

## OPTIMIZING A LINEAR FUNCTION OVER THE EFFICIENT SET OF A MULTI-OBJECTIVE TRANSPORTATION PROBLEM

Zineb Mezali<sup>1\*</sup> and Mohamed El-Amine Chergui<sup>2</sup>

Received March 8, 25 / Accepted August 4, 25

**ABSTRACT.** In this study, we propose an exact method for optimizing a linear function over the efficient set of a multi-objective transportation problem (MOTP). This type of problem arises when a decision maker needs to optimize a preference function across numerous efficient solutions. We developed a cutting plane method that iteratively adds efficient cuts to the relaxation of the original problem until a feasible and optimal solution is achieved. The method is enhanced by a lower bound, enabling a more efficient search for the optimal solution.

**Keywords:** transportation problem, multi-objective optimization, optimal efficient solution.

### 1 INTRODUCTION

The transportation problem is a fundamental optimization problem with widespread applications in logistics, supply chain management, and resource distribution. While the traditional transportation problem focuses on minimizing costs or maximizing efficiency under a single objective, real-world scenarios often involve multiple, conflicting objectives, such as minimizing cost while maximizing delivery speed or minimizing environmental impact. This gives rise to the multi-objective transportation problem (MOTP), where the goal is to identify a set of efficient (Pareto-optimal) solutions.

Different solution approaches have been developed for the MOTP. Some methods are exact, aiming to identify all efficient solutions, including those based on linear programming (e.g., Srinivasan & Thompson (1976), Diaz (1978), Aneja & Nair (1979), Isermann (1979), Shi (1995), Abd Elazeem et al. (2021)) and dynamic programming (e.g., Gallagher & Saleh (1994), Abd Elazeem et al. (2021)). Others are designed to find a single efficient solution or a compromise solution, such as goal programming (e.g., Lee & Moore (1973), Sutcliffe et al. (1984),

---

\*Corresponding author

<sup>1</sup>USTHB, RECITS Laboratory, Faculty of Mathematics, Bab Ezzouar, PO 32, 16111 Algiers, Algeria – E-mail: meزالizineb@gmail.com – <https://orcid.org/0000-0002-1285-1035>

<sup>2</sup>USTHB, RECITS Laboratory, Faculty of Mathematics, Bab Ezzouar, PO 32, 16111 Algiers, Algeria – E-mail: mchergui@usthb.dz – <https://orcid.org/0000-0002-2269-9735>

Anukokilaa et al. (2017), Nomani et al. (2017)), interactive methods (e.g., Ringuest & Rinks (1987)), lexicographic optimization (e.g., Stewart & Ittmann (1979), Mardanya et al. (2022)), the minimize distance method (e.g., Alardhi et al. (2022), Kaur et al. (2023) ), and the decomposition approach (e.g., Al-Rajhi et al. (2013), Alkhulaifi et al. (2014)). In addition, there are heuristics that produce an approximation of the Pareto front, as seen in the work of Gen et al. (1999), Mousa (2010), Zaki et al. (2012), Zheng & Chen (2013), Lemarchand et al. (2018), and Wu et al. (2022)).

A key challenge in this context is optimizing a linear function over the efficient set of a MOTP. This problem involves finding the best solution from the set of Pareto-optimal solutions without explicitly enumerating all of them.

The problem of optimizing a linear function over an efficient set involves navigating a potentially large and non-convex solution space to identify the optimal solution. This makes it a global optimization problem, as highlighted by Benson (1991) and Yamamoto (2002).

Various methods have been proposed to solve this problem for general multi-objective optimization problems. Song (2000) proposed a weighted sum method, Abbas & Chaabane (2006) introduced a method to optimize a linear function over the integer Pareto-optimal set without enumerating all efficient solutions. Their approach uses a selection technique to iteratively improve the objective value and performs two types of cuts to narrow down the feasible region until the optimal solution is found. Jorge (2009) presented an exact algorithm that optimizes a linear function over the efficient set of a multi-objective integer linear programming problem (*MOILP*) by defining a sequence of progressively more constrained single-objective integer problems, thereby eliminating undesirable solutions. Similarly, Boland et al. (2017) proposed an algorithm for optimizing a linear function over the efficient solutions of a *MOILP*, which relies on an efficient method for enumerating non-dominated points and employs a novel criteria space decomposition scheme. A branch and bound technique is also suggested for finding an optimal solution by Ouail et al. (2017), while Drici et al. (2018) combined a branch and cut technique with continuous linear fractional programming to obtain an integer optimal solution without explicitly listing all efficient solutions. Primal and dual algorithms are proposed to solve this problem in the criteria space by Zhengliang & Ehrgott (2018) and Zerfa & Chergui (2024). Additionally, Belkhiri et al. (2022) employed a branch and bound technique to find the most preferred efficient extreme point, based on a new linear criterion, among all efficient vertices of a multi-objective linear programming problem. Sharma et al. (2023) used ranked solutions of an integer linear programming problem to optimize a linear function over the efficient set of a *MOILP*. For combinatorial optimization problems, Chaabane & Lachemi (2023) developed a new exact algorithm based on dynamic programming, specifically designed to optimize a linear function over the efficient set of the binary bi-objective knapsack problem.

While these methods are effective for general multi-objective optimization problems, they are not specifically designed to handle the unique structure of transportation problems leading to excessive computational effort. This paper addresses this gap by proposing a specialized algorithm for optimizing a linear function over the efficient set of a multi-objective transportation problem.

We propose a cutting plane method that avoids exhaustive enumeration of efficient solutions by iteratively refining the search space and ensuring improvement in the objective function.

The remainder of this paper is organized as follows: Section 2 introduces the definitions and mathematical notations used throughout the paper. Section 3 presents the proposed resolution method, including its key theoretical foundations, algorithmic steps, and an illustrative example. Section 4 describes the experimental setup and computational results. Finally, Section 5 concludes the paper.

## 2 PROBLEM FORMULATION AND DEFINITIONS

We consider the following Mutli-Objective Transportation Problem (MOTP):

$$\text{Minimize } \sum_{i \in I} \sum_{j \in J} c_{ij}^k x_{ij}, \quad \forall k \in K \quad (1)$$

$$\text{Subject to } \sum_{j \in J} x_{ij} = a_i, \quad \forall i \in I \quad (2)$$

$$\sum_{i \in I} x_{ij} = d_j, \quad \forall j \in J \quad (3)$$

$$x_{ij} \geq 0, \text{ integers}, \quad \forall i \in I, j \in J \quad (4)$$

Let  $I = \{1, \dots, m\}$ ,  $J = \{1, \dots, n\}$ , and  $K = \{1, \dots, r\}$  represent the sets of sources, destinations, and criteria, respectively. The cost matrix  $c^k$  for each criterion  $k$  is defined as  $c^k = (c_{ij}^k)$  for all  $i \in I$  and  $j \in J$ . Here,  $a_i$  ( $i = 1, 2, \dots, m$ ) and  $d_j$  ( $j = 1, 2, \dots, n$ ) denote the availability at the  $i$ -th source and the demand at the  $j$ -th destination, respectively. We assume that the coefficients  $a_i$  and  $d_j$ , take non-negative integer values. Additionally, the total supply must equal the total demand ( $\sum_{i \in I} a_i = \sum_{j \in J} d_j$ ), which is necessary and sufficient for a transportation problem to have a feasible solution.

