 z_{1s} + \lambda_2 z_{2r}$ or equivalently if $l_{ij} \geq \lambda_1 \lambda_2$

*Remark 1.* Even if the real increasing of $\tilde{z}_\lambda$ is smaller than $\lambda_1 \lambda_2$, the corresponding optimal solution of $P_\lambda$ with $x_{ij} = 1$, can be non efficient because its image is located out of the triangle $\triangle Z_r Z_s$ in the objective space (see figure 5).

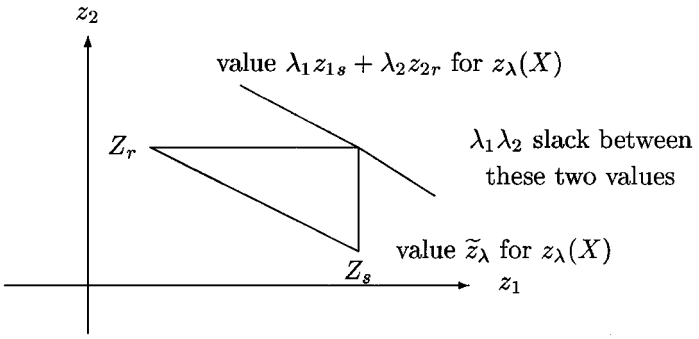


*Figure 4*. Test 1.

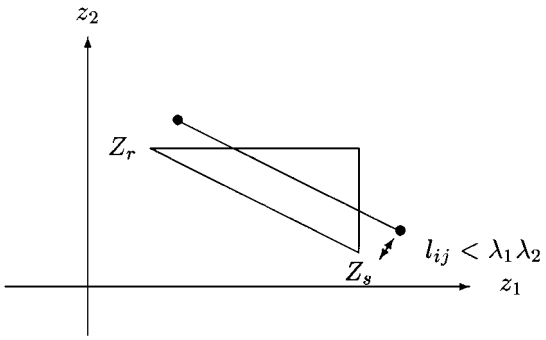


*Figure 5*. Dominated point with $l_{ij} < \lambda_1\lambda_2$.

To avoid such situation, two additional tests can be introduced to try to eliminate the corresponding solution $x_{ij} \in L$; they are build in a similar manner but with a lower bound of the increasing of $z_1(X^r)$ ($z_2(X^s)$ if $x_{ij} = 1$. Unfortunately, these tests can only be efficient if $z_{1r}$ ($z_{2s}$) is close to the optimal value of $z_1$ ($z_2$), because otherwise many reduced costs of $z_1(X^r)$ ($z_2(X^s)$) are strictly negative.

So in this first step, the lower bound $l_{ij}$ is determined for all $x_{ij} \in L$; the list is ordered by increasing values of $l_{ij}$.

Only the variables not eliminated by the test 1 are kept. Problem $P_\lambda$ is reoptimized successively for each non eliminated variable; let us note that only one iteration of the Hungarian method is needed. After the optimization,the solution is eliminated if its image in the objective space is located outside the triangle $\triangle Z_r Z_s$. Otherwise, a non dominated solution is obtained and put in a list $NS_{rs}$; at this time, the second step is applied.

***2.2.2. Second step.*** When non dominated points $Z_1, \ldots, Z_m \in NS_{rs}$ are found inside the triangle $\triangle Z_r Z_s$, the test 1 can be improved. Effectively (see figure 6), in this test the value

$$\lambda_1 z_{1s} + \lambda_2 z_{2r}$$

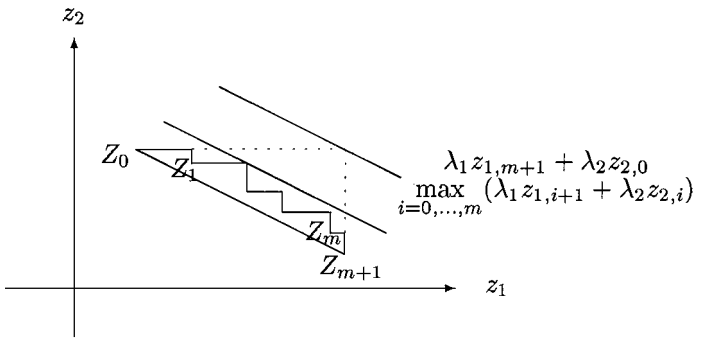


*Figure 6.* Test 2.

