



Ordered weighted average combinatorial optimization: Formulations and their properties



Elena Fernández^a, Miguel A. Pozo^{b,*}, Justo Puerto^b

^a Department of Statistics and Operational Research, BarcelonaTech, Spain

^b Department of Statistics and Operational Research-IMUS, University of Seville, Spain

ARTICLE INFO

Article history:

Received 7 June 2013

Received in revised form 5 November 2013

Accepted 7 January 2014

Available online 28 January 2014

Keywords:

Combinatorial optimization

Multiobjective optimization

Weighted average optimization

Ordered median

ABSTRACT

Multiobjective combinatorial optimization deals with problems considering more than one viewpoint or scenario. The problem of aggregating multiple criteria to obtain a globalizing objective function is of special interest when the number of Pareto solutions becomes considerably large or when a single, meaningful solution is required. Ordered weighted average or ordered median operators are very useful when preferential information is available and objectives are comparable since they assign importance weights not to specific objectives but to their sorted values. In this paper, ordered weighted average optimization problems are studied from a modeling point of view. Alternative integer programming formulations for such problems are presented and their respective domains studied and compared. In addition, their associated polyhedra are studied and some families of facets and new families of valid inequalities presented. The proposed formulations are particularized for two well-known combinatorial optimization problems, namely, shortest path and minimum cost perfect matching, and the results of computational experiments presented and analyzed. These results indicate that the new formulations reinforced with appropriate constraints can be effective for efficiently solving medium to large size instances.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Multiobjective combinatorial optimization deals with problems considering more than one viewpoint or scenario. They inherit the complexity difficulty of their scalar counterparts, but incorporate additional difficulties derived from dealing with partial orders in the objective function space. The standard solution concept is the set of Pareto solutions. However, the number of Pareto solutions can grow exponentially with the size of the instance and the number of objectives. A first approach to overcome this difficulty focuses on a specific subset of the Pareto set, such as, for instance, the supported Pareto solutions (see e.g. [4]). Those are the Pareto solutions that optimize linear scalarizations of the different objectives. However, it is possible to exhibit instances for which even the number of supported solutions grows exponentially with the size of the instance. Furthermore, focusing on supported Pareto solutions a priori excludes compromise solutions that could be preferred by the decision maker. For the above reasons, more involved decision criteria have been proposed in the field of multicriteria decision making [19]. These include objectives focusing on one particular compromise solution, which, for tractability and decision theoretic reasons, seem to be better suited when an appropriate aggregation operator is available.

In some cases, a particularly important Pareto solution related to a weighted ordered average aggregating function is sought. Provided that some imprecise preference information on the objectives is available, and that they are comparable, an averaging operator can be used to aggregate the vector of objective values of feasible solutions. The Ordered Median (OM)

* Corresponding author. Tel.: +34 671178393; fax: +34 671178393.

E-mail addresses: e.fernandez@upc.edu (E. Fernández), miguelpozo@us.es (M.A. Pozo), puerto@us.es (J. Puerto).

objective function is very useful in this context since it assigns importance weights not to specific objectives but to their sorted values. OM operators have been successfully used for addressing various types of combinatorial problems (see, for instance, [18,15,20,1,14] or, [6]).

When applied to values of different objective functions in multiobjective problems, the OM operator is called in the literature Ordered Weighted Average (OWA) [24,25]. It assigns importance weights to the sorted values of the objective function elements in a multiple objective optimization problem. The OWA has been also used in the literature under the name of Choquet optimization to address continuous problems [22,13] and more recently it has been applied to some combinatorial optimization problems, like the minimum spanning tree and 0–1 knapsack [7]. The OWA is, however, a very broad operator, which, depending on the cases, can be seen as an ordered median or as vector assignment ordered median [12], and which can be applied to any combinatorial optimization problem. We therefore believe that its full potential within combinatorial optimization is worth being exploited. This naturally leads to a thorough study of its modeling properties and alternatives, which is the focus of this paper.

From a modeling point of view, the OWA operator can be formulated with a combination of discrete and continuous decision variables linked by several families of linear constraints. Since the domain of combinatorial optimization problems can be characterized with ad hoc discrete variables and linear constraints, it becomes clear that any combinatorial optimization problem with an OWA objective can be formulated as a linear integer programming problem, by suitably relating the two sets of variables and constraints. Of course, not all formulations are equally useful. Moreover, it is not even clear that the best formulation for the domain of the combinatorial object should be preferred, because its “integration” with the formulation of the OWA may imply additional difficulties. In this work we propose three alternative basic formulations for a combinatorial object with an OWA objective. Each basic formulation uses a different set of decision variables to model the OWA objective. We study properties yielding to alternative formulations, which preserve the set of optimal solutions, and we also compare the formulations among them. In addition we propose various families of facets and valid inequalities, which can be used (independently or in combination) to reinforce the basic formulations. For keeping the extension of the paper within some reasonable limits, we report the results obtained with a particular case of the OWA operator, namely the Hurwicz criterion [10]. This criterion, which has been used by other authors in the literature (see e.g. [16,7]) is a non-monotonic and non-convex criterion. In our experience the Hurwicz criterion behaves quite similarly to other non-convex OWA criteria, so the results we report and derived conclusions can be extended to analogous criteria as well. In the final part of the paper, we focus on two classical optimization problems: shortest path and minimum cost perfect matching. For these two problems we analyze the empirical performance of the alternative basic formulations and their possible reinforcements and variations. From our computational experience we cannot conclude that any of the formulations is superior to the others since the behavior of the proposed formulation varies with the different combinatorial object to be considered (see Section 6).

The paper is structured as follows. Section 2 gives the formal definition of the OWA operator and shows that it has as particular cases both the ordered median and the vector assignment ordered median. Section 3 presents the three basic formulations, and their variations, for a combinatorial problem with an OWA objective, studies their properties and compares them, whereas Section 4 presents different families of valid inequalities and possible reinforcements. Sections 5.1 and 5.2 respectively present the formulation of the combinatorial object that we use in our empirical study of the shortest path and minimum cost perfect matching problems with an OWA objective. Finally, Section 6 describes the computational experiments that we have run and presents and analyzes the obtained numerical results. The paper ends in Section 7 with some comments and possible avenues for future research.

2. The ordered weighted average optimization

The Ordered Weighted Average (OWA) operator is defined over a feasible set $Q \subseteq \mathbb{R}^n$. Let $C \in \mathbb{R}^{p \times n}$ be a given matrix, whose rows, denoted by C^i , are associated with the cost vectors of p objective functions. The index set for the rows of C is denoted by $P = \{1, \dots, p\}$. For $x \in Q$, the vector $y \in \mathbb{R}^p$ is referred to as the outcome vector relative to C . In the following we assume $y = Cx$, with $x \in Q$. For a given y , let σ be a permutation of the indices of $i \in P$ such that $y_{\sigma_1} \geq \dots \geq y_{\sigma_p}$. Let also $\omega \in \mathbb{R}^{p+}$ denote a vector of non-negative weights. Feasible solutions $x \in Q$ are evaluated with an operator defined as $\text{OWA}_{(C,\omega)}(x) = \omega' y_{\sigma}$. The OWA optimization Problem (OWAP) is to find $x \in Q$ of minimum value with respect to the above operator, that is

$$\text{OWAP} : \min_{x \in Q} \text{OWA}_{(C,\omega)}(x).$$

Example 1. Consider

$$Q = \{x \in \{0, 1\}^3 : x_1 + x_2 + x_3 = 2\}, \quad C = \begin{pmatrix} 1 & 4 & 1 \\ 1 & 1 & 3 \\ 5 & 1 & 2 \end{pmatrix} \quad \text{and} \quad \omega' = (1 \quad 2 \quad 4).$$

Table 1 illustrates, for each feasible $x \in Q$, the values of $y = Cx$, y_{σ} and $\text{OWA}_{(C,\omega)}(x) = \omega' y_{\sigma}$. The optimal value to the OWAP is $\min_{x \in Q} \text{OWA}_{(C,\omega)}(x) = 23$.

The OWA operator is a very general function which, as we see below, has as particular cases well-known objective functions. We next describe some of them.

Table 1
Solutions $x \in Q$, values $y = Cx$, sorted values y_σ and $OWA_{(C,\omega)}(x)$ for Example 1.

x	y	y_σ	$OWA_{(C,\omega)}(x) = \omega' y_\sigma$
$(1 \ 1 \ 0)'$	$(5 \ 2 \ 6)'$	$(6 \ 5 \ 2)'$	24
$(1 \ 0 \ 1)'$	$(2 \ 4 \ 7)'$	$(7 \ 4 \ 2)'$	23
$(0 \ 1 \ 1)'$	$(5 \ 4 \ 3)'$	$(5 \ 4 \ 3)'$	25

Table 2
Solutions $x \in Q$, values $d_j x_j$, sorted $d_{\sigma_j} x_{\sigma_j}$ and $OM_{(d,\omega)}(x)$ for the OM of Example 2.

x	$(d_j x_j)_{j \in P}$	$(d_{\sigma_j} x_{\sigma_j})_{j \in P}$	$OM_{(d,\omega)}(x) = \sum_{j \in P} \omega_j d_{\sigma_j} x_{\sigma_j}$
$(1 \ 1 \ 0)'$	$(5 \ 1 \ 0)'$	$(5 \ 1 \ 0)'$	7
$(1 \ 0 \ 1)'$	$(5 \ 0 \ 2)'$	$(5 \ 2 \ 0)'$	9
$(0 \ 1 \ 1)'$	$(0 \ 1 \ 2)'$	$(2 \ 1 \ 0)'$	4

Table 3
The OM instance of Example 2 as an OWA: $y = Cx$, y_σ and $OWA_{(C,\omega)}(x)$.

x	y	y_σ	$OWA_{(C,\omega)}(x) = \omega' y_\sigma$
$(1 \ 1 \ 0)'$	$(5 \ 1 \ 0)'$	$(5 \ 1 \ 0)'$	7
$(1 \ 0 \ 1)'$	$(5 \ 0 \ 2)'$	$(5 \ 2 \ 0)'$	9
$(0 \ 1 \ 1)'$	$(0 \ 1 \ 2)'$	$(2 \ 1 \ 0)'$	4

2.1. The Ordered Median (OM) objective function

The OM objective [15] minimizes a weighted sum of ordered elements. It is a well known function that unifies many location problems as the p -median problem, the p -center problem, etc.

Let $Q \subseteq \mathbb{R}^n$ denote the feasible domain for an optimization problem and let $d \in \mathbb{R}^n$ be a cost vector and $\omega \in \mathbb{R}^n$ a given weights vector. For $x \in Q$, let σ denote a permutation of the indices of x , such that $d_{\sigma_j} x_{\sigma_j} \geq d_{\sigma_{j+1}} x_{\sigma_{j+1}}$, $j \in \{1, 2, \dots, n - 1\}$. The OM operator is $OM_{(d,\omega)}(x) = \sum_{j \in P} \omega_j d_{\sigma_j} x_{\sigma_j}$ and the OM Problem (OMP) is therefore defined as

$$OMP: \min_{x \in Q} OM_{(d,\omega)}(x) = \sum_{j \in P} \omega_j d_{\sigma_j} x_{\sigma_j}.$$

To cast the OM operator as an OWA operator, we only need to set the rows of the C matrix as $(C^i)' = d_i e^i$, $i \in \{1, \dots, n\}$, where $e^i \in \mathbb{R}^n$ is the i th unit vector of the canonical basis of \mathbb{R}^n . Let $Diag(d)$ denote the diagonal matrix whose diagonal entries are the components of the vector d , thus, $C = Diag(d)$. Then $OM_{(d,\omega)}(x) = OWA_{(Diag(d),\omega)}(x)$.

Example 2. Consider

$$Q = \{x \in \{0, 1\}^3 : x_1 + x_2 + x_3 = 2\}, \quad d = (5 \ 1 \ 2)'$$
 and $\omega = (1 \ 2 \ 4)'$.

Table 2 illustrates, for each feasible $x \in Q$, the values of $(d_j x_j)_{j \in P}$, $(d_{\sigma_j} x_{\sigma_j})_{j \in P}$, and $OM_{(d,\omega)}(x) = \sum_{j \in P} \omega_j d_{\sigma_j} x_{\sigma_j}$. The optimal OM value is $\min_{x \in Q} OM_{(d,\omega)}(x) = 4$.

To cast the OM operator as an OWA operator, we only need to set the rows of the C matrix as

$$C = Diag(d) = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

The values of $y = Cx$, y_σ and $OWA_{(C,\omega)}(x) = \omega' y_\sigma$ are shown in Table 3. The optimal OWA value is $\min_{x \in Q} OWA_{(Diag(d),\omega)}(x) = \min_{x \in Q} OM_{(d,\omega)}(x) = 4$.

2.2. The vector assignment ordered median objective function

The Vector Assignment Ordered Median (VAOM) problem was recently introduced by Lei and Church [12] in the context of discrete location–allocation problems. In this context, the VAOM generalizes both OM and Vector Assignment Median [23]. As we see below the OWA generalizes the VAOM as well. First, we briefly introduce the VAOM.

The main decisions in location–allocation problems are the set of facilities to open, and the assignment of customers to open facilities so as to satisfy their demand. Consider a given set of customers $P = \{1, \dots, p\}$, where each customer is also a potential location for a facility, and let $q \leq p$ denote the number of facilities to open. Associated with each customer $i \in P$

there is a demand a_i . A unit of demand at customer i served from facility k incurs a cost d_{ik} . We will use \mathbf{d}^i to denote the p dimensional vector of the distances associated with customer i . Usual objectives focus on service cost minimization.

Many location–allocation models allow splitting the demand at customers among several facilities, so allocating customer i to facility k means that some positive fraction of a_i is served from facility k . However, without any further incentive or constraint, in optimal solutions customers will be allocated to one single facility, the closest one among those that are open. Since such solutions often exhibit privileged customers, equity measures have been proposed to balance out the service level of the customers. This is the case of the VAOM that imposes the specific fractions of the demand at each customer to be served from the various open facilities. Let $\gamma_{i\ell}$ denote the fraction of a_i that must be served from the ℓ th closest facility to customer i where $\ell \in I = \{1, \dots, q\}$. To measure the service level of customer i in a given solution, the distances from i to the different open facilities are ordered and weighted with the values $\gamma_{i\ell}$ according to their rank in the sorted list of distances. This invites to characterize solutions by means of binary decision variables $x_{k\ell}^i$, $i, k \in P, \ell \in I$, where $x_{k\ell}^i$ is equal to 1 if i is allocated to facility k as the ℓ th closest facility. Now, the service cost of customer i can be computed as $s_i = \sum_{k \in P} \sum_{\ell \in I} a_i \gamma_{i\ell} d_{ik} x_{k\ell}^i$. Note that s_i can be expressed in a compact way as $s_i = \bar{C}^i \mathbf{x}^i$, where \mathbf{x}^i is the vector of decision variables $(x_{k\ell}^i)_{k \in P, \ell \in I} = (x_{11}^i, x_{12}^i, \dots, x_{21}^i, x_{22}^i, \dots)'$, and $(\bar{C}^i)' = (a_i \gamma_{i\ell} d_{ik})_{k \in P, \ell \in I}$.

