Paper

On MILP Models for the OWA Optimization

Włodzimierz Ogryczak[a] and Paweł Olender[a,b]

[a] Institute of Control and Computation Engineering, Warsaw University of Technology, Warsaw, Poland
[b] National Institute of Telecommunications, Warsaw, Poland

Abstract—The problem of aggregating multiple outcomes to form overall objective functions is of considerable importance in many applications. The ordered weighted averaging (OWA) aggregation uses the weights assigned to the ordered values (i.e., to the largest value, the second largest and so on) rather than to the specific coordinates. It allows to evaluate solutions impartially, when distribution of outcomes is more important than assignments these outcomes to the specific criteria. This applies to systems with multiple independent users or agents, whose objectives correspond to the criteria. The ordering operator causes that the OWA optimization problem is nonlinear. Several MILP models have been developed for the OWA optimization. They are built with different numbers of binary variables and auxiliary constraints. In this paper we analyze and compare computational performances of the different MILP model formulations.

Keywords—location problem, mixed integer (linear) programming, multiple criteria, ordered weighted averaging (OWA).

1. Introduction

Yager [1] introduced the so-called ordered weighted averaging (OWA) operator providing a parameterized family of aggregations that include the maximum, the minimum and the average criteria as special cases. Since its introduction, the OWA aggregation has been applied to many fields [2], including telecommunications [3], [4] and location analysis [5] among others.

In the OWA aggregation the weights are assigned to the ordered values (i.e., to the largest value, the second largest and so on) rather than to the specific coordinates. For a given weights vector $\mathbf{w} = (w_1, w_2, \dots, w_m)$, $w_i \geq 0$ for $i = 1, 2, \dots, m$, the OWA aggregation of an m-dimensional vector $\mathbf{x} = (x_1, x_2, \dots, x_m)$ can be mathematically defined as follows. We introduce the ordering map $\Theta : R^m \to R^m$ such that $\Theta(\mathbf{x}) = (\theta_1(\mathbf{x}), \theta_2(\mathbf{x}), \dots, \theta_m(\mathbf{x}))$, where $\theta_1(\mathbf{x}) \geq \theta_2(\mathbf{x}) \geq \dots \geq \theta_m(\mathbf{x})$ and there exists a permutation τ of set $I = \{1, 2, \dots, m\}$ such that $\theta_i(\mathbf{x}) = x_{\tau(i)}$ for $i = 1, 2, \dots, m$. Further, we apply the weighted sum aggregation to ordered vectors $\Theta(\mathbf{x})$, i.e., the OWA aggregation function has the form:

$$a_{\mathbf{w}}(\mathbf{x}) = \sum_{i=1}^{m} w_i \theta_i(\mathbf{x}). \tag{1}$$

Note that formula (1) differs from that originally introduced by Yager [1], due to not necessarily normalized weights ($\sum_{i=1}^{m} w_i = 1$ in [1]).

When applying the OWA aggregation as an optimization criterion we get

$$\min \left\{ \sum_{i=1}^{m} w_i \theta_i(\mathbf{x}) : \mathbf{x} \in Q \right\}. \tag{2}$$

In this paper we analyze mathematical programming models for problem (2) with nonnegative weights ($w_i \geq 0$). The ordering operator Θ causes that the OWA optimization problem (2) is nonlinear even for the case of linear programming (LP) form of the original constraints. Yager [6] has shown that the nature of the nonlinearity introduced by the ordering operations allows us to convert the optimization (2) into a mixed integer programming problem. Ogryczak and Śliwiński [7] have shown that the OWA optimization with the monotonic weights can be formed as a standard linear program of higher dimension. Several models have been proposed for locations problems with the OWA criterion (the so-called ordered median problems) [8], [9], but their computational performance have never been compared. We have carried out such comparison and additionally compared it with linear formulation for specific cases.

The paper is organized as follows. In the next section various models with the OWA criterion are presented. Within these models different formulations are considered regarding the number of constraints. In Section 3 the experiment procedure is presented and obtained results with computational models efficiency are discussed.

2. Model Formulations

As usually the whole model can be divided into two parts: physical and preference model. The physical model is based on the discrete facility location problem. This problem can be formulated as mixed integer linear programming. The OWA operator constitutes the preference model. In general, it can also be formulated as mixed integer linear programming. However, as mentioned earlier, in specific cases it is possible to form it as linear programming. Certainly, the whole model will remain mixed integer due to underlying location problem. In the article we are focusing on the OWA optimization so we are referring to mixed integer and linear programming meaning the preference model only.