can be replaced by the lower value

$$\max_{i=o,\ldots,m} (\lambda_1 z_{1,i+1} + \lambda_2 z_{2,i})$$

where $Z_o \equiv Z_r$ and $Z_{m+1} \equiv Z_s$

The new value corresponds to an updated upper bound of $z_\lambda(X)$ for non dominated points. With the new test

*Test 2*

$x_{ij} \in L$ can be eliminated if $\tilde{z}_\lambda + l_{ij} \geq \max_{i=o,\ldots,m} (\lambda_1 z_{1,i+1} + \lambda_2 z_{2,i})$

more variables of $L$ can be eliminated. Each time a new non dominated point is obtained, the list $NS_{rs}$ and the test 2 are updated. The procedure stops when all the $x_{ij} \in L$ have been either eliminated or analyzed. At this moment the list $NS_{rs}$ contains the non supported solutions corresponding to the triangle $\triangle Z_r Z_s$.

*Remark 2.* Each time the Hungarian method is applied, either in the first phase or in the second phase, it is necessary to obtain all optimal solutions and not only one of them.

This is not an obvious task. An enumeration procedure is needed to examine all feasible combinations of zero reduced costs if the number of such costs is larger than $n$. If many zero reduced costs exist, this enumeration procedure is time consuming.

## 3. The MOSA method

The MOSA method is an adaptation of the S.A. heuristic procedure to a multi-objective framework. Its aim is to generate a good approximation $\widehat{E(P)}$ of $E(P)$ and the procedure is valid for any number $K \geq 2$ of objectives. Similarly to a single objective heuristic in which

a potentially optimal solution emerges, in the MOSA method the set $\widehat{E(P)}$ will contain potentially efficient solutions.

In Section 3.1 the MOSA method will be reminded (for more details see Ulungu et al., 1998); an improvement of the method will be described in Section 3.2; the way to measure the performance of MOSA, by comparison between $E(P)$ and $\widehat{E(P)}$, is given in Section 3.3.

### *3.1. The basic method*

#### ***3.1.1. Preliminaries***

- A wide diversified set of weights is considered: different weight vectors $\lambda^{(l)}$, $l \in L$ are generated where $\lambda^{(l)} = (\lambda_k^{(l)},\ k = 1, \ldots, K)$ with $\lambda_k^{(l)} \geq 0\ \forall k$ and $\sum_{k=1}^{K} \lambda_k^{(l)} = 1, \forall l \in L$. This set of weights is uniformly generated.
- A scalarizing function $s(z, \lambda)$ is chosen. As specified in Ulungu et al. (1998), the effect of this choice on the procedure is small due to the stochastic character of the method. The weighted sum is very well known and it is the easiest scalarizing function:

$$s(z, \lambda) = \sum_{k=1}^{K} \lambda_k z_k$$

- The three classic parameters of a S.A. procedure are initialized

  $T_0$: initial temperature (or alternatively an initial acceptance probability $P_0$);
  $\alpha(<1)$: the cooling factor;
  $N_{\text{step}}$: the length of temperature step in the cooling schedule;

  and the two stopping criteria are fixed

  $T_{\text{stop}}$: the final temperature;
  $N_{\text{stop}}$: the maximum number of iterations without improvement

  (see Pirlot (1996) for more details).
- A neighborhood $V(X)$ of feasible solutions in the vicinity of $X$ is defined. This definition is problem dependent. It is particularly easy to define $V(X)$ in the case of the Assignment problem:

  if $X$ is characterized by $x_{ij_i} = 1 \quad i = 1, \ldots, n$
  $V(X)$ contains all the solutions $Y$ verifying
  $y_{ij_i} = 1 \quad i \in \{1, \ldots, n\} \backslash \{a, b\}$
  $y_{aj_b} = y_{bj_a} = 1$
  where $a, b$ are chosen randomly in $\{1, \ldots, n\}$.

***3.1.2. Determination of PE($\lambda^{(l)}$), $l = 1, \ldots, L$.*** For each $l \in L$ the following procedure is applied to determine a list $PE(\lambda^{(l)})$ of potentially efficient solutions.

**a) initialization**

- Draw at random an initial solution $X_0$.

- Evaluate $z_k(X_0)\ \forall k$.
- $PE(\lambda^{(l)}) = \{X_0\}$; $N_{\text{count}} = m = 0$.