In what follows, we use the following vector notations.

For each criterion  $k \in K$ , we define a row vector of size  $1 \times (m \times n)$  denoted by  $c^k$ :

$$c^k = (c_{11}^k, c_{12}^k, \dots, c_{ij}^k, \dots, c_{mn}^k), \quad \text{where } i \in I \text{ and } j \in J$$

The cost matrix  $C$  is an  $r \times (m \times n)$  matrix defined as:

$$C = \begin{pmatrix} c^1 \\ c^2 \\ \vdots \\ c^k \\ \vdots \\ c^r \end{pmatrix}$$

$X$  and  $b$  are column vectors of size  $(m \times n) \times 1$ , denoted respectively by:  $X = (x_{11}, x_{12}, \dots, x_{ij}, \dots, x_{mn})^T$ , and  $b = (a_1, a_2, \dots, a_m, d_1, d_2, \dots, d_n)^T$ .

The mathematical program associated to MOTP can be written as follows:

$$\begin{cases} \min CX \\ AX = b \\ X \geq 0, \text{ integer} \end{cases} \quad (5)$$

Let  $D = \{X \in \mathbb{Z}^{mn} \mid AX = b, X \geq 0\}$  be the feasible solutions set of MOTP, assumed to be a nonempty compact polyhedron from  $\mathbb{Z}^{mn}$ , and  $Z(D)$  its image set in the criteria space  $\mathbb{R}^r$ .

**Definition 1.** A basic feasible solution of  $D$  is an extreme point of  $D$ .

**Definition 2.** A basic feasible solution  $X \in D$ , is efficient if there is no other basic feasible solution  $Y \in D$  such that:  $CY \leq CX$  and  $CY \neq CX$ .

**Definition 3.** A basic feasible solution which is efficient is called an efficient extreme point.

**Definition 4.** The image of an efficient solution in the criteria space is called a non-dominated point.

**Definition 5.** An efficient cut for MOTP is a cut that removes integer solutions but not efficient integer solutions.

We aim to solve the problem of optimizing a linear function, over the efficient solutions set of the MOTP, consisting in a new Transportation Problem, defined as follows:

$$(O) \begin{cases} \min \sum_{i \in I} \sum_{j \in J} f_{ij} x_{ij} \\ X \in EMOTP \end{cases}$$

where  $(f_{ij}) \forall i \in I, \forall j \in J$ , is the cost matrix of the new Transportation Problem, and  $(EMOTP)$  denotes the efficient solutions set of MOTP defined as:

$$EMOTP = \{X \in D \mid \nexists Y \in D \text{ such that } CY \prec CX\}$$

where  $CY \prec CX$  indicates that  $CY$  dominates  $CX$ .

Using the vector notations, the program  $(O)$  can be written as follows:

$$(O) \begin{cases} \min FX \\ X \in EMOTP \end{cases} \quad (6)$$

where  $F = (f_{11}, f_{12}, \dots, f_{ij}, \dots, f_{mn})$  is the cost vector.

Let  $\hat{F} = (\hat{f}_{11}, \hat{f}_{12}, \dots, \hat{f}_{ij}, \dots, \hat{f}_{mn})$  be the reduced cost vector relative to  $F$ , where each  $\hat{f}_{ij}$  represents the reduced cost associated with  $x_{ij}$ .

To solve the program  $(O)$ , we define at iteration  $l$  of the algorithm, its following relaxed program:

$$(OR_l) \begin{cases} \min FX \\ X \in D_l \end{cases}$$

where the set  $D_l$  is defined recursively:

- $D_0 = D$ , the initial feasible region.
- For  $l \geq 1$ ,  $D_l = \{X \in D_{l-1} \mid X \text{ satisfies (10) and (11) and (12) (if necessary)}\}$ .

Here,  $D_l$  represents the feasible region at iteration  $l$ , obtained by applying the constraint (10) and the constraints (11) and (12) (if necessary) to the previous feasible region  $D_{l-1}$ . This recursive process ensures that  $D_l$  progressively refines the feasible region by removing integer solutions that are not efficient extreme points, reducing the search space to only the most promising solutions.

Thus, the search for an optimal solution of  $(O)$  is conducted by solving the single-objective constrained integer transportation program  $(OR_l)$ .

### 3 RESOLUTION METHOD

#### 3.1 Overview of the method

The proposed approach determines an optimal solution (denoted as  $X^*$ ) for the program  $(O)$  without the need to exhaustively enumerate all the efficient solutions within the set  $EMOTP$ . The process begins with the resolution of program  $(OR_0)$ . If the obtained solution  $X_0$  is efficient based on the efficiency test (as described in subsection 3.2), the process terminates. Else, we compute an upper bound  $ub$  for the objective function  $FX^*$  as outlined in the OTP-Algorithm (refer to subsection 3.6), and we add the constraint (8) to program  $(OR_l)$ .

Let  $X^l$  an obtained optimal solution for program  $(OR_l)$ . The descent directions from  $X^l$  are determined and the constraint (10) is added to  $(OR_l)$  to obtain a new program  $(OR_{l+1})$ . This last constraint ensures an improvement for criterion  $k$ , which means that we can continue our search for other non-dominated points in the new feasible solutions set  $D_{l+1}$ . The program  $(OR_{l+1})$  is then resolved, and if the resulting solution is not an extreme point for  $D_{l+1}$ , in other words, the obtained solution is feasible but is not a basic solution, then the constraints (11) and (12) are introduced to ensure that the number of positive variables remains less than or equal to  $m+n-1$ , and the program is resolved once again. This iterative process continues until an efficient extreme point is achieved.

#### 3.2 Efficiency test

The following mixed-integer program is solved to determine whether the solution  $X^l$  of program  $(OR_l)$ , obtained at iteration  $l$  of our algorithm, is efficient:

$$(P_{X^l}) \begin{cases} \max e^T V \\ c^k X + V = c^k X^l \quad \forall k \in K \\ X \in D_l \\ V \geq 0 \end{cases} \quad (7)$$

where  $e$  is a vector of ones and  $V \in \mathbb{R}^r$ .

The solution  $X^l$  is efficient if and only if  $(P_{X^l})$  has a maximum value of zero. Otherwise, if the maximum value of  $(P_{X^l})$  is finite and nonzero, then the corresponding solution  $Y^l$  is efficient (see Nguyen (1992)).

### 3.3 Generating an upper bound

Let  $X^*$  be an optimal solution of  $(O)$ , we define the following relation:

$$FX^* \leq ub \quad (8)$$

where:

- $FX^*$  represents the objective value at the optimal solution  $X^*$ .
- $ub$  is an upper bound on the objective value.