The VAOM operator is computed as a weighted sum of the service costs of all customers. A weight ω_j is applied to the customer with the j th lowest service level, i.e. with the j th highest service cost. For a given solution, x , and its associated vector s as defined above, let σ be a permutation of the indices of P such that $s_{\sigma_1} \geq \dots \geq s_{\sigma_p}$. Then, $\text{VAOM}_{(d, \gamma, a, \omega)}(x) = \sum_{j=1}^p \omega_j s_{\sigma_j}$ and the VAOM Problem (VAOMP) is therefore defined as

$$\text{VAOMP: } \min_{x \in Q} \text{VAOM}_{(d, \gamma, a, \omega)}(x) = \sum_{j=1}^p \omega_j s_{\sigma_j}.$$

The set of feasible solutions to the problem is fully characterized by the set of feasible assignments, since an explicit representation of the open facilities is not needed. These can be obtained directly from x by identifying the indices $k \in P$ with $x_{k\ell}^i = 1$ for some $i \in P, \ell \in I$. Thus in this problem Q is given by the set of feasible assignments. For reasons that will become evident when we cast the VAOM operator as an OWA, we express the assignment vectors x as one dimensional n vectors with $n = p^2 q$. In particular x is partitioned in p blocks, each of them associated with a different customer $i \in P$. That is, $x = (\mathbf{x}^1 | \dots | \mathbf{x}^{p'})'$. In turn, each block \mathbf{x}^i consists of p smaller blocks, each with q components. The k th block of \mathbf{x}^i contains the q components $x_{k\ell}^i$ for the indices $\ell \in I$.

Now, to cast the VAOM as an OWA operator, we define p objective functions $\bar{C}^i \mathbf{x}^i$, one associated with each customer $i \in P$. In particular, objective $\bar{C}^i \mathbf{x}^i$ represents the service cost of customer $i \in P, s^i$. With the above characterization of vectors $x \in Q$, each \bar{C}^i must be defined by an n vector. Thus expressing the VAOM as an OWA becomes basically a notation issue. For each fixed $i \in P$, again we partition the cost vector \bar{C}^i in p blocks. Similarly to the partition of vectors $x \in Q$, each block corresponds to a different customer, and has pq components. We now set at value 0 all the entries except those in the block of customer i , which are given by the entries of the vector \bar{C}^i as defined above. That is: $C^i = (\mathbf{0}_{pq} | \dots | \bar{C}^i | \dots | \mathbf{0}_{pq})$, where $\mathbf{0}_{pq} = (0, \dots, 0) \in \mathbb{R}^{pq}$. With this notation it becomes clear that $C^i x = \bar{C}^i \mathbf{x}^i$. Hence,

$$\text{VAOM}_{(d, \gamma, a, \omega)}(x) = \text{OWA}_{(C, \omega)}(x).$$

Example 3. Consider an instance of a VAOM problem with $p = 3$ customers in which $q = 2$ facilities must open. Suppose all the customers have one unit of demand, i.e. $a_1 = a_2 = a_3 = 1$, and suppose the rest of the data is the following:

$$(d_{ik})_{i, k \in P} = \begin{pmatrix} 0 & 2 & 6 \\ 2 & 0 & 4 \\ 8 & 4 & 0 \end{pmatrix}, \quad (\gamma_{i\ell})_{i \in P, \ell \in I} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \\ 1 & 0 \end{pmatrix}, \quad \omega' = (0 \quad 1 \quad 2).$$

Since $q = 2$ the feasible combinations of facilities to open are $\{1, 2\}$, $\{1, 3\}$ and $\{2, 3\}$. When the distances of each customer to the potential facilities are all different, like in this example, each combination of open facilities determines a unique feasible assignment vector x . For instance, when facilities 1 and 2 open, then customer 1, has facility 1 as the closest and facility 2 as the second closest, so $x_{11}^1 = x_{22}^1 = 1$, and $x_{12}^1 = x_{21}^1 = x_{31}^1 = x_{32}^1 = 0$. The service cost of customer 1 is thus $s_1 = \gamma_{11} d_{11} x_{11}^1 + \gamma_{12} d_{12} x_{22}^1 = 0 + 0.5 \times 2 = 1$. For customer 2 we have $x_{12}^2 = x_{21}^2 = 1$, and $x_{11}^2 = x_{22}^2 = x_{31}^2 = x_{32}^2 = 0$, with service cost $s_2 = 0 + 0.5 \times 2 = 1$. With this set of open facilities, the assignment for customer 3 is $x_{12}^3 = x_{21}^3 = 1$, and $x_{11}^3 = x_{22}^3 = x_{31}^3 = x_{32}^3 = 0$ with service cost $s_3 = 4$. Since $s_3 \geq s_1 \geq s_2$ the objective function value for this solution is thus $0 \times 4 + 1 \times 1 + 2 \times 1 = 3$.

Proceeding similarly with the other possible combinations of open facilities we obtain the complete set of feasible solutions Q , which in this example is given by the set of binary vectors given in Table 4:

For modeling the VAOM as an OWA we define the cost matrix C as:

$$C = \begin{pmatrix} 0 & 0 & | & 1 & 1 & | & 3 & 3 & | & 0 & 0 & | & 0 & 0 & | & 0 & 0 & | & 0 & 0 & | & 0 & 0 \\ 0 & 0 & | & 0 & 0 & | & 0 & 0 & | & 1 & 1 & | & 0 & 0 & | & 2 & 2 & | & 0 & 0 & | & 0 & 0 \\ 0 & 0 & | & 0 & 0 & | & 0 & 0 & | & 0 & 0 & | & 0 & 0 & | & 0 & 0 & | & 8 & 0 & | & 4 & 0 \end{pmatrix}.$$

Table 4
Complete set of feasible solutions Q as binary vectors for Example 3.

x_{11}^1	x_{12}^1	x_{21}^1	x_{22}^1	x_{31}^1	x_{32}^1	x_{11}^2	x_{12}^2	x_{21}^2	x_{22}^2	x_{31}^2	x_{32}^2	x_{11}^3	x_{12}^3	x_{21}^3	x_{22}^3	x_{31}^3	x_{32}^3
1	0	0	1	0	0	0	1	1	0	0	0	0	1	1	0	0	0
1	0	0	0	0	1	1	0	0	0	0	1	0	1	0	0	1	0
0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	1	0

Table 5
Values of y , y_σ and $OWA_{(C,\omega)}(x)$ for the feasible solutions of Example 3.

y	y_σ	$OWA_{(C,\omega)}(x) = \omega' y_\sigma$
$(1 \ 1 \ 4)'$	$(4 \ 1 \ 1)'$	3
$(3 \ 3 \ 0)'$	$(3 \ 3 \ 0)'$	3
$(4 \ 2 \ 0)'$	$(4 \ 2 \ 0)'$	2

Table 5 shows the values of y , y_σ and $OWA_{(C,\omega)}(x)$ for each $x \in Q$. The optimal value of the VAOM is $\min_{x \in Q} VAOM_{(d,\gamma,a,\omega)}(x) = \min\{3, 3, 2\} = 2$.

2.3. The vector assignment ordered median function of an abstract combinatorial optimization problem

In the above section we have applied the VAOM operator to the locations and allocations of a general multifacility location problem, according to the original definition by Lei and Church [12]. Nevertheless, this operator can be also applied to the characteristic vector of a combinatorial solution of any abstract combinatorial optimization problem, as we also did with the ordered median operator. In doing that we obtain a more general interpretation of this type of objective function that can also be cast within the OWA operator.

Let $Q \subseteq \mathbb{R}^n$ denote the feasible domain for an optimization problem, $\omega \in \mathbb{R}^{p+}$ a given vector of nonnegative weights and $P = \{1, \dots, p\}$. Recall that a VAOM operator considers for each objective function s_i , $i \in P$ different fractions, γ^i , of the cost vector d for the sorted elements of the decision vector x .

For $x \in Q$, the evaluation of the i th component of the VAOM objective is given by $s_i = \gamma^i d_i x_i$, for all $i \in P$. Let σ denote a permutation of the indices of P , such that $s_{\sigma_i} \geq s_{\sigma_{i+1}}$ for $i = 1, \dots, p-1$. The VAOM operator is $VAOM_{(d,\gamma,\omega)}(x) = \sum_{i \in P} \omega_i s_{\sigma_i}$. The reader may note that the original definition of VAOM can be accommodated to this general setting once we identify the combinatorial object Q as the set of location–allocations in the discrete location problem. In that case, there are $i = 1, \dots, n$ objective functions associated with each of the customers and then the fractions that apply to each customer i are non-null only for a subset of the open facilities (servers) corresponding to the q -closest ones.

This can be done by defining a set of variables, one per customer i , with n blocks. In the block k , $x_{i,k}^i = (x_{1k}^i, \dots, x_{nk}^i)'$ accounts for the allocation of i to any facility as the k th closest, therefore $x^i = (x_{i,1}^i | x_{i,2}^i | \dots | x_{i,n}^i)'$ for $i = 1, \dots, n$. This way, the cost vectors must also have the same structure by blocks, each block corresponding with the level of assignment, i.e. denoting by $d^i = (d_1^i, \dots, d_n^i) \in \mathbb{R}^{n^2}$ then $d^i = (d^{i,1} | d^{i,2} | \dots | d^{i,n})'$. Finally, since the fractions of costs are applied according to the level of assignment, the structure of the vector of fractions γ^i is also by blocks. Block k represents the fraction of the cost that is accounted for customer i at the k th level of assignment. Denoting by $\gamma_\ell^i = (\gamma_{i\ell}^i, \dots, \gamma_{i\ell}^i) \in \mathbb{R}^n$ then $\gamma^i = (\gamma_1^i | \gamma_2^i | \dots | \gamma_n^i)'$ for $i = 1, \dots, n$.

To cast the VAOM as an OWA operator, we only need to set $\tilde{\gamma}^i = (\underbrace{\mathbf{0}_{np}}_1 | \dots | \underbrace{\gamma^i}_i | \dots | \underbrace{\mathbf{0}_{np}}_p)'$, $\tilde{d}^i = (\underbrace{\mathbf{0}_{np}}_1 | \dots | \underbrace{d^i}_i | \dots | \underbrace{\mathbf{0}_{np}}_p)'$, $x = (\mathbf{x}^1 | \dots | \mathbf{x}^i | \dots | \mathbf{x}^n)'$ and $C^i = (\tilde{\gamma}_j^i \tilde{d}_j^i)_{j=1}^{p(pn)}$. Then, the VAOM can be written

as the following OWA operator $VAOM_{(d,\gamma,\omega)}(x) = OWA_{(C,\omega)}(x)$.

As we have shown above, OWA is a very general operator. In the following, we will work in more particular settings, namely we shall restrict ourselves to assume that Q is a combinatorial object which can be represented by a system of linear inequalities.

3. Basic formulations for the OWAP, properties and reinforcements

This section presents alternative Mixed Integer Programming (MIP) formulations for an OWAP, which are analyzed and compared. The starting point of our study is three basic formulations, which, broadly speaking, differs from one to another on how the permutation that defines the ordering of the cost function values is modeled. Two of the formulations presented use binary variables z to define the *specific positions* in the ordering of the sorted cost function values, whereas the other one uses binary variables s to define the *relative position* in the ordering of the sorted cost function values. One of the two formulations based on the z variables also uses an additional set of decision variables y for modeling the specific values of

the cost functions depending on their position in the ordering. All three formulations use a set of decision variables θ to compute the values of the objectives sorted at specific positions. In each case, alternative formulations are presented, which preserve the set of optimal solutions. Before addressing any concrete formulation we discuss the meaning of both sets of variables z and s as well as their relationships.

3.1. Alternative formulations for permutations

The essential element in our formulations rests on the representation of ordering within a MIP model. To such end, we devote this section to describe how a permutation can be represented with binary variables. Recall that we have introduced $P = \{1, \dots, p\}$ as the set of the cost function indices. Let $\pi : P \rightarrow P$ be a function representing a permutation of P . That is, it assigns the index i of each cost function (also denoted by *cost function* i) to a position indexed by j (also denoted by *position* j). Note that π is a permutation if each cost function is assigned to a single position and if each position contains a single cost function index. In what follows, we use $\pi_i = \pi(i)$ to denote the position occupied by the cost function $i \in P$ and $\sigma_j = \pi^{-1}(j)$ to denote the index of the cost function that occupies position j (we recall that the notation σ was previously used in Section 2). Note that σ also defines a permutation of the positions of P . In what follows we will indistinctively use π and σ . Slightly abusing notation we refer to π as to *the cost functions permutation* and to its inverse σ as to the *positions permutation*.