2.1. Location Problem

A standard formulation of facility location problem without the capacity limits was used. There is given a set of m sites (e.g., clients). We have to place n facilities to satisfy demands from the clients. Without loss of generality it can be assumed that the number of candidate sites is identical to the number of clients and additionally that $n \leq m$. Then each client is assigned to the facility that meets its demand. The assignment is done in such a way to optimize a given objective function. The objective function is usually based on distances (costs) between the clients and the facilities. Because we consider unlimited capacities each client is assigned the closest facility. Formally the model can be expressed in the following form:

$$\sum_{j=1}^{m} u_j = n, \tag{3}$$

$$\sum_{j=1}^{m} v_{ij} = 1 \quad \text{for} \quad i = 1,\dots,m, \tag{4}$$

$$v_{ij} \leq u_j \quad \text{for} \quad i,j = 1,\dots,m, \tag{5}$$

$$x_i = \sum_{j=1}^{m} c_{ij} v_{ij} \quad \text{for} \quad i = 1,\dots,m, \tag{6}$$

$$u_j \in \{0,1\} \quad \text{for} \quad j = 1,\dots,m, \tag{7}$$

$$v_{ij} \geq 0 \quad \text{for} \quad i,j = 1,\dots,m, \tag{8}$$

where c_{ij} denotes the cost of satisfying the total demand of client i by facility j. There are used two groups of binary variables representing, respectively, the location and the allocation decisions:

- u_j – equal 1 if a facility is built at site j and 0 otherwise,
- v_{ij} – equal 1 if the demand of client i is satisfied by facility j and 0 otherwise.

The auxiliary variable x_i (6) expresses the cost of satisfying the demand of client i. The constraint (3) enforces that exactly n facilities are placed. The fact that each client is assigned to only one facility is modeled with constraint (4). Constraint (5) ensures that the clients are assigned to the existing facilities. Thus, above formulation defines a set of attainable values Q and the corresponding cost (outcome) vectors $\mathbf{x}$. On this basis preference models with OWA criterion can be defined.

2.2. The First MILP Model (M1)

The ordering operator Θ causes that the OWA optimization problem (2) is nonlinear, however, the nonlinearity can be transformed into discrete problem. Note that the quantity $\theta_1(\mathbf{x})$ representing the worst outcome can be easily computed directly by the LP minimization:

$$\theta_1(\mathbf{x}) = \min y_1 \tag{9}$$

subject to

$$y_1 \geq x_i \quad \text{for} \quad i = 1,\dots,m. \tag{10}$$

Following Yager [6], similar formula can be given for any $\theta_k(\mathbf{x})$, although requiring the use of integer variables. Namely, for any $k = 1,2,\dots,m$ the following formula is valid [7]:

$$\theta_k(\mathbf{x}) = \min y_k \tag{11}$$

$$\text{s.t.} \quad y_k + M z_{ki} \geq x_i \quad \text{for} \quad i = 1,\dots,m, \tag{12}$$

$$\sum_{i=1}^{m} z_{ki} \leq k-1, \tag{13}$$

$$z_{ki} \in \{0,1\} \quad \text{for} \quad i = 1,\dots,m, \tag{14}$$

where M is a sufficiently large constant (larger than any possible difference between various individual outcomes y_i). Note that for $k = 1$ all the binary variables z_{1i} are enforced to 0 thus reducing the optimization to the standard LP model for that case.

The entire OWA optimization model (2) can be formulated as the following mixed integer linear programming problem (MILP) [7]:

$$\min \sum_{k=1}^{m} w_k y_k, \tag{15}$$

$$y_k + M z_{ki} \geq x_i \quad \text{for} \quad i,k = 1,\dots,m, \tag{16}$$

$$\sum_{i=1}^{m} z_{ki} \leq k-1 \quad \text{for} \quad k = 1,\dots,m, \tag{17}$$

$$z_{ki} \in \{0,1\} \quad \text{for} \quad i,k = 1,\dots,m, \tag{18}$$

$$\mathbf{x} \in Q. \tag{19}$$

This MILP model introduces $O(m^2)$ binary variables z_{ki} organized in m multiple choice constraints (special ordered sets) and m continuous variables y_k defined by the corresponding m inequalities. Actually, the original model introduced in [6] contains additional constraints

$$y_k \geq y_{k+1} \quad \text{for} \quad k = 1,\dots,m-1, \tag{20}$$

representing ordering inequalities on variables y_k. Due to minimization with nonnegative weights w_k, these inequalities are redundant in the sense that they do not affect the optimal solution. However, we will examine if they influence the computational performance of the model. Additionally, we will also consider another redundant constraint,

$$\sum_{k=1}^{m} y_k = \sum_{i=1}^{m} x_i, \tag{21}$$

which balances the total sum of coordinates of basic cost vector $\mathbf{x}$ against sorted vector $\mathbf{y}$.

Eventually, we take into consideration three different formulations of model M1:

- M1_1 – formulation (15)–(19) without the redundant constraints,
- M1_2 – formulation (15)–(19) with one redundant constraint (20),
- M1_3 – formulation (15)–(19) with two redundant constraints (20) and (21).