To determine an upper bound, we can solve the following single transportation problem, denoted as  $(P)$ :

$$(P) \begin{cases} \min \sum_{i \in I} \sum_{j \in J} f_{ij} x_{ij} + \frac{1}{r} \sum_{k \in K} \sum_{i \in I} \sum_{j \in J} c_{ij}^k x_{ij} \\ \text{subject to} \\ (2), (3), (4) \end{cases}$$

To explore potential improvements in the upper bound of the optimal value of program  $(O)$ , an adjacent efficient solution is generated by solving a sub-problem based on Isermann's efficiency-testing method for identifying efficient non-basic variables in multi-objective linear programs (see Sinha (2005), Chapter 28, p. 477). The sub-problem is formulated as follows:

$$\begin{cases} \min e^T \lambda \\ -W^T \lambda \geq 0 \\ (W^{ij})^T \lambda = 0 \\ \lambda \geq e \end{cases} \quad (9)$$

where:

- $\lambda \in \Lambda = \left\{ \lambda \in \mathbb{R}^r \mid \lambda_k > 0, \sum_{k=1}^r \lambda_k = 1 \right\}$ ,
- $W$  is the reduced cost matrix of the non-basic variables of an efficient solution  $X$ ,
- $W^{ij}$  is a column of  $W$  representing the reduced cost vector of the non-basic variable  $x_{ij}$  of  $X$ ,
- $e$  is a vector of ones.

If an optimal solution  $\lambda^*$  exists for the sub-problem (9), the non-basic variable  $x_{ij}$  is identified as efficient. By introducing  $x_{ij}$  into the basis, an adjacent efficient solution is obtained.

### 3.4 Descent directions, efficient cut and forcing the basic solution

**Theorem 1.** *The coefficient matrix  $A$  has a rank of  $m + n - 1$ . (see Gupta (1992), Chapter 8, p. 294).*

**Corollary 1.** *A basic feasible solution of  $D$  is a feasible solution that contains no more than  $m + n - 1$  positive basic variables.*

Let  $X^l = (x_{11}^l, \dots, x_{ij}^l, \dots, x_{mn}^l)$  be a solution of program  $(O)$  at iteration  $l$  of the algorithm. We define the following:

- $N_l$ : The set of indices corresponding to the non-basic variables of  $X^l$ .
- For each criterion  $k \in K$ , the reduced cost vector  $\hat{c}_l^k$  of  $X^l$  is defined as:

$$\hat{c}_l^k = \left( \hat{c}_{ij}^k \right)_{(i,j) \in N_l},$$

where  $\hat{c}_l^k$  is a row vector of size  $1 \times |N_l|$  and  $\hat{c}_{ij}^k$  is the reduced cost of the non-basic variable  $x_{ij}^l$  for criterion  $k$ .

- The reduced cost matrix  $\hat{C}_l$  is an  $r \times |N_l|$  matrix given by:

$$\hat{C}_l = \begin{pmatrix} \hat{c}_l^1 \\ \hat{c}_l^2 \\ \vdots \\ \hat{c}_l^k \\ \vdots \\ \hat{c}_l^r \end{pmatrix},$$

- $H_l = \{ij \in N_l \mid \hat{c}_{ij}^k < 0\}$ .
- $K_l = \{k \in \{1, \dots, r\} \mid H_l \neq \emptyset\}$ .
- $\lceil \hat{c}_{ij}^k \rceil$ : The ceiling function of  $\hat{c}_{ij}^k$ .

We define the following relations:

$$\sum_{k \in K_l} c^k X \leq \sum_{k \in K_l} c^k X^l + \sum_{k \in K_l} \sum_{ij \in N_l \setminus H_l} \lceil \hat{c}_{ij}^k \rceil x_{ij} + \sum_{k \in K_l} \max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\} \quad (10)$$

$$x_{ij} \leq M y_{ij} \quad (11)$$

$$\sum_{i=1}^m \sum_{j=1}^n y_{ij} \leq m + n - 1 \quad (12)$$

where:

- $y_{ij}$  is a binary variable that takes the value 1 if  $x_{ij}$  is non-zero, and 0 otherwise.
- $M$  is a sufficiently large constant.

The following results are established to support the different steps of the OTP-Algorithm presented in subsection 3.6.

**Theorem 2.** *If  $X$  is an efficient solution obtained at step  $p$  ( $p > l$ ) such that  $CX \neq CX^l$ , then  $X$  satisfies the following set of constraints:*

$$c^k X \leq c^k X^l + \sum_{ij \in N_l \setminus H_l} [\hat{c}_{ij}^k] x_{ij} + \max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\}, \quad \forall k \in K_l. \quad (13)$$

**Proof.** Assume  $X$  is a feasible solution that does not satisfy (13). Then, there exists  $k \in K_l$  such that:

$$c^k X > c^k X^l + \sum_{ij \in N_l \setminus H_l} [\hat{c}_{ij}^k] x_{ij} + \max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\}. \quad (14)$$

On the other hand, from the simplex tableau, we have the following relation:

$$c^k X = c^k X^l + \sum_{ij \in N_l} \hat{c}_{ij}^k x_{ij} \Rightarrow c^k X = c^k X^l + \sum_{ij \in N_l \setminus H_l} \hat{c}_{ij}^k x_{ij} + \sum_{ij \in H_l} \hat{c}_{ij}^k x_{ij}. \quad (15)$$

This leads to:

$$c^k X \leq c^k X^l + \sum_{ij \in N_l \setminus H_l} [\hat{c}_{ij}^k] x_{ij} + \sum_{ij \in H_l} [\hat{c}_{ij}^k] x_{ij}. \quad (16)$$

From (14) and (16), there exists  $k \in K_l$  such that:

$$\sum_{ij \in H_l} [\hat{c}_{ij}^k] x_{ij} > \max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\}. \quad (17)$$

Dividing both sides by

$$\max\left\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\right\}$$

we get:

$$\frac{\sum_{ij \in H_l} [\hat{c}_{ij}^k]}{\max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\}} x_{ij} < 1 \quad (\text{since the denominator is negative}). \quad (18)$$

We distinguish two cases:

**case1:**

Suppose that for a criterion  $k \in K_l$ , we have:  $\max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\} = \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}$ .

By definition,  $\max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}$  is the largest negative reduced cost.

Therefore,  $\frac{\sum_{ij \in H_l} \lceil \hat{c}_{ij}^k \rceil}{\max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}} > 1$  for all  $ij \in H_l$  (as  $|\sum_{ij \in H_l} \lceil \hat{c}_{ij}^k \rceil| > |\max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}|$ ).

Thus,  $x_{ij} = 0$  for all  $ij \in H_l$ .

Using (15), we get:

$$c^k X \geq c^k X^l, \quad \forall k \in K_l \quad (\text{since } c^k X - c^k X^l = \sum_{ij \in N_l \setminus H_l} \hat{c}_{ij}^k x_{ij} \geq 0). \quad (19)$$

But  $CX \neq CX^l$ . Therefore,  $CX$  is dominated by  $CX^l$ , and  $X$  is not an efficient solution.

**case2:**

Suppose that for a criterion  $k \in K_l$ , we have:  $\max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\} = -1$ .