In order to model π as a permutation, let z_{ij} be a binary decision variable defined as

$$z_{ij} = \begin{cases} 1 & \text{if cost function } i \text{ occupies position } j, \text{ (i.e. if } \pi_i = j) \\ 0 & \text{otherwise.} \end{cases}$$

The set of variables z defines a permutation if:

- (i) each position contains a single cost function:

$$\sum_{i \in P} z_{ij} = 1 \quad j \in P, \tag{1}$$

and,

- (ii) each cost function i is assigned to a single position j :

$$\sum_{j \in P} z_{ij} = 1 \quad i \in P. \tag{2}$$

In addition, we observe that since system (1)–(2) contains exactly $2p - 1$ linearly independent equations, the above permutation can also be represented without variables z_{i1} , for all $i \in P$, that can be replaced by $1 - \sum_{j \in P: j > 1} z_{ij}$. In this way, system (1)–(2) can also be rewritten as

$$\sum_{i \in P} z_{ij} = 1 \quad j \in P : j > 1, \tag{3}$$

$$\sum_{j \in P} z_{ij} \leq 1 \quad i \in P. \tag{4}$$

Example 4. Let π be a permutation defined by $\pi = (3 \ 2 \ 4 \ 1)$ or equivalently by $\sigma = (4 \ 2 \ 1 \ 3)$. Then, π can be represented by using variables z as follows:

$$(z_{i,j})_{i,j \in P} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad \text{or} \quad (z_{i,j})_{i,j \in P: j > 1} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}. \quad \square$$

An alternative representation of a permutation, which we have also found useful is based on a different set of variables defined as:

$$s_{ij} = \begin{cases} 1 & \text{if cost function } i \text{ is placed before position } j \text{ in the ordering,} \\ 0 & \text{otherwise.} \end{cases}$$

The set of variables s defines a permutation if:

- (i) for all $j \in P$ there are $j - 1$ cost functions placed before position j :

$$\sum_{i \in P} s_{ij} = j - 1 \quad j \in P, \tag{5}$$

and

- (ii) cost function i cannot be placed in position j unless it is also placed in position $j + 1$, i.e.,

$$s_{ij+1} - s_{ij} \geq 0 \quad i, j \in P : j < p. \tag{6}$$

Again we can reduce the number of decision variables, now by eliminating s_{i1} for all $i \in P$. Indeed, since there is no cost function placed before position 1 in any ordering, all the s_{i1} , $i \in P$ can be fixed to zero. In this way, permutation (3)–(4) can be also represented by means of the following reduced set of constraints:

$$\sum_{i \in P} s_{ij} = j - 1 \quad j \in P : j > 1, \tag{7}$$

$$s_{ij+1} - s_{ij} \geq 0 \quad i, j \in P : 1 < j < p. \tag{8}$$

Example 5. Let π be a permutation defined by $\pi = (3 \ 2 \ 4 \ 1)$ or equivalently by $\sigma = (4 \ 2 \ 1 \ 3)$. Then, π can be represented by using variables s as follows:

$$(s_{i,j})_{i,j \in P} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix}, \quad \text{or} \quad (s_{i,j})_{i,j \in P:j>1} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}. \quad \square$$

With the above considerations, variables z and s are related by means of

$$z_{ij} = \begin{cases} s_{ij+1} - s_{ij} & i \in P, j = 1, \dots, p - 1 \\ 1 - s_{ij} & i \in P, j = p \end{cases} \tag{9}$$

and equivalently,

$$s_{ij} = 1 - \sum_{k \geq j} z_{ik}, \quad i, j \in P. \tag{10}$$

3.2. OWAP formulations with variables for the positions of sorted cost function values

For a given feasible set $Q \subseteq \mathbb{R}^n$, consider the binary decision variables z as defined in Section 3.1 to represent the permutation π associated with the sorted cost function values $C^i x$, $i \in P$. Let also θ_j be a real decision variable equal to the value of the cost function sorted in position j . Next, we give an integer linear programming description of the OWAP where we use M to denote a non-negative upper bound of the value of all the cost functions. (We refer the interested reader to [1] or [15] for similar sets of decision variables and formulations for the discrete ordered median location problem.)

$$F_0^z : V = \min \sum_{j \in P} \omega_j \theta_j \tag{11a}$$

$$\text{s.t.} \quad \sum_{i \in P} z_{ij} = 1 \quad j \in P \tag{11b}$$

$$\sum_{j \in P} z_{ij} = 1 \quad i \in P \tag{11c}$$

$$C^i x \leq \theta_j + M(1 - z_{ij}) \quad i, j \in P \tag{11d^0}$$

$$\theta_j \geq \theta_{j+1} \quad j \in P : j < p \tag{11e}$$

$$x \in Q, z \in \{0, 1\}^{p \times p}. \tag{11f}$$

The objective function (11a) minimizes the weighted average of sorted objective function values, provided that $\theta_j, j \in P$, are enforced to take on the appropriate values. As seen, constraints (11b)–(11c) define a cost functions permutation by placing at each position of π a single cost function and each cost function at a single position of π . Constraints (11d⁰) relate cost function values with the values placed in a sorted sequence. Constraint (11e) imposes that the sorted values are ordered non-increasingly.

In the following we denote by Ω_0^z the domain of feasible solutions to formulation F_0^z . That is,

$$\Omega_0^z = \{(x, z, \theta) \text{ satisfying constraints (11b), (11c), (11d}^0), (11e), (11f)\}.$$

Consider now the family of inequalities

$$C^i x \leq \theta_j + M \left(1 - \sum_{k \geq j} z_{ik} \right) \quad i, j \in P, \tag{11d}$$

and note that, for z satisfying (11c), inequalities (11d) can be rewritten as

$$C^i x \leq \theta_j + M \sum_{k < j} z_{ik} \quad i, j \in P, \tag{11d'}$$

since for all $i, j \in P$, $1 - \sum_{k \geq j} z_{ik} = \sum_{k < j} z_{ik}$.

Remark 1. Observe that when variables z_{i1} , $i \in P$ are not defined and the permutation is described by means of inequalities (3) and (4), then constraints (11d⁰), (11d) and (11d') must consider separately the case $j = 1$ from the case $j \in P, j > 1$. In particular, the case $j = 1$ reduces to

$$C^i x \leq \theta_1 \quad i \in P, \tag{12}$$

since the first position has always a value greater than or equal to any cost function.

Let $\Omega^z = \{(x, z, \theta) \text{ satisfying constraints (11b), (11c), (11d), (11e), (11f)}\}$ denote the domain obtained from Ω_0^z when constraints (11d⁰) are replaced by constraints (11d).

Property 1. $\Omega_0^z = \Omega^z$.

Proof. It is clear that $\Omega_0^z \supseteq \Omega^z$, since for $i, j \in P$ given, the right hand side of the associated constraint (11d) is smaller than or equal to that of constraint (11d⁰).

To prove that $\Omega_0^z \subseteq \Omega^z$ also holds let $(x, z, \theta) \in \Omega_0^z$ and we show that (x, z, θ) satisfies constraints (11d). For $i, j \in P$ given, we distinguish two cases:

- If $z_{ij} = 1$ then (11d) holds for this pair of indices.
- If $z_{ij} = 0$ then by (11c), there must exist $j' \in P, j' \neq j$, such that $z_{ij'} = 1$. If $j' < j$, then $\sum_{k \geq j} z_{ik} = z_{ij} = 0$, and (11d) holds for the pair of indices i, j . Otherwise, if $j' > j$, then $\sum_{k \geq j} z_{ik} = z_{ij'} = 1$ so the right hand side of constraint (11d) for the pair i, j takes the value θ_j . Now constraint (11d⁰) for the pair of indices i, j' implies that $C^i x \leq \theta_{j'}$. By constraints (11e), we also have $\theta_j \geq \theta_{j'}$ and thus (11d) also holds for the pair of indices i, j . \square

Remark 2. Since $\Omega_0^z = \Omega^z$, an equivalent formulation for the OWAP is

$$F^z : V = \min \sum_{j \in P} \omega_j \theta_j$$

s.t. $(x, z, \theta) \in \Omega^z$.

Formulation F^z can be preferred to formulation F_0^z for solving an OWAP, since it may provide tighter linear programming bounds, given that, for fractional vectors z satisfying constraints (11b)–(11c), constraints (11d⁰) are dominated by constraints (11d).

In the search for optimal solutions to the OWAP any formulation whose optimal solution set coincides with that of the OWAP can be of interest. Such formulations could be preferred because they use fewer variables or constraints, or because their feasible domain has a structure which is easier to explore. Next we present three such formulations. All of them can be seen as relaxations of formulation F^z in the sense that their feasible domains contain Ω^z . However, all of them are valid formulations for the OWAP since they preserve the set of optimal solutions of F^z , i.e. their set of optimal solutions coincides with that of F^z . First we prove a property of optimal solutions.

Lemma 2. Let $(x^*, z^*, \theta^*) \in \Omega^z$ be an optimal solution to F^z . Then for each $j \in P$ there exists $i \in P$ with $\theta_j^* = C^i x^*$.

Proof. Let \tilde{x} be a feasible solution in Q . Then, there exists a positions permutation σ that sorts the cost function values in a non-increasing order. That is, $C^{\sigma_j} \tilde{x} \geq C^{\sigma_{j+1}} \tilde{x}, \forall j \in P \setminus \{p\}$. Therefore, we can set $\tilde{z} = (z_{\sigma_j, j})_{j \in P}$ and $\theta = (C^{\sigma_j} \tilde{x})_{j \in P}$. Since this is true for each $x \in Q$, it is true in particular for x^* . \square

From the above lemma, we observe that Ω^z is always non empty, provided that Q is non empty.

Let $\Omega_{R1}^z = \{(x, z, \theta) \text{ satisfying constraints (11b), (11c), (11d), (11f)}\}$, i.e. Ω_{R1}^z is the relaxation of the domain Ω^z once the set of constraints (11e) is removed. Next, consider the formulation

$$F_{R1}^z : V = \min \sum_{j \in P} \omega_j \theta_j$$

s.t. $(x, z, \theta) \in \Omega_{R1}^z$.

Property 3. Every optimal solution to F_{R1}^z is also optimal to F^z .

Proof. Since $\Omega^z \subseteq \Omega_{R1}^z$ it is enough to prove that any optimal solution to F_{R1}^z is feasible to F^z . Let $(x, z, \theta) \in \Omega_{R1}^z$ be an optimal solution to F_{R1}^z and σ a permutation that sorts the cost function values of x . Let us see that θ verify constraint (11e).

If $i \geq j$ then $C^{\sigma_i} x$ occupies, in the sorted sequence of objective values, a position greater than or equal to the j th. Thus, constraint (11d) implies that $\theta_j \geq \max_{i \geq j} C^{\sigma_i} x$. Since we are minimizing a function which is a linear combination with non-negative weights of the θ variables, it follows that in any optimal solution $\theta_j = \max_{i \geq j} C^{\sigma_i} x$ since, otherwise, the value of θ_j could be decreased to $\max_{i \geq j} C^{\sigma_i} x$, while keeping all other variables values unchanged and without increasing the objective function value. Therefore (11e) holds since, otherwise, there would exist j' such that $\theta_{j'+1} > \theta_{j'} \Leftrightarrow \max_{i \geq j'+1} C^{\sigma_i} x > \max_{i \geq j'} C^{\sigma_i} x$, which is not possible. \square

Consider now $\Omega_{R2}^z = \{(x, z, \theta) \text{ satisfying constraints (11b), (11d), (11f)}\}$, i.e., Ω_{R2}^z is the relaxation of the domain Ω_{R1}^z once the set of constraints (11c) is removed. Next, consider the formulation

$$F_{R2}^z : V = \min \sum_{j \in P} \omega_j \theta_j$$

s.t. $(x, z, \theta) \in \Omega_{R2}^z$.

Property 4. Every optimal solution to F_{R2}^z is also optimal to F^z .

Proof. Since $\Omega^z \subseteq \Omega_{R2}^z$ it is enough to prove that any optimal solution to F_{R2}^z is feasible to F^z . Let (x, z, θ) be an optimal solution to F_{R2}^z . If (x, z, θ) is optimal to F_{R1}^z then, by using Property 3, (x, z, θ) is also optimal to F^z . Thus, to prove that (x, z, θ) is optimal to F^z , it suffices to prove that (x, z, θ) satisfies inequalities (11c).

We prove first that $\sum_{j \in P} z_{ij} \leq 1$ for all $i \in P$. Using the notation $r_{ij} = \sum_{k \geq j} z_{ik}$, for all $i, j \in P$, constraints (11d) can be rewritten as

$$C^i x \leq \theta_j + M(1 - r_{ij}) \Leftrightarrow \theta_j \geq C^i x + M(r_{ij} - 1).$$

Therefore, for all $j \in P$,

$$\theta_j = \max_{i \in P} \{C^i x + M(r_{ij} - 1)\}.$$

Suppose there exists $i' \in P$ with $\sum_{j \in P} z_{i'j} = r > 1$, and let $j' = \arg \max \{r_{i'j} = 2 \mid j \in P\}$. If several indices exist with $\sum_{j \in P} z_{ij} > 1$ we select i' as the one with maximum associated j' .

The criterion for the selection of i' and the definition of j' imply that $r_{i'j'} = 2$ and $r_{ij'} \leq 1$ for all $i \neq i'$.

Therefore, since M is a strict upper bound on the value of any cost function, the actual value of $\theta_{j'}$ is determined by cost function i' , and we have

$$\theta_{j'} = C^{i'} x + M(r_{i'j'} - 1) = C^{i'} x + M.$$

Also, $r_{ij} \geq 2$ for all $j < j'$. Thus, $\theta_j \geq C^i x + M$ for all $j < j'$. Furthermore, $r_{ij} \leq 1$ for all $i \in P, j > j'$, implying that $\theta_j < M$ for all $j > j'$.

Observe, on the other hand, that $\sum_{j \in P} z_{i'j} > 1$ implies that there exists some $i'' \in P, i'' \neq i'$ with $\sum_{j \in P} z_{i''j} = 0$. (Otherwise, adding up all constraints (11b) we get a contradiction.)

Let us now define the solution $(x, \bar{z}, \bar{\theta}) \in \Omega_{R2}^z$ with the same x components as above, where

$$\bar{z}_{ij} = \begin{cases} 0 & \text{if } i = i', \text{ and } j = j' \\ 1 & \text{if } i = i'', \text{ and } j = j' \\ z_{ij} & \text{otherwise.} \end{cases}$$

It is clear that $\sum_{j \in P} \bar{z}_{i'j} = r - 1$, and, $\sum_{k \geq j} \bar{z}_{i'k} = r_{i'j} - 1$, for all $j \leq j'$. It is also clear that $\sum_{j \in P} \bar{z}_{i''j} = 1$, and, $\sum_{k \geq j} \bar{z}_{i''k} = 1$, for all $j \leq j'$, and 0 for $j > j'$. For all other $i \neq i', i''$, it holds that $\sum_{j \in P} \bar{z}_{ij} = \sum_{j \in P} z_{ij}$. Since $\sum_{k \geq j} \bar{z}_{ik} \leq 1$, for all $i \in P$ we now have

$$\bar{\theta}_{j'} = \max_{i \in P} \left\{ C^i x + M \left(\sum_{k \geq j'} \bar{z}_{ik} - 1 \right) \right\} < M \leq C^{i'} x + M = \theta_{j'},$$

and, $\bar{\theta}_j \leq \theta_j$, for all $j \neq j'$.

Therefore, since we are minimizing a linear function with non-negative weights of the θ variables, the objective function value of $(x, \bar{z}, \bar{\theta})$ is smaller than that of (x, z, θ) , contradicting the optimality of (x, z, θ) . Hence, $\sum_{j \in P} z_{ij} \leq 1$ for all $i \in P$.

Let us, finally, see that $\sum_{j \in P} z_{ij} \neq 0$ for all $i \in P$. Assume on the contrary that $\sum_{j \in P} z_{i'j} = 0$ for some $i' \in P$. Then, by adding up all constraints (11b) we get $p = \sum_{j \in P} (\sum_{i \in P} z_{ij}) = \sum_{i \in P} (\sum_{j \in P} z_{ij}) = \sum_{i \in P, i \neq i'} (\sum_{j \in P} z_{ij}) \leq p - 1$, which is impossible. \square

We now consider the inequality version of constraints (11b)

$$\sum_{i \in P} z_{ij} \leq 1 \quad j \in P. \tag{11b_{\leq}}$$

Remark 3. Observe that when inequalities (11b_≤) hold, constraints (11d) are no longer equivalent to (11d').