2.3. The Second MILP Model (M2)

Several MILP models have been developed for the OWA optimization within the ordered median location problems [5]. Starting from quadratic MIP, through MILP models with $O(m^3)$ binary variables and finally setting the MILP model with $O(m^2)$ binary variables and constraints [8]. Adapting the model to our notation, it can be equivalently written as follows:

$$\min \sum_{k=1}^{m} w_k y_k \,, \tag{22}$$

$$y_k \geq y_{k+1} \quad \text{for} \quad k = 1,\ldots,m-1, \tag{23}$$

$$y_k + M(1 - s_{ki}) \geq x_i \quad \text{for} \quad i,k = 1,\ldots,m, \tag{24}$$

$$\sum_{i=1}^{m} s_{ki} = 1 \quad \text{for} \quad k = 1,\ldots,m, \tag{25}$$

$$\sum_{k=1}^{m} s_{ki} = 1 \quad \text{for} \quad i = 1,\ldots,m, \tag{26}$$

$$s_{ki} \in \{0,1\} \quad \text{for} \quad i,k = 1,\ldots,m, \tag{27}$$

$$\sum_{k=1}^{m} y_k = \sum_{i=1}^{m} x_i, \tag{28}$$

$$\mathbf{x} \in Q, \tag{29}$$

where m^2 binary variables s_{ki} represent assignment of actual values x_i to the ordered ones y_k. That means, $s_{ki} = 1$ if the value of x_i is the k-th largest and zero otherwise. This model is based on a combination of assignment and sorting problems. The sorting part is realized by constraints (23), (25) and (26). For such a modeling approach the $m-1$ inequalities ordering variables y_k are necessary. On the other hand, due to minimization with the nonnegative weights w_k, the equation balancing vector $\mathbf{x}$ and $\mathbf{y}$ is redundant. Thus the model can be then considered without this (single) equation (28). Although, the full model containing the balance equation is applicable both for minimization and maximization cases. For these reasons we have analyzed two formulations of this model:

- M2_1 – formulation (22)–(29) with the redundant constraint (28),
- M2_2 – formulation (22)–(27), (29) without the redundant constraint.

2.4. LP Model

The ordering operator Θ used in the OWA aggregation is nonlinear and, in general, it is hard to implement. However, with decreasing weights the OWA aggregation is a piecewise linear convex function and its minimization can be expressed in the linear programming form [7]. This so-called deviational model is based on the linear programming representation of the cumulated ordered outcomes:

$$\overline{\theta}_k(\mathbf{x}) = \sum_{i=1}^{k} \theta_i(\mathbf{x}) \quad \text{for} \quad k = 1,\ldots,m. \tag{30}$$

The quantities $\overline{\theta}_k(\mathbf{x})$ for $k = 1,\ldots,m$ express, respectively: the worst (largest) outcome, the total of the two worst outcomes, the total of the three worst outcomes, etc. As it was shown in [7] each of these values can be found as the optimal value of the following LP problem:

$$\overline{\theta}_k(\mathbf{x}) = \min \left(k t_k + \sum_{i=1}^{m} d_{ik} \right) \tag{31}$$

subject to

$$d_{ik} \geq x_i - t_k, d_{ik} \geq 0 \quad \text{for } i = 1,\ldots,m. \tag{32}$$

The ordered outcomes can be expressed as differences $\theta_k(\mathbf{x}) = \overline{\theta}_k(\mathbf{x}) - \overline{\theta}_{k-1}(\mathbf{x})$ for $k = 2,\ldots,m$ and $\theta_1(\mathbf{x}) = \overline{\theta}_1(\mathbf{x})$. Hence, the OWA problem with weights w_k can be expressed in the form:

$$\min \sum_{k=1}^{m} (w_k - w_{k+1}) \left(k t_k + \sum_{i=1}^{m} d_{ik} \right) \tag{33}$$

$$d_{ik} \geq x_i - t_k \quad \text{for} \quad i,k = 1,\ldots,m, \tag{34}$$

$$d_{ik} \geq 0 \quad \text{for} \quad i,k = 1,\ldots,m, \tag{35}$$

$$\mathbf{x} \in Q. \tag{36}$$

For this model we also consider some redundant constraints. One of them represents ordering inequalities on variables t_k, thus taking the form:

$$t_k \geq t_{k+1} \quad \text{for} \quad k = 1,\ldots,m-1. \tag{37}$$

The second constraint concerns ordering of deviations d_{ik} but in reverse order:

$$d_{ik} \leq d_{ik+1} \quad \text{for} \quad i = 1,\ldots,m,\ k = 1,\ldots,m-1. \tag{38}$$

The last redundant constraint is a relaxed form of the previous one:

$$\sum_{i=1}^{m} d_{ik} \leq \sum_{i=1}^{m} d_{ik+1} \quad \text{for} \quad k = 1,\ldots,m-1. \tag{39}$$

We carry out computational analyzes of four following formulations:

- MLP1 – formulation (33)–(36) without the redundant constraints,
- MLP2 – formulation (33)–(36) with the redundant constraint (37),
- MLP3 – formulation (33)–(36) with the redundant constraint (38),
- MLP4 – formulation (33)–(36) with the redundant constraint (39).

3. Computational Tests

In order to analyze the computational efficiency of the presented models and their different formulations, we have applied them to various location problems and compared time needed to solve these tasks for specific formulations. The experiment procedure, including problem generation, is explained below. Next, results are presented and models comparison are discussed.