Using (14), we get:

$$c^k X > c^k X^l + \sum_{ij \in N_l \setminus H_l} \lceil \hat{c}_{ij}^k \rceil x_{ij} - 1$$

Which implies that:

$$c^k X \geq c^k X^l$$

But  $CX \neq CX^l$ . Therefore,  $CX$  is dominated by  $CX^l$ , and  $X$  is not an efficient solution. □

**Theorem 3.** *If  $X$  is an efficient solution such that  $CX \neq CX^l$ , then  $X$  satisfies the following constraint, which is the same as relation (10) :*

$$\sum_{k \in K_l} c^k X \leq \sum_{k \in K_l} c^k X^l + \sum_{k \in K_l} \sum_{ij \in N_l \setminus H_l} \lceil \hat{c}_{ij}^k \rceil x_{ij} + \sum_{k \in K_l} \max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\}. \quad (9)$$

**Proof.** Let  $X$  be an efficient solution such that  $CX \neq CX^l$ . According to Theorem 2,  $X$  satisfies the set of constraints given by Equation (13). Adding these inequalities, we obtain:

$$\sum_{k \in K_l} c^k X \leq \sum_{k \in K_l} c^k X^l + \sum_{k \in K_l} \sum_{ij \in N_l \setminus H_l} \lceil \hat{c}_{ij}^k \rceil x_{ij} + \sum_{k \in K_l} \max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\}. \quad (9)$$

Thus,  $X$  satisfies constraint (9). □

**Corollary 2.** *Constraint (9) is an efficient cut.*

**Proof.** Constraint (9) is satisfied by any efficient solution  $X$ , and its addition does not eliminate efficient solutions. Hence, (9) constitutes an efficient cut.  $\square$

**Theorem 4.** *let  $X$  be an efficient solution and let  $\hat{c}_{ij}^k \geq 0$  for all  $k \in K$ , with at least one strict inequality, be the reduced cost of the entering non-basic variable  $x_{ij}$ . Thus, the resulting solution  $X'$  is non-efficient.*

**Proof.** If  $\hat{c}_{ij}^k \geq 0$  for all  $k \in K$ , with at least one strict inequality, then:

$$CX' = CX + \sum_{k \in K} \hat{c}_{ij}^k x_{ij},$$

where  $\sum_{k \in K} \hat{c}_{ij}^k x_{ij} > 0$ . Consequently,  $CX \leq CX'$  holds with at least one strict inequality. This implies that  $CX \prec CX'$ , and thus  $X'$  is non-efficient.  $\square$

### 3.5 Termination condition

**Theorem 5.** *If the current solution  $X^l$  is not efficient and  $H_l = \emptyset$ , then  $D_l$  is an explored domain.*

**Proof.**  $H_l = \emptyset$  means that  $X^l$  is an optimal solution for all the criteria, which means that  $CX^l$  dominates  $CX$  for all  $X \in D_l$ . Thus  $D_l$  contains no efficient solutions, and  $X^*$  obtained by the OTP-Algorithm is the optimal solution for program (O).  $\square$

**Theorem 6.** *The OTP-Algorithm terminates in a finite number of iterations and returns an optimal solution for program (O).*

**Proof.** The domain of feasible solutions  $D$  for the MOTP is a compact polyhedron, containing a finite number of extreme points. Thus, the set  $EMOTP$ , which is a subset of  $D$ , also contains a finite number of extreme points. During each iteration of the OTP-Algorithm, cuts are introduced to eliminate at least the current extreme point. However, as stated in Theorem 3 and Corollary 2, no efficient extreme point is removed. Additionally, solving the programs  $(OR_l)$  and performing the efficiency test guarantee that we find an optimal solution for the program (O), which is also efficient for the MOTP problem.  $\square$

### 3.6 The OTP-Algorithm

The proposed OTP-Algorithm consists of the following steps:

**Input:**  $A, b, C$  and  $F$ .

**Output:**  $X^*$  the optimal solution of (O) and  $F^* = FX^*$ .

**Step 0: Upper Bound** : solve  $(P)$  through the Modified Distribution Stepping-Stone (MODI) method, also known as the method of multipliers (refer to Taha (2013)), and let  $X^P$  be its optimal solution and  $FX^P$  its objective value.

**Step 0.1: Efficiency test**: solve the program (refer to (7)) with  $X^l = X^P$  to test the efficiency of  $X^P$ .

**Step 0.2**: If  $X^P$  is efficient then  $X^{ub} := X^P$  and  $ub := FX^P$ . Else, let  $Y$  be the optimal solution of the program (refer to (7)) which is efficient, and update  $X^{ub}$  and  $ub$  if necessary.

**Step 0.3: Upper Bound Improvement**

- Evaluate the reduced cost  $\hat{F}$  for the solution  $X^{ub}$ .
- Solve the problem (refer to (9)) for all non-basic variables of  $X^{ub}$ .
- Introduce the efficient non-basic variable with the most negative reduced cost into the basis.
  - This generates an adjacent efficient solution with a lower (improved) objective function value  $F$ .
- Update  $X^{ub}$  and  $ub$  if necessary.

**Step 1: Initialization**: set  $l := 0; F^* = +\infty$ ;

**Step 2**: Solve  $(OR_l)$  and let  $X^l$  be the obtained optimal solution .

**Step 3**: If  $X^l$  is efficient via the efficiency test (refer to (7)), set  $X^* := X^l$  and  $F^* := FX^l$  then terminate. Else go to **Step 4**.

**Step 4**: Evaluate reduced cost  $\hat{C}_l$  for the solution  $X^l$ , and determine sets  $N_l$ ,  $H_l$  and  $K_l$ . If  $H_l$  is empty,  $X^* := X^{ub}$  and  $F^* := ub$  then terminate . Else go to **Step 5**.

**Step 5**: If  $l = 0$ , add to  $(OR_l)$  the following constraint:

$$FX \leq ub$$

Else, add the efficient cut:

$$\sum_{k \in K_l} c^k X \leq \sum_{k \in K_l} c^k X^l + \sum_{k \in K_l} \sum_{ij \in N_l \setminus H_l} [\hat{c}_{ij}^k] x_{ij} + \sum_{k \in K_l} \max\{-1, \max_{ij \in H_l} \{\lceil \hat{c}_{ij}^k \rceil\}\}$$

**Step 6**: Set  $l := l + 1$  and resolve  $(OR_l)$  with the added constraints, if the obtained solution is an extreme point then, go to **Step 3**. Else, add the constraints (11) and (12), and go to **Step 2**.

### Illustrative example

Let us consider the following problem of optimizing a linear function over an efficient set defined as:

$$(O) \left\{ \begin{array}{l} \min FX = 5x_{11} + 6x_{12} + 3x_{13} + 4x_{14} + 9x_{21} + 7x_{22} + 7x_{23} + 2x_{24} \\ \quad + 5x_{31} + x_{32} + 2x_{33} + 13x_{34} + 4x_{41} + 5x_{42} + 4x_{43} + 8x_{44} \\ X \in EMOTP \end{array} \right.$$

where  $EMOTP$  is the efficient solutions set of the following  $MOTP$  problem:

$$\left\{ \begin{array}{l} \min Z_1(X) = 14x_{11} + 7x_{12} + 11x_{13} + 12x_{14} + 9x_{21} + 9x_{22} + 13x_{23} + 11x_{24} \\ \quad + 20x_{31} + 2x_{32} + 9x_{33} + 16x_{34} + 7x_{41} + 5x_{42} + 20x_{43} + 8x_{44} \\ \min Z_2(X) = 5x_{11} + 7x_{12} + 11x_{13} + 6x_{14} + 11x_{21} + 7x_{22} + 10x_{23} + 16x_{24} \\ \quad + 11x_{31} + 20x_{32} + 2x_{33} + 13x_{34} + 18x_{41} + 20x_{42} + 16x_{43} + 9x_{44} \\ \min Z_3(X) = 6x_{11} + 6x_{12} + 18x_{13} + 8x_{14} + 20x_{21} + 12x_{22} + 20x_{23} + 2x_{24} \\ \quad + 15x_{31} + 1x_{32} + 7x_{33} + 19x_{34} + 4x_{41} + 16x_{42} + 9x_{43} + 18x_{44} \\ \min Z_4(X) = 16x_{11} + 19x_{12} + 3x_{13} + 4x_{14} + 11x_{21} + 11x_{22} + 7x_{23} + 18x_{24} \\ \quad + 5x_{31} + 1x_{32} + 10x_{33} + 20x_{34} + 8x_{41} + 5x_{42} + 4x_{43} + 16x_{44} \\ \min Z_5(X) = 19x_{11} + 19x_{12} + 18x_{13} + 7x_{14} + 17x_{21} + 15x_{22} + 14x_{23} + 17x_{24} \\ \quad + 10x_{31} + 17x_{32} + 15x_{33} + 13x_{34} + 13x_{41} + 20x_{42} + 10x_{43} + 10x_{44} \\ x_{11} + x_{12} + x_{13} + x_{14} = 9 \\ x_{21} + x_{22} + x_{23} + x_{24} = 12 \\ x_{31} + x_{32} + x_{33} + x_{34} = 11 \\ x_{41} + x_{42} + x_{43} + x_{44} = 15 \\ x_{11} + x_{21} + x_{31} + x_{41} = 2 \\ x_{12} + x_{22} + x_{32} + x_{42} = 25 \\ x_{13} + x_{23} + x_{33} + x_{43} = 3 \\ x_{14} + x_{24} + x_{34} + x_{44} = 17 \\ x_{ij} \geq 0, \text{ integers } \quad \forall i \in I, j \in J \end{array} \right.$$

The relaxed linear program  $(OR_l)$  of  $(O)$  is then written as follows:

$$(OR_l) \left\{ \begin{array}{l} \min FX = 5x_{11} + 6x_{12} + 3x_{13} + 4x_{14} + 9x_{21} + 7x_{22} + 7x_{23} + 2x_{24} \\ \quad + 5x_{31} + x_{32} + 2x_{33} + 13x_{34} + 4x_{41} + 5x_{42} + 4x_{43} + 8x_{44} \\ x_{11} + x_{12} + x_{13} + x_{14} = 9 \\ x_{21} + x_{22} + x_{23} + x_{24} = 12 \\ x_{31} + x_{32} + x_{33} + x_{34} = 11 \\ x_{41} + x_{42} + x_{43} + x_{44} = 15 \\ x_{11} + x_{21} + x_{31} + x_{41} = 2 \\ x_{12} + x_{22} + x_{32} + x_{42} = 25 \\ x_{13} + x_{23} + x_{33} + x_{43} = 3 \\ x_{14} + x_{24} + x_{34} + x_{44} = 17 \\ x_{ij} \geq 0, \text{ integers } \quad \forall i \in I, j \in J \end{array} \right.$$

**Input:**  $A$ ,  $b$ ,  $C$  and  $F$ .

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$b = (9, 12, 11, 15, 2, 25, 3, 17)^T$$

$$C = \begin{pmatrix} 14 & 7 & 11 & 12 & 9 & 9 & 13 & 11 & 20 & 2 & 9 & 16 & 7 & 5 & 20 & 8 \\ 5 & 7 & 11 & 6 & 11 & 7 & 10 & 16 & 11 & 20 & 2 & 13 & 18 & 20 & 16 & 9 \\ 6 & 6 & 18 & 8 & 20 & 12 & 20 & 2 & 15 & 1 & 7 & 19 & 4 & 16 & 9 & 18 \\ 16 & 19 & 3 & 4 & 11 & 11 & 7 & 18 & 5 & 1 & 10 & 20 & 8 & 5 & 4 & 16 \\ 19 & 19 & 18 & 7 & 17 & 15 & 14 & 17 & 10 & 17 & 15 & 13 & 13 & 20 & 10 & 10 \end{pmatrix}$$

$$F = (5, 6, 3, 4, 9, 7, 7, 2, 5, 1, 2, 13, 4, 5, 4, 8)^T$$

**Step 0: Upper Bound :**

The single transportation problem  $(P)$  is then as follows:

$$(P) \left\{ \begin{array}{l} \min \quad 17.00x_{11} + 17.60x_{12} + 15.20x_{13} + 11.40x_{14} + 22.60x_{21} + 17.80x_{22} + 19.80x_{23} + 14.80x_{24} \\ \quad + 17.20x_{31} + 9.20x_{32} + 10.60x_{33} + 29.20x_{34} + 14.00x_{41} + 18.20x_{42} + 15.80x_{43} + 20.20x_{44} \\ x_{11} + x_{12} + x_{13} + x_{14} = 9 \\ x_{21} + x_{22} + x_{23} + x_{24} = 12 \\ x_{31} + x_{32} + x_{33} + x_{34} = 11 \\ x_{41} + x_{42} + x_{43} + x_{44} = 15 \\ x_{11} + x_{21} + x_{31} + x_{41} = 2 \\ x_{12} + x_{22} + x_{32} + x_{42} = 25 \\ x_{13} + x_{23} + x_{33} + x_{43} = 3 \\ x_{14} + x_{24} + x_{34} + x_{44} = 17 \\ x_{ij} \geq 0, \text{ integers} \quad \forall i \in I, j \in J \end{array} \right.$$

After solving the single objective transportation problem  $(P)$  with the MODI method we obtain:

$$X^P = (0, 0, 0, 9, 0, 4, 0, 8, 0, 11, 0, 0, 2, 10, 3, 0)^T \text{ with objective value of } 161.$$

**Step 0.1: Efficiency test:**

$X^P$  is efficient, by applying the efficiency test (refer to (7)).

**Step 0.2:**

$X^{ub} := X^P$  and  $ub := 161$

**Step 0.3: Upper Bound Improvement:**

Let  $W$  be the reduced cost matrix corresponding to the non-basic variables  $x_{12}^{ub}, x_{13}^{ub}, x_{21}^{ub}, x_{23}^{ub}, x_{31}^{ub}, x_{33}^{ub}, x_{34}^{ub}, x_{44}^{ub}$  for the efficient solution  $X^{ub}$ .

$$\text{The matrix } W \text{ is given by: } W = \begin{pmatrix} -3 & -14 & -2 & -11 & 16 & -8 & 12 & 1 \\ 10 & 18 & 6 & 7 & -7 & -14 & -16 & -20 \\ -12 & 7 & 20 & 15 & 26 & 13 & 28 & 12 \\ 22 & 7 & -3 & -3 & 1 & 10 & 12 & 4 \\ 14 & 23 & 9 & 9 & 0 & 8 & -6 & -12 \end{pmatrix}$$

After solving the linear program (9) for each non-basic variable to determine the efficient ones, we find that only the non-basic variables  $x_{12}^{ub}, x_{13}^{ub}$ , lead to an optimal solution in linear programs (20) and (21), respectively. Therefore,  $x_{12}^{ub}$  and  $x_{13}^{ub}$  are efficient.