Let us define the domain $\Omega_{R3}^z = \{(x, z, \theta) \text{ satisfying constraints (11b}_{\leq}), (11d'), (11f)\}$.

It is clear that $\Omega^z \subseteq \Omega_{R3}^z$. However, as we next see, both sets are equivalent for the minimization of the objective (11a) in the sense that they define the same set of optimal solutions. Consider the problem

$$F_{R3}^z V = \min \sum_{j \in P} \omega_j \theta_j$$

s.t. $(x, z, \theta) \in \Omega_{R3}^z$.

Lemma 5. $\Omega_{R2}^z \subseteq \Omega_{R3}^z$.

Proof. We prove that any feasible solution $(x, z, \theta) \in \Omega_{R2}^z$ verifies that $(x, z, \theta) \in \Omega_{R3}^z$. To prove this, it is only necessary to prove that (x, z, θ) verifies (11d'). From (11d) we have that (x, z, θ) verifies

$$\theta_j \geq \max_i \left\{ C^i x - M \left(1 - \sum_{k \geq j} z_{ik} \right) \right\}, \quad j \in P \tag{17}$$

and for (11d'), we have to prove that (x, z, θ) also verifies

$$\theta_j \geq \max_i \left\{ C^i x - M \left(\sum_{k < j} z_{ik} \right) \right\}, \quad j \in P. \tag{18}$$

We distinguish the following cases:

- If $\sum_{k \geq j} z_{i'k} = r > 1$ for some i' then

$$\theta_j \geq C^{i'} x + (r - 1)M \geq \max_i \left\{ C^i x - M \left(\sum_{k < j} z_{ik} \right) \right\}, \tag{19}$$

and the result holds.

- If $\sum_{k \geq j} z_{ik} = 1$ for all $i \in P$ then $\theta_j \geq \max_i \{C^i x\} \geq \max_i \{C^i x - M(\sum_{k < j} z_{ik})\}$ and the result is also proven.
- If $\sum_{k \geq j} z_{i'k} = 0$ for some i' then we distinguish two subcases. If $\sum_{k < j} z_{i'k} \geq 1$ then from (17) we easily get that (18) holds. Otherwise, $\sum_{k \in P} z_{i'k} = 0$ and by (11b) there does exist an i'' such that $\sum_{k \geq j} z_{i''k} = r > 1$. Thus, by using (19), Eq. (18) also holds. \square

Property 6. F^z and F_{R3}^z have the same set of optimal solutions.

Proof. Since $\Omega^z \subset \Omega_{R2}^z$ and $\Omega_{R2}^z \subset \Omega_{R3}^z$ then $\Omega^z \subset \Omega_{R3}^z$ and it is enough to prove that any optimal solution to F_{R3}^z is feasible to F^z . Since the set of optimal solutions of F^z and F_{R2}^z coincides, we only need to prove that any optimal solution of F_{R3}^z is feasible for F_{R2}^z .

To see that any optimal solution (x, z, θ) to F_{R3}^z is feasible to F_{R2}^z , it is enough to see that $(x, z, \theta) \in \Omega_{R2}^z$, i.e. it satisfies inequalities (11b) and (11d).

By a similar argument to the one applied in Property 4, any optimal solution (x, z, θ) of F_{R3}^z satisfies $\sum_{j \in P} z_{ij} = 1$. Therefore, satisfying inequality (11d') implies inequality (11d).

Now (11b) follows directly from (11c) and (11b_≤) since, otherwise, the sum of all constraints (11b_≤) would not coincide with the sum of all constraints (11c).

To see that (x, z, θ) also satisfies (11b), let us suppose w.l.o.g. that there exists exactly one $j' \in P$ such that $\sum_{i \in P} z_{ij'} = 0$. Then, by adding up all constraints (11b_≤) we have $p - 1 \geq \sum_{j \in P} \sum_{i \in P} z_{ij} = \sum_{i \in P} \sum_{j \in P} z_{ij}$. Therefore, there must exist $i' \in P$ such that $\sum_{j \in P} z_{i'j} = 0$. Thus, we observe that we can construct $(x, \bar{z}, \bar{\theta})$, another optimal solution to F_{R3}^z , setting $\bar{z}_{ij} = z_{ij}$, if $i \neq i'$ and $\bar{z}_{i'k} = 1$ for any k . Clearly, $(x, \bar{z}, \bar{\theta})$ is a feasible solution to F_{R3}^z for some suitable $\bar{\theta}$, satisfying in addition

$$C^{i'} x \leq \bar{\theta}_k + M \sum_{\ell < k} z_{i'\ell}, \quad \forall k \in P.$$

Therefore, this inequality allows for any $k \in P$ that $\bar{\theta}_k$ assumes a value smaller than or equal to θ_k , the one associated with the solution (x, z, θ) , and therefore its objective value is at least as good as the previous one. Hence, $(x, \bar{z}, \bar{\theta})$ is also optimal. In addition, values \bar{z} satisfy by construction that $\sum_{i \in P} \bar{z}_{ij'} = \sum_{i \neq i'} z_{ij'} + \bar{z}_{i'j'} = 0 + 1 = 1$. Therefore (11b) holds. \square

We can now relate the domains of the formulations considered so far.

Proposition 7. The following relationships hold

$$\Omega_0^z \equiv \Omega^z \subsetneq \Omega_{R1}^z \subsetneq \Omega_{R2}^z \subsetneq \Omega_{R3}^z.$$

Proof. • $\Omega^z \subsetneq \Omega_{R1}^z$: Every feasible solution in Ω^z verifies inequalities of Ω_{R1}^z . However, a feasible solution in Ω_{R1}^z with $\theta_j \leq \theta_{j+1}$ for some $j \in P$ is not feasible in Ω^z .
 • $\Omega_{R1}^z \subsetneq \Omega_{R2}^z$: Every feasible solution in Ω_{R1}^z verifies the inequalities of Ω_{R2}^z . However, a feasible solution in Ω_{R2}^z where for some $i \in P$, $z_{ij} = 1$, for all $j \in P$ is not feasible in Ω_{R1}^z .
 • $\Omega_{R2}^z \subsetneq \Omega_{R3}^z$: Every feasible solution in Ω_{R2}^z verifies the inequalities of Ω_{R3}^z . However, a feasible solution in Ω_{R3}^z with $z_{ij} = 0$, $i, j \in P$ is not feasible in Ω_{R2}^z . \square

Proposition 8. The dimension of Ω_0^z is $p^2 - p + 1 + \dim(Q)$.

Proof. Suppose $Q \subseteq \mathbb{R}^n$. Then, Ω_0^z is embedded in a space of dimension $p^2 + p + n$. Furthermore, since there are $2p - 1$ linearly independent equations in (11b) and (11c) and the dimension of Q does not depend on relations (11b)–(11e), then the dimension of (11b)–(11f) is at most $p^2 - p + 1 + \dim(Q)$. Denote $q = \dim(Q)$ and $\rho = p^2 - 2p + 1$. Next, we show that there exist $q + \rho + p + 1$ (equal to $p^2 - p + 2 + \dim(Q)$) affinely independent points in Ω_0^z and consequently, the dimension of Ω_0^z is $p^2 - p + 1 + \dim(Q)$.

Let $v = (v_j)_{j \in P}$ where $v_j = M + p - j + 1$ for $M > 0$ and sufficiently large. Denoting by $\mathbf{e}^j \in \mathbb{R}^p$ the j th vector of the canonical basis in \mathbb{R}^p and $0 < \varepsilon < 1$, let $\theta^j = \{v + \varepsilon \mathbf{e}^j, j \in P\}$. Moreover, let $\theta^{p+1} = (M, \dots, M)'$. We observe that the vectors $\theta^j, j = 1, \dots, p + 1$ are affinely independent and each one of them satisfies inequalities (11e).

Next, since $\dim(Q) = q$, we take $q + 1$ arbitrary affinely independent vectors $x^i \in Q, i = 1, \dots, q + 1$. Furthermore, let $z^k \in \{0, 1\}^{p^2} k = 1, \dots, \rho + 1$, be $\rho + 1$ affinely independent vectors satisfying (11b) and (11c). Note that the latter is always possible since there are p^2 degrees of freedom for the coordinates of z variables and only $2p$ equations being one of them linearly dependent of the others.

Now, we prove that any point of the form $((x^i)', (z^k)', (\theta^l)')' i = 1, \dots, q + 1, k = 1, \dots, \rho + 1, l = 1, \dots, p + 1$ satisfies (11b)–(11e). Indeed, by construction the first block of coordinates defines a point in Q , the second block satisfies (11b) and (11c) and the third one (11e). Thus, it remains to prove that such a generic point also satisfies (11d) as follows:

$$C^i x^i \leq M \leq M + p - j + 1 \leq \theta_j^l + M(1 - z_{ij}^k), \quad \forall i, j.$$

Consider the $q + \rho + p$ points defined as the column vectors of the matrix $A = (A^1 | A^2 | A^3)$ where

$$A^1 = \begin{pmatrix} x^1 & x^2 & \dots & x^q \\ z^2 & z^1 & \dots & z^1 \\ \theta^2 & \theta^1 & \dots & \theta^1 \end{pmatrix}, \quad A^2 = \begin{pmatrix} x^1 & x^1 & \dots & x^1 \\ z^1 & z^2 & \dots & z^\rho \\ \theta^2 & \theta^1 & \dots & \theta^1 \end{pmatrix}, \quad A^3 = \begin{pmatrix} x^1 & x^1 & x^1 & \dots & x^1 \\ z^1 & z^3 & z^1 & \dots & z^1 \\ \theta^1 & \theta^2 & \theta^3 & \dots & \theta^p \end{pmatrix}.$$

By construction, each submatrix A^i has its column vectors linearly independent from one another since the i th block is formed by linearly independent vectors. Next, clearly each column vector of A^1 is linearly independent from those of A^2 and A^3 and each column vector of A^2 is linearly independent from those of A^3 . Therefore, the rank of A is $q + \rho + p = q + p^2 - p + 1$.

Finally, the column vectors of A are linearly independent and feasible points of (11b)–(11e). In addition, we can easily construct another feasible point, different from those considered previously and affinely independent of all of them, namely $((x^{q+1})', (z^{\rho+1})', (\theta^{p+1})')$. Hence the dimension of Ω^z is $q + \rho + p = q + p^2 - p + 1$. \square

Proposition 9. The following inequalities define facets in Ω_0^z :

$$C^i x \leq \theta_p + M(1 - z_{ip}) \quad i \in P \tag{20}$$

$$\theta_j \geq \theta_{j+1} \quad j \in P : j < p. \tag{21}$$

Proof. (20) is a facet defining inequality:

We prove that for each $i' \in P$ there exist $\dim(\Omega_0^z) = p^2 - p + \dim(Q) + 1$ affinely independent points of Ω_0^z that verify $C^{i'} x = \theta_p + M(1 - z_{i'p})$.

As in the proof of the above proposition, we take $q + 1$ arbitrary affinely independent points $x^i, i = 1, \dots, q + 1$ in Q . Furthermore, let $z^k \in \{0, 1\}^{p^2} k = 1, \dots, \rho$, be ρ affinely independent points (recall that $\rho = p^2 - 2p + 1$) satisfying (11b), (11c) and $z_{i'p} = 1$. Note that the latter is always possible since there are p^2 degrees of freedom for the coordinates of z variables and $2p$ non redundant equations ($2p - 1$ as in the case above and $z_{i'p} = 1$).

Let $v^l = (v_j^l)_{j \in P}$ where $v_j^l = C^{i'} x^l + M + p - j$ if $j < p$ and $v_p^l = C^{i'} x^l$ for $M > 0$ and sufficiently large. Denoting by $\mathbf{e}^j \in \mathbb{R}^p$ the j th vector of the canonical basis in \mathbb{R}^p and $0 < \varepsilon < 1$, let $\bar{\theta}^{lj} = \{v^l + \varepsilon \mathbf{e}^j, j \in P\}$ if $j < p$ and $\bar{\theta}^{lp} = v^l, \bar{\theta}^{l,p+1} = (C^{i'} x^l + M, \dots, C^{i'} x^l + M, C^{i'} x^l)'$. We observe that for each l fixed, the vectors $\bar{\theta}^{lj} j = 1, \dots, p + 1$ are affinely independent and each one of them satisfies inequalities (11e).

Now, we prove that any point of the form $((x^l)', (z^k)', (\theta^{lj})')' k = 1, \dots, \rho, j = 1, \dots, p + 1$ satisfies (11b)–(11e) and $z_{i'p}^k = 1$. Indeed, by construction the first block of coordinates defines a point in Q , the second block satisfies (11b), (11c) and $z_{i'p} = 1$, and the third one (11e). Thus, it remains to prove that such a generic point also satisfies (11d). We distinguish two cases:

- If $j < p$ then

$$C^i x^l \leq C^{i'} x^l + M + p - j + 1 + M = C^{i'} x^l + M + p - j + M(1 - z_{ij}^k) = \bar{\theta}^{lj} + M(1 - z_{ij}^k), \quad \forall i.$$

- If $j = p$ we have that

$$C^i x^l \leq C^{i'} x^l + M = C^{i'} x^l + M(1 - z_{ip}^k), \quad \forall i \neq i',$$

$$C^{i'} x^l \leq C^{i'} x^l = C^{i'} x^l + M(1 - z_{i'p}^k), \quad \text{otherwise. (Recall that } z_{i'p}^k = 1.)$$

Consider the $q + \rho - 1 + p$ points defined as the column vectors of the matrix $\bar{A} = (\bar{A}^1 | \bar{A}^2 | \bar{A}^3)$ where

$$\bar{A}^1 = \begin{pmatrix} x^1 & x^2 & \dots & x^q \\ z^2 & z^1 & \dots & z^1 \\ \bar{\theta}^{11} & \bar{\theta}^{21} & \dots & \bar{\theta}^{q1} \end{pmatrix}, \quad \bar{A}^2 = \begin{pmatrix} x^1 & x^1 & \dots & x^1 \\ z^1 & z^2 & \dots & z^{\rho-1} \\ \bar{\theta}^{12} & \bar{\theta}^{11} & \dots & \bar{\theta}^{11} \end{pmatrix}, \quad \bar{A}^3 = \begin{pmatrix} x^1 & x^1 & x^1 & \dots & x^1 \\ z^1 & z^3 & z^1 & \dots & z^1 \\ \bar{\theta}^{11} & \bar{\theta}^{12} & \bar{\theta}^{13} & \dots & \bar{\theta}^{1p} \end{pmatrix}.$$

By construction, each submatrix \bar{A}^i has its column vectors linearly independent from one another since the i th block is formed by linearly independent vectors. Next, clearly each column vector of \bar{A}^1 is linearly independent from those of \bar{A}^2 and \bar{A}^3 and each column vector of \bar{A}^2 is linearly independent from those of \bar{A}^3 . Therefore, the rank of A is $q + \rho - 1 + p = q + p^2 - p$.