3.1. Experiments Design

The general scheme of experiments is analogous to that presented in [10]. To evaluate the models on different cases, basic parameters characterizing the location problem have been chosen and their sets of considered value were determined. Then, based on combinations of these parameters various instances of problem location have been defined. The parameters that have been considered are: the number of sites (locations), the number of service points to be placed and type of problem defined by the vector of weights in the OWA aggregation.

The number of sites is very important parameter because, in fact, it determines the size of the problem. We have considered smaller sizes for the mixed integer formulations, and bigger for the linear model:

- for the MILP models – SC1: $m = 8$, SC2: $m = 10$, SC3*: $m = 12$, SC4*: $m = 15$,
- additionally for the LP model – SC5: $m = 20$, SC6: $m = 25$, SC7: $m = 30$.

In cases SC3 and SC4 (marked by an asterisk) only one MILP formulation has been tested (for problems with monotonic weights) in order to compare it with the linear formulation.

The second parameter, the number of facilities, has been defined as proportional to the problem size (m value). Following cases have been assumed: $n = \lceil \frac{m}{4} \rceil$, $n = \lceil \frac{m}{3} \rceil$, $n = \lceil \frac{m}{2} \rceil$, $n = \lceil \frac{m}{2} + 1 \rceil$, where $\lceil a \rceil$ is the smallest integer value not smaller than a.

Type of problem defined by the vector of weights $\mathbf{w}$ plays an important role. It allows to represent a wide range of problems (strictly speaking the preferences), which is directly connected with a problem structure and thus with problem complexity. We have examined 12 problem types:

- TC1: N-median, i.e. $\mathbf{w} = (\underbrace{1,\ldots,1}_{m})$,
- TC2: N-center problem, i.e. $\mathbf{w} = (1,\underbrace{0,\ldots,0}_{m-1})$,
- TC3: k-centra problem, i.e. $\mathbf{w} = (\underbrace{1,\ldots,1}_{k},0,\ldots,0)$, where $k = \lfloor \frac{m}{3} \rfloor$,
- TC4: $k_1 + k_2$-trimmed mean problem, i.e., $\mathbf{w} = (\underbrace{0,\ldots,0}_{k_1},1,\ldots,1,\underbrace{0,\ldots,0}_{k_2})$, where $k_1 = \lceil \frac{m}{10} \rceil$, and $k_2 = \lceil n + \frac{m}{10} \rceil$,
- TC5: $\mathbf{w}$ with binary entries alternating 0 and 1, and beginning with 1, i.e. $\mathbf{w} = (1,0,1,0,1,0,\ldots)$,
- TC6: Such as TC5, but beginning with 0, i.e. $\mathbf{w} = (0,1,0,1,0,1,\ldots)$,
- TC7: The repetition of the sequence $(1,1,0)$, i.e. $\mathbf{w} = (1,1,0,1,1,0,\ldots)$,
- TC8: The repetition of the sequence $(1,0,0)$, i.e. $\mathbf{w} = (1,0,0,1,0,0,\ldots)$,
- TC9: Beginning with m (size of the problem) and decreasing by 1, i.e., $\mathbf{w} = (m, m-1, \ldots, 2, 1)$,
- TC10: Such as TC9, but in reverse order (increasing), i.e., $\mathbf{w} = (1, 2, \ldots, m-1, m)$,
- TC11: Beginning with $3m$ and decreasing in a piecewise linear manner, k weights by 3, next k weights by 2 and rest by 1, i.e.,
$$\mathbf{w} = (3m, \underbrace{3(m-1),\ldots,3(m-k)}_{k}, \underbrace{3(m-k)-2,\ldots,3(m-k)-2k}_{k}, 3m-5k-1, 3m-5k-2,\ldots),$$
where $k = \lfloor \frac{m}{3} \rfloor$,
- TC12: Such as TC11, but in reverse order (increasing), i.e.,
$$\mathbf{w} = (\ \ldots\ , 3m-5k-2, 3m-5k-1, \underbrace{3(m-k)-2k,\ldots,3(m-k)-2}_{k}, \underbrace{3(m-k),\ldots,3(m-1)}_{k}, 3m),$$
where $k = \lfloor \frac{m}{3} \rfloor$.

The first two of the eight problems (TC1–TC8) are basic problems in the location theory [10]. The next two are not so popular but also used in this field. Problems TC5–TC8 are in some sense artificial and have been used particularly to test the computational efficiency. The last four problems have monotonic weights. Depending on the type of monotonicity, they are simpler (TC9, TC11 with decreasing weights) or harder (TC10, TC12 with increasing weights) problems. These types of the problems can be treated as extended versions of max min (TC9, TC11) and max max (TC10, TC12) objective functions, respectively.

For each size case we have generated 15 cost matrices, which have zero on the main diagonal and the remaining entries randomly generated from a discrete uniform distribution in the interval $[1, 100]$. These matrices have been assigned to each combination of the parameters with corresponding problem size. Thus, we have received a set of test problem instances.