$$\begin{cases} \min & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 \\ 3\lambda_1 - 10\lambda_2 + 12\lambda_3 - 22\lambda_4 - 14\lambda_5 \geq 0 \\ 14\lambda_1 - 18\lambda_2 - 7\lambda_3 - 7\lambda_4 - 23\lambda_5 \geq 0 \\ 2\lambda_1 - 6\lambda_2 - 20\lambda_3 + 3\lambda_4 - 9\lambda_5 \geq 0 \\ 11\lambda_1 - 7\lambda_2 - 15\lambda_3 + 3\lambda_4 - 9\lambda_5 \geq 0 \\ -16\lambda_1 + 7\lambda_2 - 26\lambda_3 - \lambda_4 \geq 0 \\ 8\lambda_1 + 14\lambda_2 - 13\lambda_3 - 10\lambda_4 - 8\lambda_5 \geq 0 \\ -12\lambda_1 + 16\lambda_2 - 28\lambda_3 - 12\lambda_4 + 6\lambda_5 \geq 0 \\ -\lambda_1 + 20\lambda_2 - 12\lambda_3 - 4\lambda_4 + 12\lambda_5 \geq 0 \\ 3\lambda_1 - 10\lambda_2 + 12\lambda_3 - 22\lambda_4 - 14\lambda_5 = 0 \\ \lambda_i \geq 1, \quad i = 1, 2, 3, 4, 5 \end{cases} \quad (20)$$

$$\begin{cases} \min & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 \\ 3\lambda_1 - 10\lambda_2 + 12\lambda_3 - 22\lambda_4 - 14\lambda_5 \geq 0 \\ 14\lambda_1 - 18\lambda_2 - 7\lambda_3 - 7\lambda_4 - 23\lambda_5 \geq 0 \\ 2\lambda_1 - 6\lambda_2 - 20\lambda_3 + 3\lambda_4 - 9\lambda_5 \geq 0 \\ 11\lambda_1 - 7\lambda_2 - 15\lambda_3 + 3\lambda_4 - 9\lambda_5 \geq 0 \\ -16\lambda_1 + 7\lambda_2 - 26\lambda_3 - \lambda_4 \geq 0 \\ 8\lambda_1 + 14\lambda_2 - 13\lambda_3 - 10\lambda_4 - 8\lambda_5 \geq 0 \\ -12\lambda_1 + 16\lambda_2 - 28\lambda_3 - 12\lambda_4 + 6\lambda_5 \geq 0 \\ -\lambda_1 + 20\lambda_2 - 12\lambda_3 - 4\lambda_4 + 12\lambda_5 \geq 0 \\ 14\lambda_1 - 18\lambda_2 - 7\lambda_3 - 7\lambda_4 - 23\lambda_5 = 0 \\ \lambda_i \geq 1, \quad i = 1, 2, 3, 4, 5 \end{cases} \quad (21)$$

The reduced cost vector of  $X_{ub}$  with respect to  $F$  is given by  $\hat{F} = (-3, -3, -5, 0, 3, 0, 1, 0, 5, 0, 2, 17, 0, 0, 0, 8)$ .

$\min\{\hat{f}_{12}, \hat{f}_{13}\} = \min\{-3, -5\}$ , thus the efficient non-basic variable  $x_{13}^{ub}$  is introduced into the basis of the solution  $X_{ub}$ .

The updated upper bound is:  $X^{ub} := (0, 0, 3, 6, 0, 1, 0, 11, 0, 11, 0, 0, 2, 13, 0, 0)$  and  $ub := 146$

**Step 1:**  $l := 0; F^* = +\infty;$

**Step 2:** After solving the single objective transportation ( $OR_0$ ) with the MODI method we obtain:

$$X^0 = (1, 0, 3, 5, 0, 0, 0, 12, 0, 11, 0, 0, 1, 14, 0, 0)^T.$$

**Step 3:**

The efficiency test (refer to (7)) indicates that  $X^0$  is not efficient.

**Step 4:**

The reduced cost matrix  $\hat{C}_0$  corresponding to  $X^0$  is :  $\hat{C}_0 =$

$$\begin{pmatrix} -5 & -4 & -2 & 3 & 16 & 8 & 14 & 16 & 3 \\ 0 & -4 & -10 & -11 & -7 & -22 & -6 & -8 & -10 \\ -12 & 20 & 0 & 8 & 26 & 6 & 28 & -7 & 12 \\ 6 & -19 & -16 & -10 & 1 & 19 & 28 & 9 & 20 \\ -7 & -12 & -21 & -14 & 0 & 6 & 15 & -2 & 9 \end{pmatrix},$$

We then define the following sets:

$$N_0 = \{(2, 1), (3, 1), (1, 2), (2, 2), (2, 3), (3, 3), (4, 3), (3, 4), (4, 4)\}$$

$$H_0 = N_0$$

$$K_0 = \{1, 2, 3, 4, 5\}.$$

**Step 5:**

Therefore, the constraints of the upper bound and the efficient cut are respectively:

$$\begin{aligned} &5x_{11} + 6x_{12} + 3x_{13} + 4x_{14} + 9x_{21} + 7x_{22} + 7x_{23} + 2x_{24} \\ &+ 5x_{31} + 2x_{32} + 2x_{33} + 13x_{34} + 4x_{41} + 5x_{42} + 4x_{43} + 8x_{44} \leq 146 \end{aligned}$$

$$\begin{aligned} &60x_{11} + 52x_{12} + 61x_{13} + 37x_{14} + 48x_{21} + 54x_{22} + 53x_{23} + 64x_{24} \\ &+ 18x_{31} + 41x_{32} + 41x_{33} - 4x_{34} + 50x_{41} + 66x_{42} + 34x_{43} + 17x_{44} \leq 2616 \end{aligned}$$

**Step 6:**

Setting  $l := 1$ , after adding the constraints to  $(OR_0)$ , the resulting problem, denoted as  $(OR_1)$ , is then solved:

$$\left\{ \begin{array}{l} \min FX = 5x_{11} + 6x_{12} + 3x_{13} + 4x_{14} + 9x_{21} + 7x_{22} + 7x_{23} + 2x_{24} \\ \quad + 5x_{31} + x_{32} + 2x_{33} + 13x_{34} + 4x_{41} + 5x_{42} + 4x_{43} + 8x_{44} \\ x_{11} + x_{12} + x_{13} + x_{14} = 9 \\ x_{21} + x_{22} + x_{23} + x_{24} = 12 \\ x_{31} + x_{32} + x_{33} + x_{34} = 11 \\ x_{41} + x_{42} + x_{43} + x_{44} = 15 \\ x_{11} + x_{21} + x_{31} + x_{41} = 2 \\ x_{12} + x_{22} + x_{32} + x_{42} = 25 \\ x_{13} + x_{23} + x_{33} + x_{43} = 3 \\ x_{14} + x_{24} + x_{34} + x_{44} = 17 \\ 5x_{11} + 6x_{12} + 3x_{13} + 4x_{14} + 9x_{21} + 7x_{22} + 7x_{23} + 2x_{24} \\ \quad + 5x_{31} + x_{32} + 2x_{33} + 13x_{34} + 4x_{41} + 5x_{42} + 4x_{43} + 8x_{44} \leq 146 \\ 60x_{11} + 52x_{12} + 61x_{13} + 37x_{14} + 48x_{21} + 54x_{22} + 53x_{23} + 64x_{24} \\ \quad + 18x_{31} + 41x_{32} + 4x_{33} - 4x_{34} + 50x_{41} + 66x_{42} + 34x_{43} + 17x_{44} \leq 2616 \\ x_{ij} \geq 0, \text{ integers} \quad \forall i \in I, j \in J \end{array} \right.$$

The resulting solution is:  $X^1 = (0, 1, 3, 5, 0, 0, 0, 12, 0, 11, 0, 0, 2, 13, 0, 0)^T$ .