Finally, the column vectors of A together with the point $((x^{q+1})', (z^{\rho+1})', (\theta^{q+1,j})')$ are feasible points of (11b)–(11e) that satisfy $C^i x = \theta_p + M(1 - z_{ip})$; and this last vector is clearly affinely independent from the those in \bar{A} , therefore (20) is a facet defining inequality for Ω^z .

(21) is a facet defining inequality:

In order to prove that for each $j' \in P \setminus \{p\}$ there exist $\dim(\Omega_0^z) = p^2 - p + \dim(Q) + 1$ affinely independent points of Ω_0^z that verify $\theta_j = \theta_{j+1}$, we can proceed analogously as before considering $v = (v_j)_{j=1}^p$, where $v_j = M + p - j + 1$ if $j \neq j' + 1$ and $v_{j'+1} = M + p - j' + 2$ and the points $\hat{\theta}^j = \{v + \varepsilon(\mathbf{e}^j + \mathbf{e}^{j'+1}), j \in P \setminus \{p\}\}$. In addition, we take $\hat{\theta}^p = (M, \dots, M)'$. We observe that the vectors $\hat{\theta}^j, j = 1, \dots, p$ are affinely independent and each one of them satisfies $\hat{\theta}_j^j = \hat{\theta}_{j+1}^j$.

Any point of the form $((x^i)', (z^k)', (\hat{\theta}^l)')$ $i = 1, \dots, q + 1, k = 1, \dots, \rho + 1, l = 1, \dots, p$ satisfies (11b)–(11e) and $\hat{\theta}_j^l = \hat{\theta}_{j+1}^l$.

Consider the $q + \rho + p - 1$ points defined as the column vectors of the matrix $\hat{A} = (\hat{A}^1 | \hat{A}^2 | \hat{A}^3)$ where

$$\hat{A}^1 = \begin{pmatrix} x^1 & x^2 & \dots & x^q \\ z^2 & z^1 & \dots & z^1 \\ \hat{\theta}^2 & \hat{\theta}^1 & \dots & \hat{\theta}^1 \end{pmatrix}, \quad \hat{A}^2 = \begin{pmatrix} x^1 & x^1 & \dots & x^1 \\ z^1 & z^2 & \dots & z^\rho \\ \hat{\theta}^2 & \hat{\theta}^1 & \dots & \hat{\theta}^1 \end{pmatrix}, \quad \hat{A}^3 = \begin{pmatrix} x^1 & x^1 & x^1 & \dots & x^1 \\ z^1 & z^3 & z^1 & \dots & z^1 \\ \hat{\theta}^1 & \hat{\theta}^2 & \hat{\theta}^3 & \dots & \hat{\theta}^{p-1} \end{pmatrix}.$$

By construction, each submatrix \hat{A}^i has its column vectors linearly independent from one another since the i th block is formed by linearly independent vectors. Next, clearly each column vector of \hat{A}^1 is linearly independent from those of \hat{A}^2 and \hat{A}^3 and each column vector of \hat{A}^2 is linearly independent from those of \hat{A}^3 . Therefore, the rank of \hat{A} is $q + \rho + p - 1 = q + p^2 - p$.

Finally, the column vectors of \hat{A} are linearly independent and are also feasible points of (11b)–(11e) that satisfy $\theta_j = \theta_{j+1}$. Next, we can easily add a new feasible point, for instance $((x^{q+1})', (z^{\rho+1})', (\hat{\theta}^p)')$ that also satisfies $\theta_j = \theta_{j+1}$ and that is clearly affinely independent from the those in \hat{A} . Hence, (21) is a facet defining inequality for Ω^z . \square

Table 6 summarizes the previous proposed formulations. Formulas included on each formulation have been checked (\checkmark) whereas those not appearing are marked with a dot (\cdot).

3.3. OWAP formulations with variables for the values of cost functions occupying specific sorted positions

Another OWAP formulation can be obtained by defining an additional set of continuous variables $y = (y_{ij})_{i,j \in P} \in \mathbb{R}^{p \times p}$, where y_{ij} denotes the value of cost function i if it occupies the j th position in the ordering. The formulation is as follows:

$$F_0^{zy} : V = \min \sum_{j \in P} \omega_j \sum_{i \in P} y_{ij} \tag{22a}$$

$$\text{s.t.} \quad \sum_{i \in P} z_{ij} = 1 \quad j \in P \tag{22b}$$

$$\sum_{j \in P} z_{ij} = 1 \quad i \in P \tag{22c}$$

$$C^i x \leq \sum_{i' \in P} y_{i'j} + M(1 - z_{ij}) \quad i, j \in P \tag{22d^0}$$

$$\sum_{i \in P} y_{ij} \geq \sum_{i \in P} y_{ij+1} \quad j \in P : j < p \tag{22e}$$

$$x \in Q, z \in \{0, 1\}^{p \times p}. \tag{22f}$$

Next we study some properties of formulation F_0^{zy} and relate it to the OWAP formulations presented above. Denote by Ω_0^{zy} the domain of Problem F_0^{zy} . Consider first, for any $M > 0$ sufficiently large, the following set of inequalities

$$y_{ij} \leq Mz_{ij}, \quad i, j \in P. \tag{22g}$$

Table 6
Summary of the proposed formulations for the OWAP.

	F_0^z	F^z	F_{R1}^z	F_{R2}^z	F_{R3}^z
$\min \sum_{j \in P} \omega_j \theta_j$	✓	✓	✓	✓	✓
$\sum_{i \in P} z_{ij} = 1, j \in P$	✓	✓	✓	✓	·
$\sum_{j \in P} z_{ij} = 1, i \in P$	✓	✓	✓	·	·
$\sum_{i \in P} z_{ij} \leq 1, j \in P$	·	·	·	·	✓
$C^i x \leq \theta_j + M(1 - z_{ij}), i, j \in P$	✓	·	·	·	·
$C^i x \leq \theta_j + M(1 - \sum_{k \geq j} z_{ik}), i, j \in P$	·	✓	✓	✓	·
$C^i x \leq \theta_j + M \sum_{k < j} z_{ik}, i, j \in P$	·	·	·	·	✓
$\theta_j \geq \theta_{j+1}, j \in P : j < p$	✓	✓	·	·	·
$x \in Q, z \in \{0, 1\}^{p \times p}$	✓	✓	✓	✓	✓

Property 10. There is an optimal solution to F_0^{zy} for which constraints (22g) hold.

Proof. Observe that constraints (22d⁰) imply that $\sum_{k \in P} y_{kj} \geq C^i x$ for all $i, j \in P$ with $z_{ij} = 1$. Since constraints (22b) indicate that for $j \in P$ fixed there exists a unique index, say $i(j)$ with $z_{i(j),j} = 1$, the above condition reduces to $\sum_{k \in P} y_{kj} \geq C^{i(j)} x$, for all $j \in P$. Because of the non-negativity of the cost coefficients, we can thus deduce that an optimal solution exists to F_0^{zy} in which

$$\sum_{k \in P} y_{kj} = C^{i(j)} x, \quad \text{for all } j \in P. \tag{23}$$

Let now $(x, y, z) \in \Omega_0^{zy}$ be such an optimal solution, and suppose it violates some constraint (22g). That is, there exist $i', j' \in P$ with $y_{i'j'} > Mz_{i'j'}$. Hence, $\sum_{i \in P} y_{ij'} > Mz_{i'j'}$, contradicting (23) unless $z_{i'j'} = 0$. In other words, $i(j') \neq i'$.

Consider now the solution (x, \bar{y}, z) , with the same x and z values as before where \bar{y} is defined as follows:

$$\bar{y}_{ij} = \begin{cases} 0 & \text{if } i = i', \text{ and } j = j' \\ y_{i(j'),j'} + y_{i'j'} & \text{if } i = i(j'), \text{ and } j = j' \\ y_{ij} & \text{otherwise.} \end{cases}$$

Indeed $(x, \bar{y}, z) \in \Omega_0^{zy}$, as it is immediate to check that it satisfies constraints (22b)–(22f). Furthermore, by construction, it satisfies the constraint (22g) associated with i', j' . Finally, note that it is optimal to F_0^{zy} , since $\sum_{i \in P} \bar{y}_{ij} = \sum_{i \in P} y_{ij}$, for all $j \in P$. □

Note that if there is $j \in P$ with $\omega_j = 0$ then it is possible to have optimal solutions to F_0^{zy} that do not satisfy constraints (22g). However, because of Property 10, constraints (22g) can be useful to restrict the domain where optimal solutions are sought. Let

$$\Omega^{GS'} = \{(x, y, z, \theta) \text{ satisfying constraints (22b), (22c), (22d}^0), (22e), (22f), (22g)\}.$$

Then, a different formulation that also ensures to obtain an optimal solutions to F_0^{zy} is:

$$F^{GS'} : V = \min \sum_{j \in P} \omega_j \theta_j$$

s.t. $(x, y, z, \theta) \in \Omega^{GS'}$.

Formulation $F^{GS'}$ is closely related to the formulation used in Galand and Spanjaard [7] for modeling the minimum cost spanning tree OWAP. In their formulation they operate on a domain which is like $\Omega^{GS'}$ except that constraints (22d⁰) have been substituted by constraints

$$\sum_{j \in P} y_{ij} = C^i x \quad i \in P. \tag{22h}$$

Let $\Omega^{GS} = \{(x, y, z, \theta) \text{ satisfying constraints (22b), (22c), (22e), (22f), (22g), (22h)}\}$, denote the domain used in [7]. Then, it is straightforward to conclude the following.

Property 11. The domains Ω^{GS} and $\Omega^{GS'}$ satisfy $\Omega^{GS} \subseteq \Omega^{GS'}$. Moreover, if (x^*, y^*, z^*) is an optimal solution of $F^{GS'}$ then it is also optimal for F^{GS} and conversely.

We can also relate F_0^{zy} with F_0^z and its variations. In particular, because of the relationship

$$\theta_j = \sum_{i \in P} y_{ij}, \quad j \in P \tag{25}$$

we have:

Table 7
Summary of the proposed formulations for the OWAP.

	F_0^{zy}	F^{zy}	F_{R1}^{zy}	F_{R2}^{zy}	F_{R3}^{zy}
$\min \sum_{j \in P} \omega_j \sum_{i \in P} y_{ij}$	✓	✓	✓	✓	✓
$\sum_{i \in P} z_{ij} = 1, j \in P$	✓	✓	✓	✓	·
$\sum_{j \in P} z_{ij} = 1, i \in P$	✓	✓	✓	·	·
$\sum_{i \in P} z_{ij} \leq 1, j \in P$	·	·	·	·	✓
$C^i x \leq \sum_{j \in P} y_{ij} + M(1 - z_{ij}), i, j \in P$	✓	·	·	·	·
$C^i x \leq \sum_{j \in P} y_{ij} + M(1 - \sum_{k \geq j} z_{ik}), i, j \in P$	·	✓	✓	✓	·
$C^i x \leq \sum_{j \in P} y_{ij} + M \sum_{k < j} z_{ik}, i, j \in P$	·	·	·	·	✓
$\sum_{i \in P} y_{ij} \geq \sum_{i \in P} y_{ij+1}, j \in P : j < p$	✓	✓	·	·	·
$x \in Q, z \in \{0, 1\}^{p \times p}$	✓	✓	✓	✓	✓

Property 12. For each optimal solution to F_0^{zy} , $(x^*, y^*, z^*, \theta^*)$, there exists (x^*, z^*, θ^*) optimal solution for F_0^z and conversely. Moreover, $\sum_{j \in P} w_j \sum_{i \in P} y_{ij}^* = \sum_{j \in P} w_j \theta_j^*$.

By the above result, we can derive variations of F^{zy} similar to the ones obtained for F^z with similar properties. These constructions are straightforward and therefore are left for the interested readers.

Table 7 summarizes the formulations proposed in this subsection that can be derived from those of Section 3.2. Constraints included in each formulation have been checked (✓) whereas those not appearing are marked with a dot (·).

3.4. Using variables defining relative positions of sorted cost function values

We close this section with another formulation which uses decision variables defining the relative positions of the sorted cost function values. As we have seen in Section 3.1 it is possible to describe permutations with variables representing the relative positions of the sorted values. Next we use such variables to obtain formulations for the OWAP.

For $i, j \in P$, consider the binary variable s_{ij} , $i, j \in P$ as

$$s_{ij} = \begin{cases} 1 & \text{if cost function } i \text{ is placed before position } j \text{ in the ordering,} \\ 0 & \text{otherwise.} \end{cases}$$

As we have seen in Section 3.1, for all $i, j \in P$, $s_{ij} = 1 - \sum_{k \geq j} z_{ik}$, $i, j \in P$. Therefore, variables z and s are related by means of

$$z_{ij} = \begin{cases} s_{ij+1} - s_{ij} & i \in P, j = 1, \dots, p - 1 \\ 1 - s_{ij} & i \in P, j = p. \end{cases} \tag{26}$$

Thus, we can reformulate the OWAP in the new space of the s variables as

$$F^s : V = \min \sum_{j \in P} \omega_j \theta_j \tag{27a}$$

$$\text{s.t. } \sum_{i \in P} s_{ij} = j - 1 \quad j \in P \tag{27b}$$

$$s_{ij+1} - s_{ij} \geq 0 \quad i, j \in P : j < p \tag{27c}$$

$$C^i x \leq \theta_j + Ms_{ij} \quad i, j \in P \tag{27d}$$

$$\theta_j \geq \theta_{j+1} \quad j \in P : j < p \tag{27e}$$

$$x \in Q, s \in \{0, 1\}^{p \times p}. \tag{27f}$$

Since F^s is obtained from F^z by a change of variable and there is a one to one correspondence between feasible solutions, we can state the following result. Let Ω^s be the feasible region of Problem F^s .

Property 13. For each solution $(x, s, \theta) \in \Omega^s$ there exists $(x, z, \theta) \in \Omega^z$ with equal objective value and conversely.