3.2. Results

The efficiency comparison has been carried out based on the average computational time needed to solve a problem. We have compared computational time for specific sizes and problem types averaging over instances of cost matrices and cases of facilities number to be placed. The complete results for each model are presented in Tables 1–3.

First, we have examined the influence of redundant constraint on the computational efficiency of MILP models. In order to check the change of the performance we have compared different formulations within the individual models, which were presented in Subsections 2.2 and 2.3.

Table 1
Average solution time for MILP model M1

Formulation		M1_1	M1_2	M1_3
SC1	TC1	0.291	0.222	0.025
	TC2	0.011	0.014	0.034
	TC3	0.033	0.055	0.055
	TC4	0.115	0.152	0.186
	TC5	0.081	0.107	0.051
	TC6	0.094	0.126	0.163
	TC7	0.129	0.148	0.055
	TC8	0.049	0.073	0.056
	TC9	0.216	0.204	0.068
	TC10	0.500	0.411	0.145
	TC11	0.241	0.221	0.053
	TC12	0.302	0.241	0.052
SC2	TC1	1.728	1.378	0.047
	TC2	0.020	0.022	0.045
	TC3	0.126	0.225	0.141
	TC4	1.311	1.315	1.537
	TC5	0.289	0.456	0.173
	TC6	0.483	0.665	0.923
	TC7	0.512	0.699	0.135
	TC8	0.150	0.258	0.140
	TC9	1.143	0.969	0.185
	TC10	12.721	2.820	1.379
	TC11	1.217	1.002	0.153
	TC12	2.909	1.784	0.149
SC3	TC9	–	–	0.708
	TC11	–	–	0.543
SC4	TC9	–	–	2.515
	TC11	–	–	1.760

The results for the first model are given in Table 1. For better illustration of the differences we present it graphically for SC1: $m = 8$ in Fig. 1 (for SC2 the results are similar). One may notice that adding the redundant constraints leads to significant time reduction for the problems with all non-zero weights (TC1, TC9–TC12). The situation is different in the case of problems that are focused on minimizing larger values of the outcome vector components (TC2–TC4), where the time slightly increases. For the other problems (TC5–TC8) it is hard to define a clear trend of change. In particular, comparing the results for TC5 and TC6, for which weight vectors are alternating sequences of 0 and 1 (where in TC5 sequence begins with 1, and in TC6 with 0), one may notice a significant difference in the time change due to adding the redundant constraints. The same situation is for the second model, which can be

Table 2
Average solution time for MILP model M2

Formulation		M2_1	M2_2
SC1	TC1	0.120	6.553
	TC2	1.697	0.089
	TC3	2.317	0.176
	TC4	4.848	0.697
	TC5	0.742	1.806
	TC6	5.207	2.025
	TC7	0.561	3.478
	TC8	0.978	0.808
	TC9	1.045	4.988
	TC10	1.843	6.820
	TC11	0.574	5.481
	TC12	0.279	7.038
SC2	TC1	0.478	177.263
	TC2	14.421	0.426
	TC3	89.987	2.397
	TC4	225.014	6.421
	TC5	8.289	17.388
	TC6	192.945	31.407
	TC7	5.987	37.549
	TC8	16.010	6.949
	TC9	13.095	117.014
	TC10	63.474	197.318
	TC11	6.506	133.892
	TC12	1.993	222.029

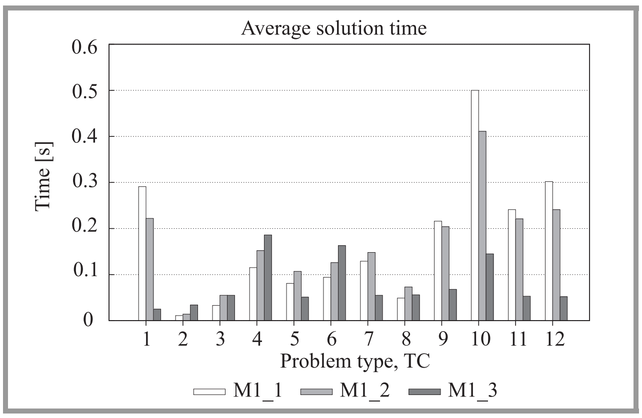


Fig. 1. Model M1 formulations comparison ($m = 8$).

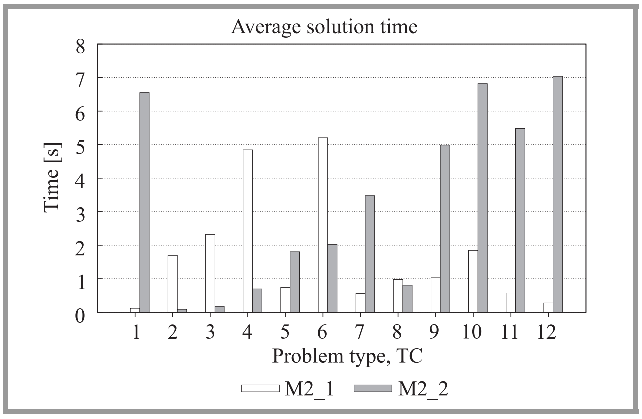


Fig. 2. Model M2 formulations comparison ($m = 8$).

seen in Table 2 and Fig. 2. The relationships discussed above are even more apparent here. In particular, there is a greater time increase after adding the redundant constraints in the case of problems TC2–TC4.