$X^1$  is, by definition, an extreme point and an efficient solution by applying the efficiency test (refer to (7)).

The algorithm terminates and the optimal solution for  $(O)$  is:

$$X^* = (0, 1, 3, 5, 0, 0, 0, 12, 0, 11, 0, 0, 2, 13, 0, 0)^T \text{ with } F^* = 143.$$

**4 EXPERIMENTS**

We conducted a comparison with BCO-method (see Belkhiri et al. (2022) ), which is a method for optimizing a linear function over an efficient set. The experiments were performed on an Intel Core i5-7300U CPU (2.60 GHz) and 8 GB RAM and both OTP-Algorithm and BCO-method were implemented in MATLAB R2015b.  $(OR_l)$  programs are solved using IBM ILOG CPLEX Optimization Studio V12.10.0.

In our experiments, we randomly generated 140 instances of a multi-objective transportation problem with  $r$  objectives and  $n$  resources. The coefficients for the objective functions  $C$  are uniformly distributed in  $[1, 50]$ , while the coefficients for  $F$  belong to  $[1, 30]$ . Similarly, both resource availabilities  $a_i$  and demands  $b_j$  range uniformly in  $[1, 100]$ . The computational experiments were limited to a maximum runtime of 2 hours. In Table 1 we can see results including the mean of the running times.

**Table 1** – Comparison results.

| $r$ | $n = m$ | OTP-Algorithm.CPU (seconds) | BCO-method.CPU (seconds) |
|-----|---------|-----------------------------|--------------------------|
|     |         | mean                        | mean                     |
| 2   | 10      | 0.24                        | 9.38                     |
|     | 20      | 0.36                        | 15.19                    |
|     | 30      | 0.94                        | 24.40                    |
|     | 40      | 1.67                        | 1980.19                  |
|     | 50      | 2.70                        | 4120.00                  |
|     | 60      | 3.88                        | 5400.90                  |
| 3   | 10      | 0.27                        | 30.54                    |
|     | 20      | 0.48                        | 50.54                    |
|     | 30      | 11.29                       | 64.20                    |
|     | 40      | 16.78                       | 3080.00                  |
|     | 50      | 62.01                       | > 7200                   |
|     | 60      | 122.41                      | > 7200                   |
| 4   | 10      | 0.29                        | 127.42                   |
|     | 20      | 1.98                        | 198.20                   |
|     | 30      | 31.24                       | 1476.41                  |
|     | 40      | 376.02                      | > 7200                   |
|     | 50      | 1372.27                     | > 7200                   |
|     | 60      | 3731.86                     | > 7200                   |
| 5   | 10      | 1.23                        | 650.65                   |
|     | 20      | 19.86                       | 1740.01                  |
|     | 30      | 118.30                      | 2240.01                  |
|     | 40      | 716.15                      | > 7200                   |
|     | 50      | 1808.87                     | > 7200                   |
|     | 60      | 4183.18                     | > 7200                   |
| 6   | 10      | 3.43                        | 3886.15                  |
|     | 20      | 29.31                       | > 7200                   |
|     | 30      | 292.36                      | > 7200                   |
|     | 40      | 1923.35                     | > 7200                   |

We observe that for both algorithms, the mean CPU time increases with problem size ( $n = m$ ) and the number of criteria  $r$ . BCO-method is significantly more sensitive to increases in problem size and criteria compared to OTP-Algorithm. For example:

- For  $r = 2$ , BCO-method's CPU time increases from 9.38 seconds ( $n = m = 10$ ) to 5400 seconds ( $n = m = 60$ ).
- For  $r = 6$ , the BCO-method fails to solve problems larger than ( $n = m = 20$ ) within 2 hours limit, while the OTP-Algorithm still performs relatively well.

OTP-Algorithm is highly efficient for smaller problems and scales better with increasing problem size and criteria.

Even for large problems ( $n = m = 60$ ) and high criteria ( $r = 5$ ), OTP-Algorithm completes computations in 4183.18 seconds, whereas BCO-method fails to solve such problems within 2 hour limit.

Increasing the number of criteria ( $r$ ) has a more pronounced effect on BCO-method than on OTP-Algorithm.

For example, for  $n = m = 10$ :

- $r = 2$ , BCO-method takes 9.38 seconds, while OTP-Algorithm takes 0.24 seconds.
- $r = 6$ , BCO-method takes 3886.15 seconds, while OTP-Algorithm takes only 3.43 seconds.

The results clearly demonstrate that OTP-Algorithm, operating in the criteria space, is better suited for solving big instances. In contrast, BCO-method, operating in the decision space, is less effective for larger instances.

## 5 CONCLUSION

In this study, an exact method is developed to solve the problem of finding an optimal solution for a single-objective transportation problem over the set of efficient solutions of a multi-objective transportation problem (MOTP). An aspect of our approach involves using descent directions of the criteria to construct new effective cuts. Additionally, by evaluating an upper bound for the objective function of the single-objective transportation problem, we introduced another type of cut. The combination of these cuts contributed positively to the rapid convergence of our method. We conducted a comparative study with the BCO-method. The results demonstrate the efficiency of the proposed method. Future research, particularly in the direction of hybrid approaches, holds great potential for further improving the robustness and efficiency of our cutting plane method.

## References

- ABBAS M & CHAABANE D. 2006. Optimizing a linear function over an integer efficient set. *European Journal of Operational Research*, **174**(4): 1140–1161.
- ABD ELAZEEM AEM, MOUSA AAA, EL-SHORBAGY MA, ELAGAN SK & ABO-ELNAGA Y. 2021. Detecting all non-dominated points for multi-objective multi-index transportation problems. *Sustainability*, **13**(3): 1372.
- AL-RAJHI J, ABDELWALI HA, ELLAIMONY EE & SWILEM S. 2013. A decomposition algorithm for solving a class of bi-criteria multistage transportation problem with case study. *Int. J. Innov. Res. Sci. Eng. Technol.*, **2**(9).