By analogy with the notation used in Section 3.2 let us define the following domains and problems related to F^s :

$$F_{R1}^s : V = \min \sum_{j \in P} \omega_j \theta_j$$

$$\text{s.t. } (x, s, \theta) \in \Omega_{R1}^s$$

with $\Omega_{R1}^s = \{(x, s, \theta) \text{ satisfying constraints (27b), (27c), (27d), (27f)}\}$.

$$F_{R2}^s : V = \min \sum_{j \in P} \omega_j \theta_j$$

s.t. $(x, s, \theta) \in \Omega_{R2}^s$

with $\Omega_{R2}^s = \{(x, s, \theta) \text{ satisfying constraints (27b), (27d), (27f)}\}$.

$$F_{R3}^z : V = \min \sum_{j \in P} \omega_j \theta_j$$

s.t. $(x, z, \theta) \in \Omega_{R3}^z$

with $\Omega_{R3}^s = \{(x, s, \theta) \text{ satisfying constraints (27b}_{\leq}), (27d), (27f)\}$, where $(27b_{\leq})$ are the inequality version of constraints (27b). That is,

$$\sum_{i \in P} s_{ij} \leq j - 1 \quad j \in P. \tag{27b_{\leq}}$$

Property 14. *The following relationships hold.*

1. Every optimal solution to F_{R1}^s is optimal to F^s and conversely.
2. Every optimal solution to F_{R2}^s is optimal to F^s and conversely.
3. Every optimal solution to F_{R3}^z is optimal to F^s and conversely.
4. $\Omega^s \subsetneq \Omega_{R1}^s \subsetneq \Omega_{R2}^s \subsetneq \Omega_{R3}^s$.

Proof. The proofs of the above statements follow directly from the relationship that links variables z and s , namely (9) and (10). Specifically, statement 1 follows from Property 3, statement 2 from Property 4, statement 3 from Property 6 and statement 4 from Proposition 7. □

3.5. Formulations summary

Table 8 summarizes the previous formulations in this subsection. Constraints included in each formulation have been checked (✓) whereas those not appearing are marked with a dot (·).

4. Valid inequalities and reinforcements for the OWAP formulation

4.1. Valid inequalities for the (OWAP) formulation

In this section we derive different valid inequalities for all the formulations presented in previous sections. For the sake of simplicity, we present all inequalities for the formulations developed in Section 3.2. However, all these inequalities can be easily adapted to the remaining formulations just by means of the substitutions explained by Eqs. (10) and (25).

- *Constraints related to bounds of cost function values.* Let $l_i (u_i)$ denote the minimum (maximum) objective value relative to cost function $i \in P$, respectively. It is clear that $l_i (u_i)$ are valid lower (upper) bounds on the value of objective i , independently of the position of cost function i in the ordering. Therefore we obtain the following two sets of constraints which are valid for the OWAP:

$$l_i \leq C^i x \leq u_i \quad i \in P. \tag{31}$$

- *Constraints related to bounds of values in specific positions.* Let $l_j^{\pi} (u_j^{\pi})$ denote the j th lowest (largest) value of $l_i (u_i)$. Then, $l_j^{\pi} (u_j^{\pi})$ is a valid lower (upper) bound of the objective function sorted in position j , that is

$$l_j^{\pi} \leq \theta_j \leq u_j^{\pi} \quad j \in P. \tag{32}$$

- *Constraints related to bounds of cost function values in specific positions.* Let l_{ij} and u_{ij} denote valid lower and upper bounds on the value of objective i if it occupies position j , respectively. Then, lower and upper bounds on the value of objective i are

$$\min_{j \in P} l_{ij} \leq C^i x \leq \max_{j \in P} u_{ij} \quad i \in P. \tag{33}$$

Analogously to (32), we can sort the j th lowest (largest) value of $\min_{j \in P} l_{ij}$ obtaining the following inequality

$$\min_{i \in P} l_{ij} \leq \theta_j \leq \max_{i \in P} u_{ij} \quad j \in P. \tag{34}$$

- There are also different bounds on the value of the cost function i and the value of the cost function sorted in position j :

$$\sum_{j \in P} \max\{l_i, l_j^{\pi}\} z_{ij} \leq C^i x \leq \sum_{j \in P} \min\{u_i, u_j^{\pi}\} z_{ij} \quad i \in P \tag{35}$$

$$\sum_{i \in P} \max\{l_i, l_j^{\pi}\} z_{ij} \leq \theta_j \leq \sum_{i \in P} \min\{u_i, u_j^{\pi}\} z_{ij} \quad j \in P. \tag{36}$$

Table 8
Summary of the proposed formulations for the OWAP.

	F^S	F_{R1}^S	F_{R2}^S	F_{R3}^S
$\min \sum_{j \in P} \omega_j \theta_j$	✓	✓	✓	✓
$\sum_{i \in P} s_{ij} = j - 1, j \in P$	✓	✓	✓	·
$s_{j+1} - s_{ij} \geq 0, i, j \in P : j < p$	✓	✓	·	·
$\sum_{i \in P} s_{ij} \leq j - 1, j \in P$	·	·	·	✓
$C^i x \leq \theta_j + Ms_{ij}, i, j \in P$	✓	✓	✓	✓
$\theta_j \geq \theta_{j+1}, j \in P : j < p$	✓	·	·	·
$x \in Q, z \in \{0, 1\}^{p \times p}$	✓	✓	✓	✓

- The inclusion of the following constraint also allows to consider, in the original formulations in Section 3, weights $\omega \in \mathbb{R}$ that, consequently, could take both negative and positive values.

$$\theta_j \leq \max_{i \in P} \{u_{ij}, C^i x + M(1 - z_{ij})\} \quad i, j \in P. \tag{37}$$

- *Constraints related to positions in the ordering.* Constraints (38) impose that the position values are ordered in a non-increasing order.

$$\theta_j \geq \theta_{j+1} \quad j \in P \setminus \{p\}. \tag{38}$$

- *Constraints related to subsets of cost functions.* Next, we observe that for any subset $I \subseteq P$, of size $k = 1, \dots, p$

$$\sum_{i \in I} C^i x \leq \sum_{j=1}^k \theta_j \quad I \subseteq P. \tag{39}$$

In particular, we consider the cases when $I = \{i\}, I = \{i, i' \in P\}, I = P \setminus \{i\}$ and $I = P$.

4.2. Valid inequalities for the (OWAP2) formulation

Note first that all previous inequalities from Section 4.1 can be applied to the two-index formulation of the OWAP substituting $\theta_j = \sum_{i \in P} y_{ij}$. Additionally, the following inequalities provide a reinforcement to the formulations using y variables:

- The following inequality combined with (22e) improves considerably the LP relaxation of the OWAP

$$\sum_{k \in P} y_{ik} = C^i x \quad i \in P. \tag{40}$$

- Constraint (22e) can be disaggregated by $j \in P$ as:

$$y_{ij} \leq \sum_{i' \in P} y_{i'j} + \min\{u_i, u_j^\pi\} \left(1 - \sum_{k \geq j} z_{ik}\right) \quad i, j \in P. \tag{41}$$

- We can also establish a lower bound on the value of cost function $i \in P$ if it is ordered in position $j \in P$ by relating the x, y and z variables as follows:

$$C^i x \leq y_{ij} + u_j^\pi (1 - z_{ij}) \quad i, j \in P. \tag{42}$$

Observe that, for i, j fixed, the above constraint imposes a lower bound on the value y_{ij} only when cost function $i \in P$ is ordered in position $j \in P$, and becomes inactive otherwise.

- We can also relate the values of two different cost functions between them, depending on their positions. In particular,

$$\sum_{k \geq j+1} y_{ik} \leq y_{i'j} + u_i (1 - z_{i'j} - z_{ij}) \quad i, i', j \in P, i \neq i', j \neq p. \tag{43}$$

For i, i', j fixed, constraint (43) establishes that when cost function i' occupies position j , its value cannot be smaller than that of cost function i , provided that cost function i is ordered after j . Observe that the constraint becomes inactive when i is ordered before j or is in j position (since in this case $\sum_{k \geq j+1} y_{ik} = 0$) and when i' does not occupy position j .

- A better effectiveness of the previous inequalities can be obtained by means of

$$y_{ij+1} \leq y_{i'j} + (1 - z_{ij+1})u_{ij+1} + (1 - z_{i'j})u_{i'j} \quad i, i', j \in P, i \neq i', j \neq p \tag{44}$$

which can be further reinforced to

$$y_{ij+1} \leq y_{i'j} + (1 - z_{ij+1}) \min\{u_i, u_{j+1}^\pi\} + (1 - z_{i'j}) \min\{u_{i'}, u_j^\pi\} \quad i, i', j \in P, i \neq i', j \neq p. \tag{45}$$

4.3. Lower and upper bounds: elimination tests

Several of the inequalities presented above use valid lower and upper bounds on the values of the different cost functions, l_i and u_i , respectively. As mentioned above, the minimum and the maximum objective value with respect to each cost function provide such bounds. However, tighter bounds can be very useful for obtaining tighter constraints. One possibility is to use lower and upper bounds on the value of each objective for the different positions in the ordering. In particular, if L_{ij} and U_{ij} denote valid lower and upper bounds on the value of objective i if it occupies position j , respectively, then lower and upper bounds on the value of objective i are $l_i = \min_{j \in P} L_{ij}$ and $u_i = \max_{j \in P} U_{ij}$, respectively. For $i, j \in P$ given, L_{ij} and U_{ij} can be obtained in different ways. One alternative is to solve the linear programming (LP) relaxation of the formulation, both for the minimization and the maximization of cost function i , with the additional constraint that it occupies position j . In this case L_{ij} (U_{ij}) is the optimal value of the minimization (maximization) OWAP problem in which we fix the ordering variable at value 1, i.e. $z_{ij} = 1$.

Next we present simple tests which can help to eliminate some variables by fixing their values. Broadly speaking these tests compare the value of a lower bound associated with the decision of setting (or not setting) objective i at position j with the value of a known upper bound. If the value of the lower bound exceeds the value of the upper bound, the associated decision variable can be fixed. Any feasible solution yields a valid upper bound, which corresponds to its value with respect to the objective function. In the following we use U to denote the value of the upper bound corresponding to the best-known solution. We also denote by L_{ij}^0 the optimal value of the minimization OWAP problem in which we fix the ordering variable at value 0, i.e. $z_{ij} = 0$. Then for each $i \in P, j \in P$ we have

- If $L_{ij} > U$ then $z_{ij} = 0$ (no optimal solution will have objective i in position j).
- If $L_{ij}^0 > U$ then $z_{ij} = 1$ (no optimal solution will not have objective i in position j).

5. The OWA problem on shortest paths and minimum cost perfect matchings

This section presents the formulations of the combinatorial objects that we use in our computational experiments, namely shortest paths and minimum cost perfect matchings. In order to test our results we have chosen two of the most well-known formulations for these two problems. These formulations have to be combined with those presented in previous sections to provide valid OWAP models for the Shortest Path Problem (SPP) (see e.g. [2,21]) and the Perfect Matching Problem (PMP) (see e.g. [3,8]). All the details are given in what follows.

5.1. The shortest path problem

We consider now the OWAP when Q is the SPP (see e.g. [2]). Let $G = (V, E)$ be an undirected graph with set of vertices V , $|V| = n$ and set of edges E , $|E| = m$. In addition to the sets of variables required to model the order of the p cost functions ranked by non-increasing criterion values, we will need additional variables used to model the structure of the combinatorial object (shortest path in this case). For modeling the shortest path between two selected vertices, $u_1, u_n \in V$ we use a flow-based formulation, in which binary design variables x are related to continuous flow variables φ . In particular, for each $e = (u, v) \in E$ let

$$x_e \equiv x_{uv} = \begin{cases} 1 & \text{edge } e \equiv (u, v) \text{ is in the shortest path,} \\ 0 & \text{otherwise.} \end{cases}$$

As usual, paths between $u_1, u_n \in V$ can be obtained by identifying the arcs that are used when one unit of flow is sent from u_1 to u_n . For the flow variables we consider a directed network, with set of vertices V and set of arcs A which contains two arcs, one in each direction, associated with each edge of E . For each $(u, v) \in A$ we define the decision variables φ_{uv} which represents the amount of flow through arc (u, v) . Then a characterization of the domain of feasible solutions (Q) for the SPP is:

$$\sum_{(u,v) \in A} \varphi_{u,v} - \sum_{(u,v) \in A} \varphi_{v,u} = 1 \quad u = u_1 \tag{46a}$$

$$\sum_{(u,v) \in A} \varphi_{u,v} - \sum_{(u,v) \in A} \varphi_{v,u} = -1 \quad u = u_n \tag{46b}$$

$$\sum_{(u,v) \in A} \varphi_{u,v} - \sum_{(u,v) \in A} \varphi_{v,u} = 0 \quad u \in V \setminus \{u_1, u_n\} \tag{46c}$$

$$\varphi_{u,v} + \varphi_{v,u} \leq x_{uv} \quad (u, v) \in E \tag{46d}$$

$$\varphi_{uv} \geq 0 \quad (u, v) \in A \tag{46e}$$

$$x_e \in \{0, 1\} \quad e \in E. \tag{46f}$$

Constraints (46a)–(46c) guarantee flow conservation at any vertex of the network. Constraints (46d) relate the φ and x variables, by imposing that all the edges used for sending flow in some direction are activated.

5.2. The perfect matching problem

We consider now the OWAP when Q is the PMP (see e.g. [3]). It is well known that the PMP is polynomially solvable by using the Blossom algorithm [3]. However, to the best of our knowledge it is not known how such an algorithm could be used for solving an OWAP in which Q is given by the set of perfect matchings on a given graph. Indeed, this can be done by using any of the OWAP formulations we have introduced in the previous sections.

Let $G = (V, E)$ be an undirected graph with set of vertices V , $|V| = n$ and set of edges E , $|E| = m$. In addition to the sets of variables required to model the order of the p cost functions ranked by non-increasing criterion values, we will need additional variables used to model the structure of the combinatorial object (perfect matching in this case). For modeling the perfect matching we use binary design variables x associated with the edges of the graph. In particular, for each $e = (u, v) \in E$ let

$$x_e \equiv x_{uv} = \begin{cases} 1 & \text{edge } e \equiv (u, v) \text{ is in the matching,} \\ 0 & \text{otherwise.} \end{cases}$$

We introduce some additional notation. For $S \subset V$, $E(S) = \{e = (u, v) \in E \mid u, v \in S\}$ and $\delta(S) = \{e = (u, v) \in E \mid u \in S, v \notin S\}$. When S is a singleton, i.e. $S = \{u\}$ with $u \in V$ we simply write $\delta(\{u\}) = \delta(u)$. Then, a characterization of the domain of feasible solutions for the PMP (Q) is:

$$\sum_{e \in \delta(u)} x_e = 1 \quad u \in V \quad (47a)$$

$$x_e \in \{0, 1\} \quad e \in E. \quad (47b)$$

Constraints (47a) guarantee that in the solution the degree of every vertex is one.