**b) iteration** $m$

- Draw at random a solution $Y \in V(X_m)$
- evaluate $z_k(Y)$ and determine $\Delta z_k = z_k(Y) - z_k(X_m)\ \forall k$
- Calculate $\Delta s = s(z(Y), \lambda) - s(z(X_m), \lambda)$
  If $\Delta s \leq 0$, we accept the new solution:

$$X_{m+1} \longleftarrow Y \quad N_{\text{count}} = 0$$

Else we accept the new solution with a certain probability $p = \exp(-\frac{\Delta s}{T_m})$:

$$X_{m+1} \begin{cases} \overset{p}{\longleftarrow} Y & N_{\text{count}} = 0 \\ \overset{1-p}{\longleftarrow} X_m & N_{\text{count}} = N_{\text{count}} + 1 \end{cases}$$

- If necessary, update the list $PE(\lambda^{(l)})$ in regard with solution $Y$.
- $m \leftarrow m + 1$

| | | | |
|---|---|---|---|
| – If $m(\text{mod } N_{\text{step}}) = 0$ | then | $T_m = \alpha T_{m-1}$; | |
| | else | $T_m = T_{m-1}$. | |
| – If $N_{\text{count}} = N_{\text{stop}}$ | or | $T < T_{\text{stop}}$ | then stop; |
| | | | else iterate. |

***3.1.3. Generation of $\widehat{E(P)}$.*** Because of the use of a scalarizing function, a given set of weight $\lambda^{(l)}$ induces a privileged direction on the efficient frontier. The procedure generates only a good subset of potentially efficient solutions in that direction. Nevertheless, it is possible to obtain solutions which are not in this direction, because of the large exploration of the set of feasible solutions at high temperature i.e. at the beginning of the S.A. procedure (Pirlot, 1996); these solutions are often dominated by some solutions generated with other weight sets.

To obtain a good approximation $\widehat{E(P)}$ to $E(P)$ it is thus necessary to filter the set $\bigcup_{l=1}^{|L|} PE(\lambda^{(l)})$ by pairwise comparisons to remove the dominated solutions. This filtering procedure is denoted by $\bigwedge$ such that

$$\widehat{E(P)} = \bigwedge_{l=1}^{|L|} PE(\lambda^{(l)})$$

A great number of experiments is required to determine the number $L$ of set of weights sufficient to give a good approximation of the whole efficient frontier.

### 3.2. *An improvement of the MOSA method*

The MOSA method is modified by initialization of the method with a greedy approach. At each different weight vectors $\lambda^{(l)}$, $l \in L$ the S.A. optimizes the scalarizing function $s(z, \lambda)$ and chooses a random feasible assignment as starting point. In this new version of MOSA, a greedy step is considered to produce the starting point. The greedy step is defined as follow:

- Initialization

$N = \{1, \ldots, n\}$
$R = \emptyset$ indices of rows assigned
$C = \emptyset$ indices of columns assigned
$it = 0$

- Iteration *it*

$$C_{i^* j^*}^{(\lambda)} = \min_{\substack{i \in N \setminus R \\ j \in N \setminus C}} \left\{ C_{ij}^{(\lambda)} = \sum_{k=1}^{K} \lambda_k C_{ij}^k,\ i, j = 1, \ldots, n \right\}$$

$R = R \cup \{i^*\}$;
$C = C \cup \{j^*\}$;
$it = it + 1$ (a new assignment is defined);
if $(it = n)$ stop;
else iterate.

The performance of this improvement of the MOSA method is discussed in Section 4.2.

### 3.3. *The measures of performance of the MOSA method*

To measure whether $\widehat{E(P)}$ is a good approximation of $E(P)$ we can compare them for instances of problem $(P)$ whose $E(P)$ can be exactly generated by the method described in Section 2.

A first obvious measure of the quality of $\widehat{E(P)}$ is the proportion of efficient solutions it contains:

$$M_1 = \frac{|\widehat{E(P)} \bigcap E(P)|}{|E(P)|}$$

But this is a very "rough" measure because a multi-objective heuristic doesn't necessarily generate the whole efficient set $E(P)$.

As the different triangles $\triangle Z_r Z_s$ (see Section 2) are existence zones for efficient solutions, another possible measure is the proportion of solutions in $\widehat{E(P)}$ which at least are in those

triangles:

$$M_2 = \frac{|T(\widehat{E(P)})|}{|\widehat{E(P)}|}$$

where $T(\widehat{E(P)})$ is the subset of $\widehat{E(P)}$ whose solutions are included in the existence zones for efficient solutions.