Next we have juxtaposed the results of model M1 with the results of model M2. For this purpose the corresponding formulations of these models have been confronted, namely the formulation with and without redundant constraints. In the former case the formulation M1_3 is compared with the formulation M2_1 (Fig. 3). As seen for all types of problems, the first model shows much better performance than the second one. There is a similar situation for the formulations without the redundant constraints, where the formulation M1_1 has been compared with the formulation M2_2 (Fig. 4). Here, also the solution time for the first model turns out to be much shorter than that for its counterpart for all types of problems. The scale of the differences are especially noteworthy and reaches one or even two (three for SC2) orders of magnitude. On this basis, model M1 seems to be more efficient than model M2.

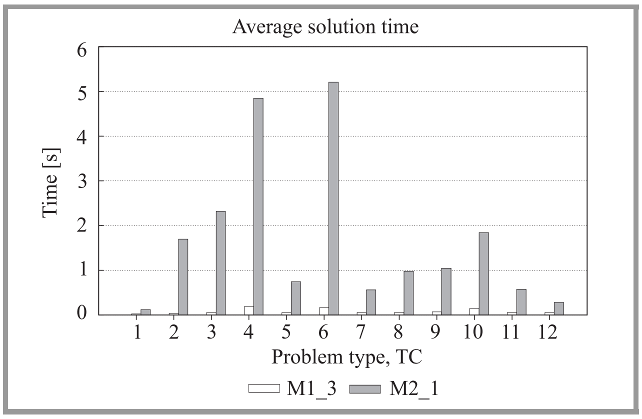


Fig. 3. The comparison of formulations with the redundant constraints ($m = 8$).

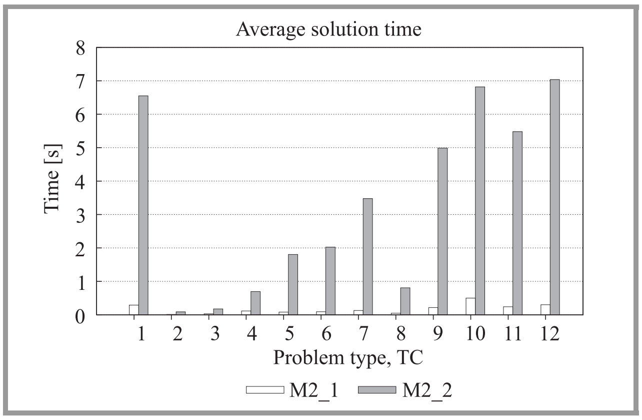


Fig. 4. The comparison of formulations without the redundant constraints ($m = 8$).

As mentioned earlier, the specific problems with appropriately monotonic (decreasing in the case of minimization) OWA weights can be formulated as the standard linear programming models. We have examined whether MILP models could also take advantage of this special structure. For this purpose we have compared their computational time for the problems with decreasing weights (easier problems – TC9, TC11) and increasing weights (harder problems – TC10, TC12).

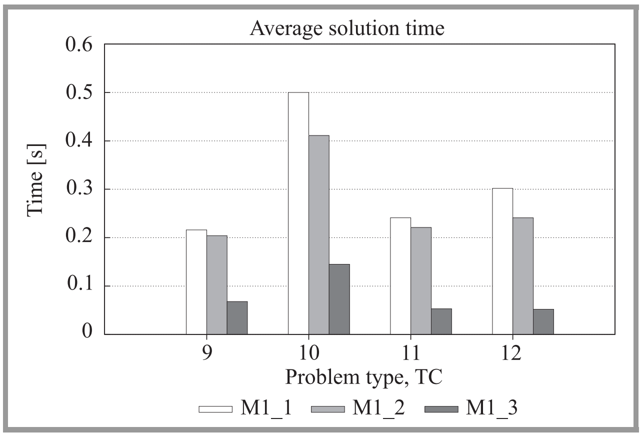


Fig. 5. Model M1 comparison with monotonic weights ($m = 8$).

The analysis shows that model M1 (Fig. 5), in the case of problems TC9 and TC10, demonstrates actually much better solution time for decreasing weights (TC9). In the case of problems TC11 and TC12, the differences are not so significant, and for the formulation M1_3 the solution times can be considered equal.

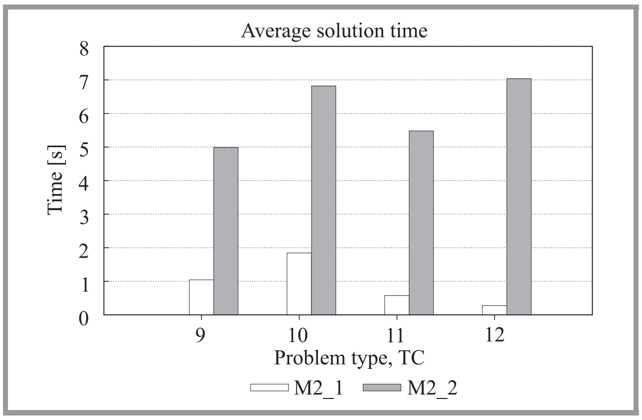


Fig. 6. Model M2 comparison with monotonic weights ($m = 8$).