6. Computational experience

In this section we report on the results of some computational experiments we have run, in order to compare empirically the proposed formulations and reinforcements. We have studied the OWAP over the two combinatorial objects proposed: shortest paths and minimum cost perfect matchings. The best formulation obtained for each combinatorial object, has been later used for studying the proposed valid inequalities, including them one by one separately. Then, for each combinatorial object, we have obtained results for 14 basic formulations (i.e., without adding any valid inequality) plus 19 “reinforced” formulations. For the sake of readability, we display results in tables just for the three best basic formulations and graphics for both basic and reinforced formulations. For further details, the reader may refer to Fernández et al. [5] in order to check all the results obtained in the computational experiments organized by tables.

The OWA operator allows to model various aggregation functions according to the vector of weights w (see e.g. [16]). Some examples are the minimum, maximum, median, center or k -centrum functions. Therefore, the variation of w into non/monotonic or non/symmetric weights is directly connected with a problem structure and thus with a problem complexity [11]. Some elegant linearization of OWA functions have been proposed in the literature for some subclasses of OWA operators (see e.g. [17,18], for convex OWA with decreasing weights). For keeping the extension of the paper within some reasonable limits, in our computational experience we study a particular case of the OWA operator, namely the Hurwicz criterion [10], defined as $\alpha \max_{i \in P} y_i + (1 - \alpha) \min_{i \in P} y_i$. This criterion is non-monotonic and non-convex and, in our experience, its behavior in terms of computational effort to get optimal solutions is similar to that of other non-convex OWA criteria. In addition, this objective has been already considered when analyzing the behavior of OWA operators in multiobjective optimization (see e.g. [7]) and it is of special interest for being non-convex since the sorting weights, α , are not in a non-increasing order [9,20]. The considered values of α are $\{0.4, 0.6, 0.8\}$ and the number of objectives ranges in $|P| \in \{4, 7, 10\}$. Graphs generation is described below considering three different sizes of the graph according to $|V| \in \{100, 225, 400\}$. In addition, for each selection of the parameters $(|V|, p, \alpha)$, 10 instances were randomly generated so, in total, we have a set of 270 benchmark instances. All instances were solved with the MIP Xpress optimizer, under a Windows 7 environment in an Intel(R) Core(TM)i7 CPU 2.93 GHz processor and 8 GB RAM. Default values were used for all solver parameters. A CPU time limit of 600 s was set.

For the benchmark instances, we generated square grid networks produced as with the SPGRID generator of Cherkassky et al. [2] for both combinatorial objects. Nodes of these graphs correspond to points on the plane with integer coordinates $[x, y]$, $1 \leq x \leq \sqrt{|V|}$, $1 \leq y \leq \sqrt{|V|}$. These points are connected “forward” by arcs of the form $([x, y], [x + 1, y])$, $1 \leq x < \sqrt{|V|}$, $1 \leq y \leq \sqrt{|V|}$; “up” by arcs of the form $([x, y], [x, y + 1])$, $1 \leq x \leq \sqrt{|V|}$, $1 \leq y < \sqrt{|V|}$ and “down” by arcs of the form $([x, y], [x, y - 1])$, $1 \leq x \leq \sqrt{|V|}$, $1 < y \leq \sqrt{|V|}$ and by arcs of the form $([x, y], [x + 1, y - 1])$, $1 \leq x \leq \sqrt{|V|}$, $1 < y \leq \sqrt{|V|}$. The components of the cost vectors are randomly drawn from a uniform distribution on $[1, 100]$. Note also that shortest paths are computed between nodes 1 and $|V|$ whereas node $|V|$ is removed for the PMP when $|V|$ is odd.

Each of our tables reports the following items. Each row corresponds to a group of 10 instances with the same characteristics $(|P|, |V|, \alpha)$ indicated in the first three columns. Column $t(\#)$ reports firstly the average running time in seconds of the 10 instances of the row. In addition, if at least one instance reaches the CPU time limit, we indicate in brackets the number of instances that could be solved to optimality within the maximum CPU time limit and, in such a case, we compute

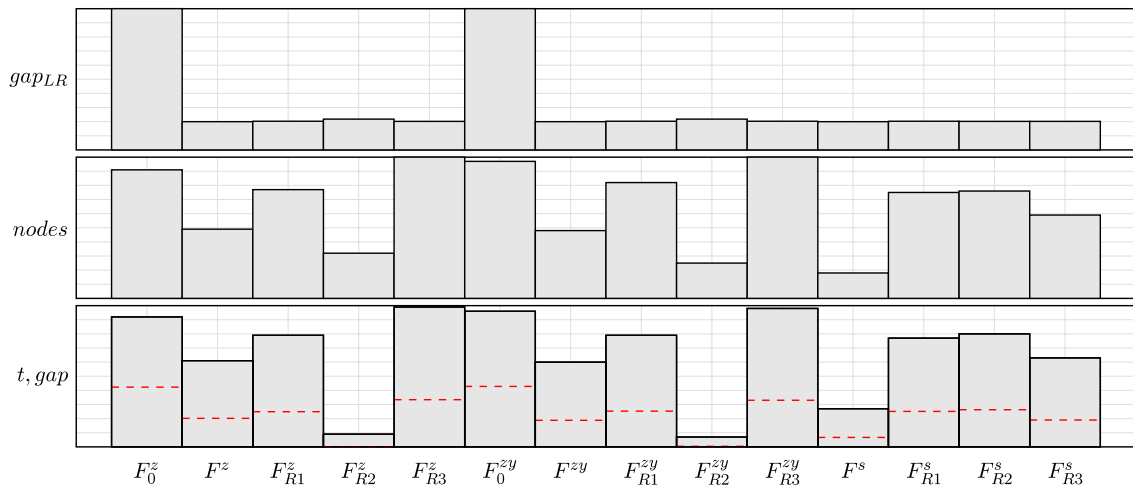


Fig. 1. Comparative results for the proposed OWAP basic formulations applied to the shortest path problem ($p = 10$, $|V| = 400$).

the average running time by using the CPU time limit for those instances that could not be solved to optimality. Column t^*/gap^* reports the biggest CPU time over the 10 instances of the group. Whenever the time limit is reached, the relative gap (indicated with a percentage %) is reported instead. Column *nodes* indicates the average number of nodes explored in the branch and bound tree and column *gap_{LR}* reports the relative gap computed with the best solution found by the solver and the linear relaxation optima at the root node. All tables report analogous items for the different formulations described along the paper. The best three formulations for each combinatorial object are F_{R2}^z , F_{R2}^{zy} , and F^s for the SPP; and F_{R1}^z , F_{R1}^{zy} , and F_{R1}^s for the PMP. Entries in bold remark best values among the 16 basic formulations (all tables are available at [5]).

Figs. 1 and 2 summarize the comparative results of all proposed basic formulations applied to each combinatorial object respectively. In these graphics the *x*-axis displays the different variations of the formulations presented in Section 3 and the *y*-axis the features analyzed. All displayed bars represent percentages of mean values computed over 90 instances with $|V| = 400$. These are the 90 hardest instances for the solver among the 270 we generated.

In particular the row labeled with “*t, gap*” shows a bar with the mean values of the running times measured in percentage over 600 s. For those instances reaching the time limit, we compute the mean running time taking the value of the time limit. Moreover, a dashed line indicates the percentage of worst case gap among those instances that have reached the time limit. The columns in the row labeled with “*nodes*” show the percentage of nodes over 10^6 that have been visited in the branch and bound tree. The columns in the row labeled with “*gap_{LR}*” report the percentage gap relative to the best solution found by the solver and the linear relaxation optima at the root node.

From the results displayed in Table 9 and Fig. 1, we observe first that the *gap_{LR}* is similar for all formulations except for F_0^z and F_0^{zy} , where a 100% of gap is reached. Formulations F_{R2}^z and F_{R2}^{zy} increase slightly the *gap_{LR}* in comparison with the remaining formulations but this does not affect negatively in the exploration as we see next. The values of *nodes* and *t, gap* are strongly related for each one of the formulations. F_0^z , F_{R3}^z , F_0^{zy} and F_{R3}^{zy} give the worst values. In contrast, F_{R2}^z , F_{R2}^{zy} and F^s produce the best values. In addition, we observe a regular behavior among all formulations with *s* variables, namely F^s , F_{R1}^s , F_{R2}^s and F_{R3}^s . Regarding the PMP, analogous conclusions can be obtained in Table 10 and Fig. 2 for the *gap_{LR}* and the relations between *nodes* and *t, gap*. However, in this case, formulations F_{R1}^z and F_{R1}^{zy} produce the best values together with F_{R1}^s , F_{R2}^s and F_{R3}^s .

Figs. 3 and 4 report analogous items as Figs. 1 and 2, but now when the valid inequalities of Section 4 are incorporated to the best basic formulations obtained for each combinatorial object. The *x*-axis displays the different variations in the formulations, starting first with the best basic formulation. Next labels refer to the valid inequality that has been added. Labels of the valid inequalities correspond with those of Section 4, where “0.1” and “0.2” refer to the two inequalities displayed in a single equation (for example the two valid inequalities of Eq. (31) are labeled as (31.1) and (31.2)). In the following we will refer indistinctly to a valid inequality and the formulation that includes such valid inequality. All displayed bars represent percentages of mean values computed over 30 random instances with $p = 10$, $|V| = 400$ and $\alpha \in \{0.4, 0.6, 0.8\}$.

From the results displayed in Fig. 3, we observe first that the *gap_{LR}* is similar for all formulations but (32.1), (36.1), (37), (39.1) and (40). As compared with F_{R2}^{zy} , formulation (36.1) improves the values of *gap_{LR}*, *nodes* and *t, gap*. However, (32.1), (39.1) and (40) improve *gap_{LR}* but are not able to improve *nodes* or *t, gap*. We also note that (37) increases *gap_{LR}* since this gap is computed with a (low quality) best solution found by the solver and the linear relaxation optima at the root node. In addition, formulations (31.1), (31.2) and (39.4) provide promising results in comparison with the values of *nodes* and *t, gap*.

From the results displayed in Fig. 4, we observe first that the *gap_{LR}* is similar for all formulations but (32.1), (36.1), (37) and (40). As compared with F_{R1}^z , formulations (32.1) and (36.1), improve *gap_{LR}* and *nodes* or *t, gap*. However, (40) improves *gap_{LR}* but is not able to improve *nodes* or *t, gap* in comparison with the best basic formulation for PMP, namely F_{R1}^z . We also note that (37) increases *gap_{LR}* since this gap is computed with a (low quality) best solution found by the solver and the

Table 9

Results obtained for the three best OWAP basic formulations applied to the shortest path problem.

Inst			F_{R2}^z				F_{R2}^{zy}				F^s			
$ V $	p	α	t (#)	t^*/gap^*	#nodes	gap_{LR}	t (#)	t^*/gap^*	#nodes	gap_{LR}	t (#)	t^*/gap^*	#nodes	gap_{LR}
100	4	0.4	0.5	0.6	15	55.79	0.4	0.6	13	55.79	8.4	59.9	16959	53.72
100	4	0.6	0.5	0.6	41	40.17	12	116.5	56370	40.17	31.9	213.4	113492	37.39
100	4	0.8	0.4	0.5	61	24.26	0.4	0.5	48	24.26	2.4	7.9	2871	20.79
100	7	0.4	0.6	0.7	200	52.77	0.6	0.8	177	52.77	121 (8)	40.66%	210086	51.11
100	7	0.6	0.7	0.8	360	38.19	0.8	1.6	468	38.19	121 (8)	22.74%	193167	36.01
100	7	0.8	0.9	1.6	839	23.76	0.9	1.4	760	23.76	20.8	125.9	36870	21.08
100	10	0.4	2.1	4.2	6658	51.98	2.5	10.3	9239	51.98	195.2 (7)	43.64%	273035	49.29
100	10	0.6	4.1	13.2	16386	37.89	2.8	11.1	9985	37.89	178.6 (8)	24.44%	238513	34.4
100	10	0.8	5.5	27.9	23353	24.83	13.1	49.4	57599	24.83	95.3	500.7	127230	20.61
225	4	0.4	0.8	1	48	55.77	0.8	1.1	45	55.77	64.4 (9)	52.43%	29874	55
225	4	0.6	0.8	1	44	39.42	0.8	1	49	39.42	91.5 (9)	31.77%	41747	38.31
225	4	0.8	0.8	1.1	95	22.13	0.8	1.2	84	22.13	243.8 (6)	14.08%	70842	20.7
225	7	0.4	1.2	1.3	99	52.61	1.3	1.8	151	52.61	129 (8)	49.52%	41763	51.29
225	7	0.6	3.3	8.8	1554	37.63	16.2	143.6	10871	37.63	185.6 (7)	31.25%	63146	35.83
225	7	0.8	4.6	22.1	3082	22.76	2.6	6.1	1204	22.76	305 (5)	14.19%	105127	20.44
225	10	0.4	9.1	62.7	6427	51.68	5.4	24.9	4222	51.68	317.1 (5)	49.98%	95076	50.33
225	10	0.6	15.2	56.6	10148	37.07	10.8	39.7	7537	37.07	319.5 (5)	32.14%	96370	35.15
225	10	0.8	38.1	147.8	41223	23.12	29.6	141.5	32090	23.12	279.6 (6)	15.16%	85419	20.81
400	4	0.4	1.4	1.8	57	55.07	1.3	1.6	55	55.07	3.3	16.8	286	54.44
400	4	0.6	1.6	2	95	38.71	1.5	1.8	76	38.71	88.5 (9)	35.79%	13806	37.84
400	4	0.8	1.8	2.9	182	21.57	1.8	3.1	265	21.57	255.9 (6)	17.21%	49806	20.47
400	7	0.4	6.5	41.1	1102	52.72	19.3	169	4370	52.72	76.4 (9)	50.67%	9192	51.85
400	7	0.6	9.4	62.6	2952	37.41	63.4 (9)	33.32%	10416	37.48	70.9 (9)	34.59%	8711	36.27
400	7	0.8	8.1	30.2	1994	21.87	7.2	24.6	1999	21.87	368.8 (4)	18.29%	32614	20.41
400	10	0.4	158.5 (9)	1.09%	100979	51.8	116.1	242.7	83991	51.8	306.4 (5)	48.93%	24184	50.73
400	10	0.6	61.8	121.5	37448	36.48	33.2	115.9	17308	36.48	132.4 (8)	31.48%	10395	35.01
400	10	0.8	229.9 (8)	0.61%	143034	21.82	155.6 (9)	0.04%	104042	21.82	142.6 (8)	17.18%	12829	19.99

Table 10

Results obtained for the three best OWAP basic formulations applied to the perfect matching problem.