Nevertheless, a solution belonging to $\widehat{E(P)} \setminus T(\widehat{E(P)})$ is not necessarily a bad approximation of an efficient solution. So, it is interesting to define a global distance between $\widehat{E(P)}$ and $E(P)$ (see Czyzak and Jaszkiweicz, 1998 and Ulungu et al., 1998). If

$$d(X, Y) = \sum_{k=1}^{K} w_k |z_k(X) - z_k(Y)|$$

where $w_k$ is a weight to take into account the variation range of criterion $z_k$ and

$$d(\widehat{E}, Y) = \min_{X \in \widehat{E(P)}} d(X, Y)$$

the distance between $Y \in E(P)$ and the closest solution in $\widehat{E(P)}$.
We can consider

- an average distance between $\widehat{E(P)}$ and $E(P)$

$$D_1(\widehat{E}, E) = \frac{1}{|E(P)|} \sum_{Y \in E(P)} d(\widehat{E}, Y)$$

- a worst case distance between $\widehat{E(P)}$ and $E(P)$

$$D_2(\widehat{E}, E) = \max_{Y \in E(P)} d(\widehat{E}, Y)$$

- a measure of the uniformity of quality of $\widehat{E(P)}$

$$\frac{D_2(\widehat{E}, E)}{D_1(\widehat{E}, E)}$$

These three measures of quality of $\widehat{E(P)}$ will be used in the following section to evaluate the performance of the MOSA method for the bi-objective Assignment problem.

*Table 1.* Results of the two phase method.

| $n$ | $E(P)$ | $SE(P)$ | $NSE(P)$ | cputime |
|---|---|---|---|---|
| 5 | 8 | 3 | 5 | 5 |
| 10 | 16 | 6 | 10 | 10 |
| 15 | 39 | 12 | 27 | 14 |
| 20 | 54 | 13 | 41 | 61 |
| 25 | 71 | 25 | 46 | 102 |
| 30 | 88 | 26 | 62 | 183 |
| 35 | 82 | 27 | 55 | 384 |
| 40 | 126 | 52 | 74 | 1203 |
| 45 | 113 | 41 | 72 | 3120 |
| 50 | 156 | 57 | 99 | 3622 |

## 4. Experimental results

### *4.1. The two phase method*

We have generated 10 problems of dimension varying from $n = 5$ till $n = 50$ by steps of 5. For the instances considered here, the inputs (two square matrices of size $n \times n$) are randomly chosen in $[0, 30]$.

All the tests have been made on a DEC 3000 ALPHA station and the results are reported in Table 1. The cputime is measured in seconds.

The table gives the number of supported and non supported efficient solutions for each dimension $n$. The increasing of the number of supported solutions and of the number of non supported solutions are approximatively the same. We note that this fact is different for the bi-objective knapsack problem in which the number of non supported solutions increases faster compared with the number of supported efficient solutions (see Visée et al., 1998). We also observe that the cputime used by the method increases exponentially with the size of the problem.

### *4.2. The MOSA method*

The parameters of the MOSA method have been fixed to the following values $P_0 = 0.5$, $\alpha = 0.975$, $N_{\text{step}} = 10.000$, $T_{\text{stop}} = 1e - 04$, $N_{\text{step}} = 20.000$, and $L = 25$. The numerical results are reported in Table 2 (for the MOSA method) and in Table 3 (for the improved MOSA method).

From these numerical results, we can observe that:

- as expected, we obtain good values for $M_1$ for problems with small dimensions, but $M_1$ decreases rapidly to zero for larger problems ($n \geq 20$).
- the second version of the MOSA method is only a improvement for not too small problems ($n \geq 20$).

*Table 2*. Results of the MOSA method.

| $n$ | $\lvert E(P)\rvert$ | $\lvert\widehat{E(P)}\rvert$ | $M_1$ | $M_2$ | $D_1$ | $D_2$ | $D_2/D_1$ | cputime |
|---|---|---|---|---|---|---|---|---|
| 5 | 8 | 8 | 100 | 100 | 0.0 | 0.0 | – | 5 |
| 10 | 16 | 14 | 81.2 | 92.8 | 0.016 | 0.117 | 6.90 | 80 |
| 15 | 39 | 29 | 10.2 | 41.3 | 0.046 | 0.184 | 3.91 | 94 |
| 20 | 54 | 36 | 0.0 | 36.1 | 0.042 | 0.137 | 3.20 | 54 |
| 25 | 71 | 47 | 0.0 | 8.5 | 0.060 | 0.112 | 1.85 | 74 |
| 30 | 88 | 39 | 0.0 | 10.2 | 0.086 | 0.270 | 3.13 | 171 |
| 35 | 82 | 57 | 0.0 | 1.7 | 0.089 | 0.178 | 1.98 | 184 |
| 40 | 126 | 65 | 0.0 | 0.0 | 0.080 | 0.143 | 1.79 | 144 |
| 45 | 113 | 69 | 0.0 | 0.0 | 0.093 | 0.153 | 1.64 | 298 |
| 50 | 156 | 106 | 0.0 | 0.0 | 0.115 | 0.194 | 1.68 | 290 |