Considering model M2 (Fig. 6) it can be seen that the formulation without the redundant constraints (M2_2) has also slightly shorter computational time for decreasing weights. The same is true for the formulation M2_1, when comparing TC9 with TC10 problem. However, the situation looks differently for the formulation M2_1 for TC11 and TC12 problem. Here, the problem with increasing weights has a shorter solution time. This suggests that the MILP models do not guarantee better performance for the problems with decreasing weights.

Because of the above results obtained for MILP models for monotonic weights we have carried out the direct comparison between the mixed integer and the linear (Subsection 2.4) OWA formulations. The results for the linear OWA formulations are given in Table 3. We have compared

Table 3
Average solution time for LP model

Formulation		MLP1	MLP2	MLP3	MLP4
SC3	TC1	0.008	0.007	0.008	0.008
	TC2	0.033	0.032	0.033	0.031
	TC3	0.042	0.041	0.042	0.042
	TC9	0.047	0.048	0.054	0.054
	TC11	0.036	0.036	0.042	0.041
SC4	TC1	0.012	0.012	0.013	0.012
	TC2	0.058	0.056	0.057	0.057
	TC3	0.067	0.064	0.065	0.067
	TC9	0.076	0.075	0.090	0.089
	TC11	0.059	0.057	0.071	0.071
SC5	TC1	0.018	0.019	0.019	0.018
	TC2	0.179	0.181	0.182	0.185
	TC3	0.199	0.199	0.203	0.205
	TC9	0.207	0.210	0.261	0.270
	TC11	0.139	0.138	0.182	0.183
SC6	TC1	0.024	0.024	0.029	0.024
	TC2	0.400	0.396	0.405	0.403
	TC3	0.528	0.529	0.531	0.534
	TC9	0.485	0.482	0.613	0.648
	TC11	0.305	0.285	0.390	0.395
SC7	TC1	0.032	0.030	0.035	0.032
	TC2	1.383	1.376	1.383	1.399
	TC3	1.163	1.157	1.170	1.164
	TC9	1.271	1.302	1.709	1.733
	TC11	0.750	0.727	0.979	1.017

the most efficient (in the sense of considered problem types) MILP formulation (M1_3) and basic formulation of linear model (MLP1) for the problems with decreasing weights (TC9, TC11). As shown in the graphical comparison (Figs. 7 and 8), even the best considered mixed integer programming model has much worse performance than the linear formulation of OWA. The differences reach

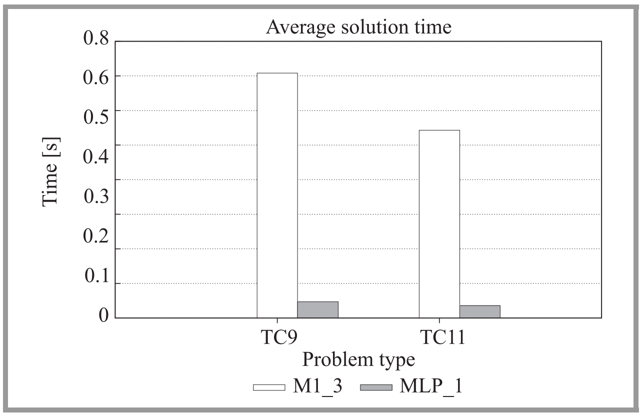


Fig. 7. MILP and LP models comparison with decreasing weights ($m = 12$).

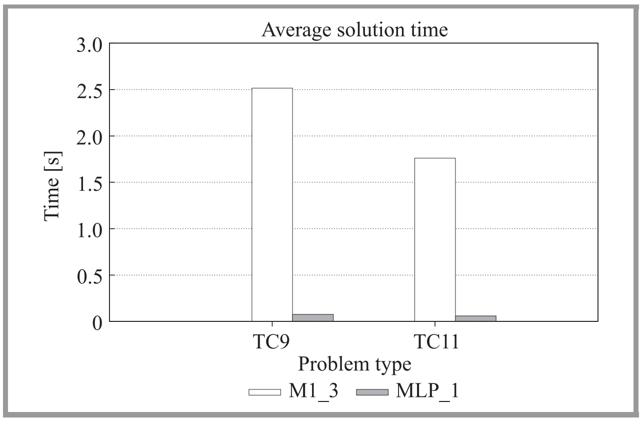


Fig. 8. MILP and LP models comparison with decreasing weights ($m = 15$).

two orders of magnitude. Therefore, even if sometimes MILP models solve the OWA optimization with appropriate monotonic weights (simpler problems) more effectively than the general case OWA, they are still much less efficient than the linear OWA formulation for these specific cases.