Inst			F_{R1}^z				F_{R1}^{zy}				F_{R1}^s			
$ V $	p	α	t (#)	t^*/gap^*	#nodes	gap_{LR}	t (#)	t^*/gap^*	#nodes	gap_{LR}	t (#)	t^*/gap^*	#nodes	gap_{LR}
100	4	0.4	0.6	0.7	186	55.44	0.6	0.7	139	55.44	0.5	0.7	147	55.44
100	4	0.6	0.6	0.7	152	38.98	0.6	0.7	140	38.98	0.6	0.8	175	38.98
100	4	0.8	0.6	0.7	302	21.53	0.7	0.8	329	21.53	0.6	0.7	157	21.53
100	7	0.4	1	1.3	236	52.18	1	1.2	256	52.18	1	1.2	205	52.18
100	7	0.6	1.1	1.4	480	35.97	1.2	1.8	591	35.97	1.2	1.6	529	35.97
100	7	0.8	1.4	2	965	20.27	1.5	2.3	1008	20.27	1.3	1.8	1075	20.27
100	10	0.4	1.5	1.9	299	50.66	1.7	4.1	580	50.66	1.5	1.9	333	50.66
100	10	0.6	1.9	2.6	963	34.85	1.9	2.9	985	34.85	2	2.8	922	34.85
100	10	0.8	6	19.4	6329	20.2	5.6	17.6	5364	20.2	6.1	19.8	7018	20.2
225	4	0.4	2.1	4.4	1188	55.09	2	2.9	990	55.09	1.9	4.1	1095	55.09
225	4	0.6	1.7	2.9	1236	38.57	1.7	2.5	1239	38.57	1.7	2.2	982	38.57
225	4	0.8	1.9	3.2	1101	21.09	1.9	3.7	1240	21.09	2	3.6	1221	21.09
225	7	0.4	7.1	22.8	9208	52.34	8.4	36	5617	52.34	8.7	29.3	6308	52.34
225	7	0.6	10	16	6038	36.27	9.7	18.3	6206	36.27	8.8	15.9	5432	36.27
225	7	0.8	17.2	62.5	10491	20.32	17.1	48.7	10746	20.32	14.7	50.1	9525	20.32
225	10	0.4	7.5	13.2	2136	50.25	7.4	12.4	2464	50.25	7.8	15.5	2265	50.25
225	10	0.6	32.4	123.2	15537	34.56	33.9	90.1	13763	34.56	31.5	70.1	15465	34.56
225	10	0.8	295 (8)	0.32%	114029	19.62	338.7 (7)	12.07%	130079	19.7	344.7 (8)	0.33%	133025	19.62
400	4	0.4	7.3	22.3	3345	55.37	6.3	15.5	2546	55.37	6.1	9.6	2777	55.37
400	4	0.6	6.7	11.9	4103	39.04	7.5	16.7	4044	39.04	8.7	25.3	6589	39.04
400	4	0.8	9	22.1	5397	21.03	11.4	44.9	6263	21.03	9.2	19.4	5194	21.03
400	7	0.4	34.4	144.4	10464	52.05	48.9	257	15696	52.05	37.7	218.2	11164	52.05
400	7	0.6	83.4	250.9	27604	36.12	74.5	209.5	26944	36.12	78.7	185.1	28692	36.12
400	7	0.8	84.4	187.6	28762	20.19	98.2	182.5	35369	20.19	92.6	206.4	34328	20.19
400	10	0.4	68.4	197.4	13777	50.58	86.7	387.2	17514	50.58	91.9	407.6	19024	50.58
400	10	0.6	289.4 (9)	0.11%	61886	34.54	335 (9)	0.24%	69457	34.54	285.7	563.5	59428	34.54
400	10	0.8	583.5 (1)	0.42%	97022	19.5	599 (1)	0.4%	93171	19.5	577 (1)	0.43%	97258	19.52

linear relaxation optima at the root node. In addition, formulations (35.2), (36.2) and (39.3) provide promising results in comparison with the values of *nodes* or *t*, *gap*.

In summary, we observe the performance of the OWAP formulation depends on its combination with the considered combinatorial object. In particular we conclude, from our computational experience, that for the SPP, it is convenient to apply F_{R2}^{zy} reinforced with (31.1) and (31.2); although rather similar results can be obtained with F_{R2}^z . The conclusion for the

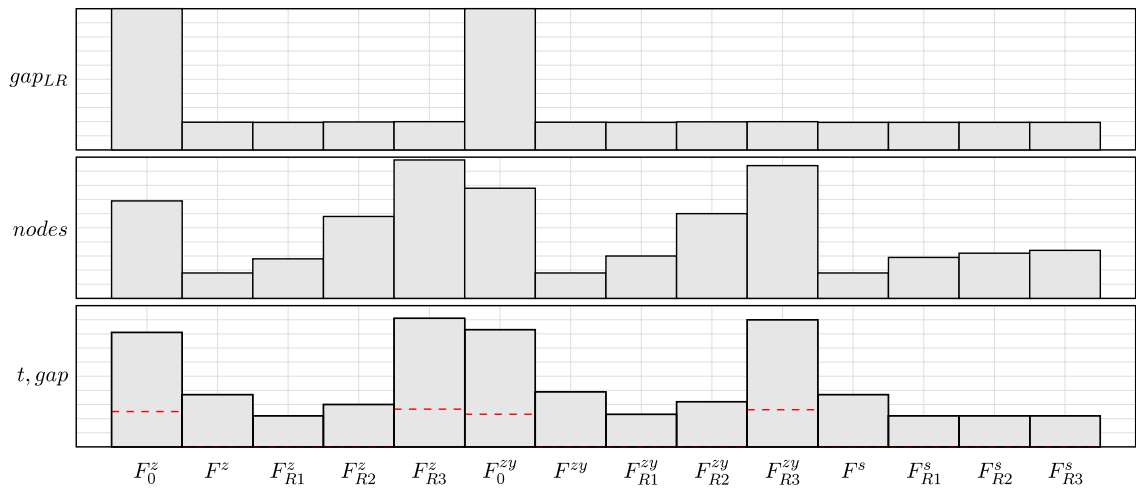


Fig. 2. Comparative results for the proposed OWAP basic formulations applied to the perfect matching problem ($p = 10, |V| = 400$).

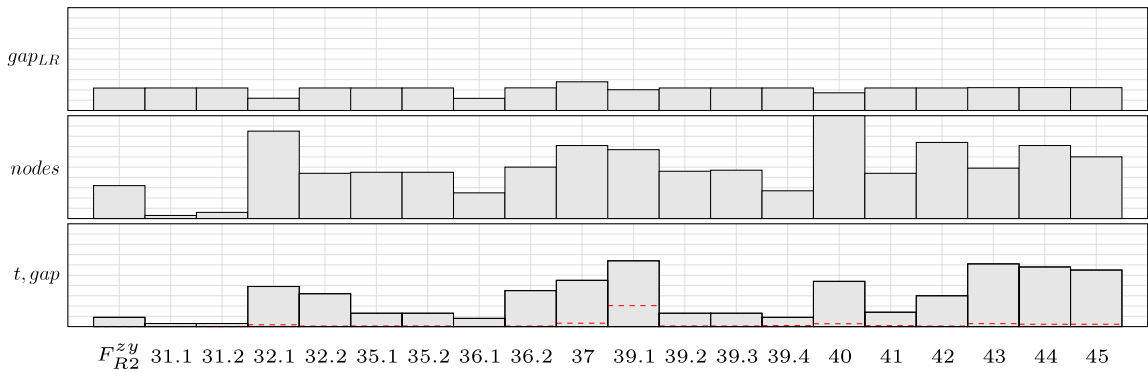


Fig. 3. Comparative results for the proposed OWAP reinforced formulations applied to the shortest path problem ($p = 10, |V| = 400$).

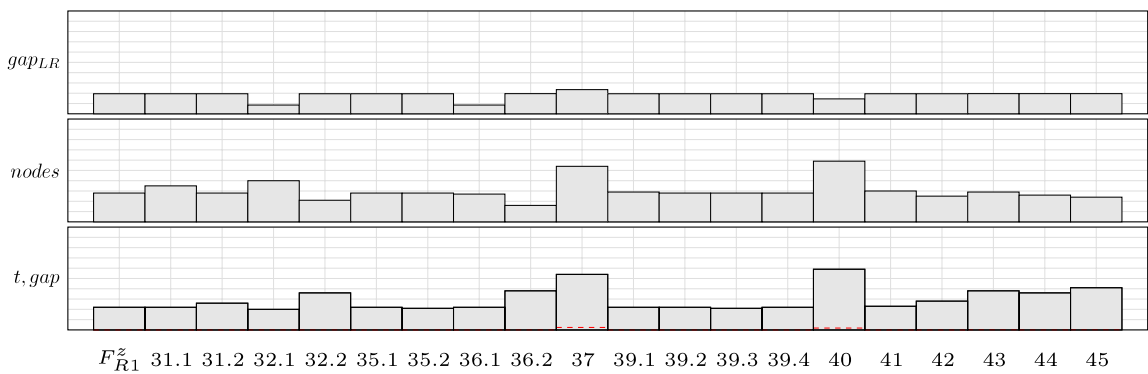


Fig. 4. Comparative results for the proposed OWAP reinforced formulations applied to the perfect matching problem ($p = 10, |V| = 400$).

PMP is different, because the best basic formulation is now F_{R1}^z and the reinforcements (32.1). Once more, rather similar results are obtained for F_{R1}^{zy} and F_{R1}^s . Therefore, we cannot conclude whether there is a formulation superior to all the others regardless of the domain Q to be considered. For this reason it is important to have developed the catalog of formulations and valid inequalities presented in this paper. In general, it is advisable to test them depending on the combinatorial object to be considered.

7. Conclusions

In this work we have presented and revisited different mathematical programming formulations for the OWAP using different sets of decision variables. These formulations reinforced with appropriate constraints have shown to be rather promising for efficiently solving many medium size OWAP instances. However, from the obtained results it is also clear that for solving larger OWAP instances with more objective functions further improvements are needed. Our current research focuses on the design of more sophisticated elimination tests as well as from alternative formulations leading to tighter LP bounds.

Acknowledgments

This research has been partially supported by the Spanish Ministry of Science and Education through grants MTM2012-36163-C06-05 and MTM2010-19576-C02-01, by Junta de Andalucía grant FQM5849 and by ERDF funds. This support is gratefully acknowledged.

References

- [1] N. Boland, P. Domínguez-Marín, S. Nickel, J. Puerto, Exact procedures for solving the discrete ordered median problem, *Comput. Oper. Res.* 33 (11) (2006) 3270–3300.
- [2] B. Cherkassky, A. Goldberg, T. Radzik, Shortest paths algorithms: theory and experimental evaluation, *Math. Program.* 73 (1996) 129–174.
- [3] J. Edmonds, Paths, trees, and flowers, *Canad. J. Math.* 17 (1965) 449–467.
- [4] M. Ehrgott, *Multicriteria Optimization*, Springer, 2005.
- [5] E. Fernández, M. Pozo, J. Puerto, A modeling framework for ordered weighted average combinatorial optimization, 2013. arXiv:1306.1426.
- [6] E. Fernández, J. Puerto, A. Rodríguez-Chia, On discrete optimization with ordering, *Ann. Oper. Res.* (2012), <http://dx.doi.org/10.1007/s10479-011-1044-7>.
- [7] L. Galand, O. Spanjaard, Exact algorithms for owa-optimization in multiobjective spanning tree problems, *Comput. Oper. Res.* 39 (2012) 1540–1554.
- [8] M. Grötschel, O. Holland, Solving matching problems with linear programming, *Math. Program.* 33 (3) (1985) 243–259.
- [9] J. Grzybowski, S. Nickel, D. Pallaschke, R. Urbanski, Ordered median functions and symmetries, *Optimization* 60 (7) (2011) 801–811.
- [10] L. Hurwicz, Optimality criteria for decision making under ignorance, Cowles Commission Discussion Paper, Chicago, 1951, p. 370.
- [11] A. Kasperski, P. Zielinski, Combinatorial optimization problems with uncertain costs and the OWA criterion, 2013. arXiv:1305.5339.
- [12] T.L. Lei, R.L. Church, Vector assignment ordered median problem: a unified median problem, *Int. Reg. Sci. Rev.* (2012), <http://dx.doi.org/10.1177/0160017612450710>.
- [13] J. Lesca, M. Minoux, P. Perny, Compact versus noncompact {LP} formulations for minimizing convex choquet integrals, *Discrete Appl. Math.* 161 (2013) 184–199.
- [14] A. Marín, S. Nickel, J. Puerto, S. Velten, A flexible model and efficient solution strategies for discrete location problems, *Discrete Appl. Math.* 157 (5) (2009) 1128–1145.
- [15] S. Nickel, J. Puerto, *Location Theory: A Unified Approach*, Springer-Verlag, Heidelberg, Germany, 2005.
- [16] W. Ogryczak, P. Olender, On MILP models for the OWA optimization, *J. Telecommun. Inf. Technol.* 2 (2012) 5–12.
- [17] W. Ogryczak, T. Sliwinski, On solving linear programs with the ordered weighted averaging objective, *European J. Oper. Res.* 148 (1) (2003) 80–91.
- [18] W. Ogryczak, A. Tamir, Minimizing the sum of the k largest functions in linear time, *Inform. Process. Lett.* 85 (3) (2003) 117–122.
- [19] P. Perny, O. Spanjaard, An axiomatic approach to robustness in search problems with multiple scenarios, in: *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence*, 2003, pp. 469–476.
- [20] J. Puerto, A. Tamir, Locating tree-shaped facilities using the ordered median objective, *Math. Program.* 102 (2) (2005) 313–338.
- [21] R. Ramaswamy, J.B. Orlin, N. Chakravarti, Sensitivity analysis for shortest path problems and maximum capacity path problems in undirected graphs, *Math. Program.* 102 (2) (2005) 355–369.
- [22] D. Schmeidler, Integral representation without additivity, *Proc. Amer. Math. Soc.* 97 (2) (1986) 255–261.
- [23] J.R. Weaver, R.L. Church, A median location model with non-closest facility service, *Transp. Sci.* 19 (1985) 58–74.
- [24] R. Yager, On ordered weighted averaging aggregation operators in multicriteria decision making, *IEEE Trans. Syst. Man Cybern.* 18 (1988) 183–190.
- [25] R. Yager, J. Kacprzyk, *The Ordered Weighted Averaging Operators: Theory and Applications*, Kluwer, Norwell, MA, 1997.