- ALARDHI M, ABDELWALI HA, KHALFAN AM & ABDELATI MH. 2022. Using the minimize distance method to find the best compromise solution of multi-objective transportation problem with case study. *International Journal of Traffic and Transportation Engineering*, **11**(2): 24–30.
- ALKHULAIFI K, ALRAJHI J, ELLAIMONY EE, ALARDHI M & ABDELWALI HA. 2014. An algorithm for solving bi-criteria large scale transshipment problems. *Global Journal of Researches in Engineering: B Automotive Engineering*, **14**(4): 1–2.
- ANEJA YP & NAIR KP. 1979. Bicriteria transportation problem. *Management Science*, **25**(1): 73–78.
- ANUKOKILAA P, RADHAKRISHNAN B & RAJESHWARIA M. 2017. Multi-objective transportation problem by using goal programming approach. *International Journal of Pure and Applied Mathematics*, **117**(11): 393–403.
- BELKHIRI H, CHERGUI MEA & OUAIL FZ. 2022. Optimizing a linear function over an efficient set. *Operational Research*, **1**(1): 1–19.
- BENSON HP. 1991. An all-linear programming relaxation algorithm for optimizing over the efficient set. *Journal of Global Optimization*, **1**(1): 83–104.
- BOLAND N, CHARKHGARD H & SAVELSBERGH M. 2017. A new method for optimizing a linear function over the efficient set of a multiobjective integer program. *European Journal of Operational Research*, **260**(3): 904–919.
- CHAABANE D & LACHEMI N. 2023. Optimizing over the efficient set of the binary bi-objective knapsack problem. *Yugoslav Journal of Operations Research*, **33**(1): 91–110.
- DIAZ JA. 1978. Solving multiobjective transportation problem. *Ekonomicko-Matematicky Obzor*, **14**: 267–274.
- DRICI W, OUAIL FZ & MOULAI M. 2018. Optimizing a linear fractional function over the integer efficient set. *Annals of Operations Research*, **267**(1): 135–151.
- GALLAGHER RJ & SALEH OA. 1994. Constructing the set of efficient objective values in linear multiple objective transportation problems. *European Journal of Operational Research*, **73**(1): 150–163.
- GEN M, LI Y & IDA K. 1999. Solving multi-objective transportation problem by spanning tree-based genetic algorithm. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, **82**(12): 2802–2810.
- GUPTA R. 1992. *Operations research*. Krishna Prakashan. Available at: [https://books.google.dz/books?id=Ez\\_JBUtbglIC](https://books.google.dz/books?id=Ez_JBUtbglIC).
- ISERMANN H. 1979. The enumeration of all efficient solutions for a linear multiple-objective transportation problem. *Naval Research Logistics Quarterly*, **26**(1): 123–139.

JORGE JM. 2009. An algorithm for optimizing a linear function over an integer efficient set. *European Journal of Operational Research*, **195**(1): 98–103.

KAUR L, SINGH S, BHANDARI AS, SINGH S & RAM M. 2023. An effective approach for solving multi-objective transportation problem. *Journal of Reliability and Statistical Studies*, pp. 153–170.

LEE SM & MOORE LJ. 1973. Optimizing transportation problems with multiple objectives. *AIIE transactions*, **5**(4): 333–338.

LEMARCHAND L, MASSÉ D, REBREYEND P & HÅKANSSON J. 2018. Multiobjective optimization for multimode transportation problems. *Advances in Operations Research*, **2018**(1): 8720643.

MARDANYA D, MAITY G & KUMAR ROY S. 2022. The multi-objective multi-item just-in-time transportation problem. *Optimization*, **71**(16): 4665–4696.

MOUSA A. 2010. Using genetic algorithm and TOPSIS technique for multiobjective transportation problem: a hybrid approach. *International journal of computer mathematics*, **87**(13): 3017–3029.

NGUYEN NC. 1992. An algorithm for optimizing a linear function over the integer efficient set. Tech. rep.. Konrad-Zuse-Zentrum für Informationstechnik Berlin.

NOMANI MA, ALI I & AHMED A. 2017. A new approach for solving multi-objective transportation problems. *International journal of management science and engineering management*, **12**(3): 165–173.

OUAIL FZ, CHERGUI MEA & MOULAI M. 2017. An exact method for optimizing a linear function over an integer efficient set. *WSEAS Transactions on Circuits and Systems*, **16**(3): 141–148.

RINGUEST JL & RINKS DB. 1987. Interactive solutions for the linear multiobjective transportation problem. *European Journal of operational research*, **32**(1): 96–106.

SHARMA V ET AL. 2023. Optimization of a linear function over an integer efficient set. *Journal of Industrial and Management Optimization*, **19**(12): 8633–8656.

SHI Y. 1995. A transportation model with multiple criteria and multiple constraint levels. *Mathematical and computer modelling*, **21**(4): 13–28.

SINHA SM. 2005. *Mathematical Programming: Theory and Methods*. Elsevier.

SONG JH. 2000. A linear program approach for optimizing a linear function over an efficient set. *European Journal of Operational Research*, **26**(3): 220–226.

- SRINIVASAN V & THOMPSON GL. 1976. Algorithms for minimizing total cost, bottleneck time and bottleneck shipment in transportation problems. *Naval Research Logistics Quarterly*, **23**(4): 567–595.
- STEWART T & ITTMANN HW. 1979. Two-stage optimization in a transportation problem. *Journal of the Operational Research Society*, **30**(10): 897–904.
- SUTCLIFFE C, BOARD J & CHESHIRE P. 1984. Goal programming and allocating children to secondary schools in Reading. *Journal of the Operational Research Society*, **35**(8): 719–730.
- TAHA HA. 2013. *Operations Research: An Introduction*. Pearson Education India, Delhi.
- WU J, PU C, DING S, CAO G, XIA C & PARDALOS PM. 2022. Multi-objective optimization of transport processes on complex networks. *IEEE Transactions on Network Science and Engineering*, **10**(2): 780–794.
- YAMAMOTO Y. 2002. Optimization over the efficient set: Overview. *Journal of Global Optimization*, **22**(3): 285–317.
- ZAKI SA, ABD ALLAH AM, GENEEDI HM & ELMEKAWY AY. 2012. Efficient multiobjective genetic algorithm for solving transportation, assignment, and transshipment problems, .
- ZERFA L & CHERGUI MEA. 2024. An efficient branch-and-bound algorithm to optimize a function over a nondominated set. *International Transactions in Operational Research*, .
- ZHENG YJ & CHEN SY. 2013. Cooperative particle swarm optimization for multiobjective transportation planning. *Applied intelligence*, **39**: 202–216.
- ZHENGLIANG L & EHRGOTT M. 2018. Primal and dual algorithms for optimization over the efficient set. *Optimization*, **67**(10): 1661–1686.

**How to cite**

MEZALI Z AND CHERGUI ME-A. 2025. Optimizing a linear function over the efficient set of a multi-objective transportation problem. *Pesquisa Operacional*, **45**: e294863. doi: 10.1590/0101-7438.2025.045.00294863.

**Authors' Contributions**

This work was conducted as part of the LMD PhD thesis of Ms. Zineb Mezali, under the supervision of Professor M.E-A. Chergui. The authors' contributions, following the CRediT taxonomy, are as follows: M.E-A. Chergui: Conceptualization; Methodology; Supervision; Writing – Review & Editing. Zineb Mezali: Data Curation; Formal Analysis; Software; Visualization; Writing – Original Draft Preparation.

**Data Availability**

The only data used in the experimental study were randomly generated, as described in the Experiments section of the article.

**Editor responsible for the review**

Editor-in-Chief: Annibal Parracho Sant'Anna.