*Table 3*. Results of the improved MOSA method.

| $n$ | $\lvert E(P)\rvert$ | $\lvert\widehat{E(P)}\rvert$ | $M_1$ | $M_2$ | $D_1$ | $D_2$ | $D_2/D_1$ | cputime |
|---|---|---|---|---|---|---|---|---|
| 5 | 8 | 7 | 87.5 | 100 | 0.018 | 0.148 | 0.8 | 5 |
| 10 | 16 | 12 | 56.2 | 91.6 | 0.041 | 0.151 | 3.64 | 75 |
| 15 | 39 | 28 | 25.6 | 67.8 | 0.028 | 0.169 | 5.92 | 97 |
| 20 | 54 | 38 | 3.7 | 36.8 | 0.044 | 0.143 | 3.19 | 63 |
| 25 | 71 | 40 | 0.0 | 12.5 | 0.048 | 0.142 | 2.96 | 106 |
| 30 | 88 | 42 | 3.4 | 42.8 | 0.052 | 0.280 | 5.48 | 226 |
| 35 | 82 | 47 | 0.0 | 17.0 | 0.035 | 0.205 | 5.79 | 186 |
| 40 | 126 | 69 | 0.0 | 0.0 | 0.051 | 0.126 | 2.46 | 139 |
| 45 | 113 | 61 | 0.0 | 4.9 | 0.039 | 0.152 | 3.83 | 306 |
| 50 | 156 | 96 | 0.0 | 0.0 | 0.044 | 0.142 | 3.22 | 246 |

- the number of solutions in $\lvert\widehat{E(P)}\rvert$ is not a good indicator of the performance of a method because these solutions are generally not efficient. For instance for $n = 45$, the MOSA method generates 69 solutions and the improved MOSA method only 61; but these latest solutions form a better approximation of $E(P)$ as indicated by the measures $M_2$ and $D_1$.
- the index $M_2$ indicates that the approximation is certainly not bad for problems with medium dimensions because a relatively large number of generated solutions is in the potential zones of efficient solutions.

  The area of these zones are decreasing with the number of variables; for large scale problems, a large number of supported efficient solutions are very close and thus, these areas are really small so that $M_2$ normally becomes small and decreases to zero.
- the numerical experiments show that the values of $D_1$ (average distance), $D_2$ (worst distance) and $D_2/D_1$ (uniformity) are good and quite stable with the size of problem $n$.
- the same set of parameters has been used to test all the problems. The number of weights $L = 25$ and the length of the temperature step $N_{\text{step}} = 10.000$, are not the best choice

for the small problems and this explains the higher cputimes for the MOSA method for the small problems compared to the cputimes of the exact method.

For larger problems ($n \geq 25$), the cputime for MOSA method becomes attractive with respect to the cputime of the exact method. Better approximations of the efficient set could be obtained in these cases, by increasing the parameters $L$, $N_{\text{step}}$, and $N_{\text{stop}}$.

- the behaviour of the improved MOSA method is especially noticeable for large scale problems ($n \geq 25$). The improved method gives smaller values for $D_1$, but not always smaller values for $D_2$. Moreover when $D_2$ is improved, the improvement is proportionnally smaller for $D_2$ that for $D_1$. This explains that the ratio $D_2/D_1$ is higher for the improved MOSA method, which induces a non relative uniformity of the approximation.

  In the case of small problems, the starting points are to close the efficient frontier and in this situation, some non supported efficient solutions become more difficult to reach.
- To show the generality of the MOSA method (which can be applied to many other MOCO problems) we don't use any information furnished by the exact method. Nevertheless, when an exact method is available to at least generate easily $SE(P)$ or the extreme solutions of $SE(P)$ (like for the bi-objective assignment problem) it is possible to initialize the MOSA method with this set of solutions. Of course then, the performances will be improved and in particular the measure $M_1$ will be strictly positive.

## 5. Conclusions

In this paper, we present an exact method and the so-called MOSA method (Multi-Objective Simulated Annealing) for the resolution of the linear Assignment problem with two objectives.

Our method differs of the Czyzak's method (Czyzak and Jaszkiewicz, 1998) in which the weights $\lambda$ change at each iteration of the S.A. prodedure. Even if such procedure provides a large diversification, in our opinion there is a risk to not be close enough of the efficient frontier due to a lack of intensification. Nevertheless, it will be interesting to compare both approaches.

The MOSA method remains valid for a larger number of objectives and for large scale problems. In this paper, we focus on bi-objective problems, because we are in that case able to determine the exact efficient set and to compare it to approximation set provided by the MOSA method. Two different measures are designed to evaluate the proximity and the uniformity of approximation set with respect to the exact efficient set. Numerical tests are presented and discussed, showing that MOSA method provides a good approximation of the efficient set and that the results are stable with respect to the size of the problem.

In comparison with the bi-objective Knapsack problem (see Visée et al., 1998), the bi-objective linear Assignment problem generates a smaller number of efficient solutions for a same number of variables of the problem.

More research is planned in this field and in particular it will be of interest to extend and diversify the set of test problems, to study some other neighbourhood structures and to experiment the MOSA method on linear Assignment problem with more than two objectives.

## References

Ben Abdelaziz, F., J. Chaouachi, and S. Krichen (1997). "A Hybrid Heuristic for Multiobjective Knapsack Problems." Technical Report, Institut Supérieur de Gestion, Tunisie.

Czyzak, P., M. Hapke, and A. Jaszkiewicz (1994). "Application of the Pareto-Simulated Annealing to the Multiple Criteria Shortest Path Problem." Technical Report, Politechnika Poznanska Instytut Informatyki.

Czyzak, P. and A. Jaszkiewicz (1998). "Pareto Simulated Annealing–A Metaheuristic Technique for Multiple Objective Combinatorial Optimization." *Journal of Multi-Criteria Decision Analysis* 7, 34–47.

Gandibleux, X., N. Mezdaoui, and A. Fréville (1996). "Multiobjective Tabu Search Procedure to Solve Combinatorial Optimisation Problems." Working Paper, LAMIH-LIMAV, Université de Valenciennes.

Hansen, P. (1997). "Tabu Search for Multiobjective Optimization: MOTS." Technical Report, Institute of Mathematical Modelling, Technical University of Denmark.

Pirlot, M. (1996). "General Local Search Methods." *EJOR* 92, 493–511.

Serafini, P. (1992). "Simulated Annealing for Multiple Objective Optimization Problems." In *Proceedings of the Tenth International Conference on Multiple Criteria Decision Making*, Taipei, July 1992. vol. 1, pp. 87–96.

Teghem, J. (1996). "Programmation linéaire." *Editions de l'Université de Bruxelles et Ellipses.*

Ulungu, E.L. (1993). "Optimisation combinatoire multicritére: détermination de l'ensemble des solutions efficaces et méthodes interactives." Thèse de doctorat, Université de Mons-Hainaut (Belgique).

Ulungu, E.L. and J. Teghem (1994). "Multi-Objective Combinatorial Optimization Problems: A Survey." *Journal of Multiple–Criteria Decision Analysis* 3/2, 83–104.

Ulungu, E.L. and J. Teghem (1995). "The Two Phases Method: An Efficient Procedure to Solve Bi-Objective Combinatorial Optimization Problems." *Journal Foundations of Computing & Decision Sciences* 20(2), 149–165.

Ulungu, E.L., J. Teghem, and Ph. Fortemps (1995). "Heuristics for Multi-Objective Combinatorial Optimization by Simulated Annealing." In *Multiple Criteria Decision Making: Theory and Applications. Proceeding of the 6th National Conference on Multiple Criteria Decision Making*, Beijing, China, Aug. 1995, pp. 228–238.

Ulungu, E.L., J. Teghem, Ph. Fortemps, and D. Tuyttens (1999). "MOSA Method: A Tool for Solving MOCO Problems." *Journal of Multi-Criteria Decision Analysis* 8, 221–236.

Ulungu, E.L., J. Teghem, and Ch. Ost (1998). "Efficiency of Interactive Multi-Objective Simulated Annealing Through a Case Study." *Journal of the Operational Research Society* 49, 1044–1050.

Visée, M., J. Teghem, M. Pirlot, and E.L. Ulungu (1998). "Two-Phases Method and Branch and Bound Procedures to Solve the Bi-Objective Knapsack Problem." *Journal of Global Optimization* 12, 139–155.