Knowing that the linear programming formulation of the OWA has better computational performances than the mixed integer linear programming formulation and that the redundant constraints can significantly improve efficiency of the latter one, we have tested the influence of the redundant constraints on the linear programming model. We have considered four formulations from Subsection 2.4 for the problems with decreasing weights (TC9, TC11) and additionally TC1–TC3 as non-increasing. The results are presented in Table 3. In Fig. 9 the case for $m = 30$ is

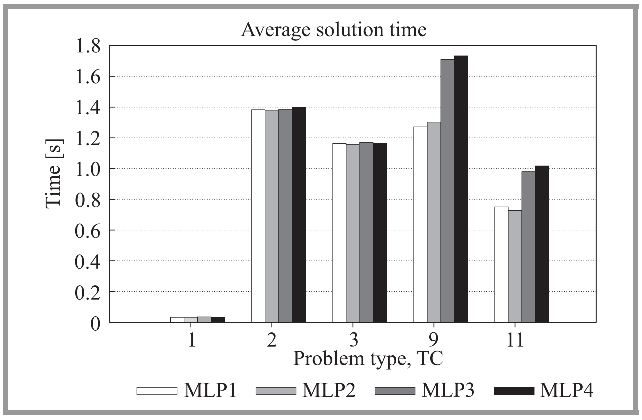


Fig. 9. Model LP formulations comparison ($m = 30$).

shown. One may notice no difference in the case of problems TC1–TC3. For formulation MLP1 and MLP2 there is also no difference in case of the other problems. However, for MLP3 and MLP4 formulations there is about 30% performance deterioration in the case of problems TC9 and TC11. Similar situation occurs for other size cases. Thus, it seems that the redundant constraints do not improve the performance of the linear programming model of the OWA optimization.

4. Conclusions

The paper analyzes two models of mixed integer programming and one linear programming model for optimization of the OWA criterion. Experiments were conducted to compare the computational efficiency of different formulations of these models. Based on the obtained results it can be concluded that the redundant constraints added to MILP models of OWA can significantly shorten the computational time for certain types of localization problems (certain classes of OWA weights vectors). Secondly, the model M1 appears to be much more efficient than the model M2. Besides, if the problem has special structure, which allows one to formulate OWA criterion as standard linear formulation, this should be exploited, as it greatly increases its computational efficiency. However, adding the redundant constraints to the linear programming OWA formulation does not help and may increase the computational time.

Because the results presented here are based on an average solution time, it seems desirable to conduct a more detailed statistical analysis (e.g., minimum, maximum, variance) of the results. Perhaps it will allow to find new dependencies and determine more detailed model characteristics. Better efficiency of the model M1 suggests also an opportunity to apply it to quadratic assignment problem (QAP), from which some transformations for the model M2 have been exploited [8].

Acknowledgment

The research was partially supported by the National Science Centre (Poland) under grant 2011/01/B/ST7/02967 “Integer programming models for joint optimization of link capacity assignment, transmission scheduling, and routing in fair multicommodity flow networks.”

References

[1] R. R. Yager, “On ordered weighted averaging aggregation operators in multicriteria decision making”, *IEEE Trans. Sys., Man and Cyber.*, vol. 18, pp. 183–190, 1988.

[2] R. R. Yager, J. Kacprzyk, and G. Beliakov (Eds.), *Recent Developments in the Ordered Weighted Averaging Operators: Theory and Practice*. Springer, 2011.

[3] W. Ogryczak, T. Śliwiński, and A. Wierzbicki, “Fair resource allocation schemes and network dimensioning problems”, *J. Telecom. Inform. Technol.*, no. 3, pp. 34–42, 2003.

[4] M. Köppen, K. Yoshida, and M. Tsuru, Y. Oie, “Annealing heuristic for fair wireless channel allocation by exponential ordered-ordered weighted averaging operator maximization”, in *Proc. IEEE/IPSJ Int. Symp. Appl. Internet*, Munich, Germany, 2011, pp. 538–543.

[5] S. Nickel and J. Puerto, *Location Theory: A Unified Approach*. Berlin, Springer, 2005.

[6] R. R. Yager, “Constrained OWA aggregation”, *Fuzzy Sets and Systems*, vol. 81, pp. 89–101, 1996.

[7] W. Ogryczak and T. Śliwiński, “On solving linear programs with the ordered weighted averaging objective”, *Eur. J. Oper. Res.*, vol. 148, pp. 80–91, 2003.

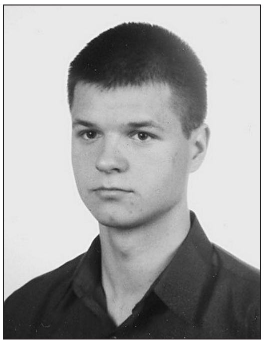
[8] N. Boland, P. Dominguez-Marin, S. Nickel, and J. Puerto, “Exact procedures for solving the discrete ordered median problem”, *Comput. Oper. Res.*, vol. 33, pp. 3270–3300, 2006.

[9] W. Ogryczak and A. Tamir, “Minimizing the sum of the k largest functions in linear time”, *Inform. Process. Letters*, vol. 85, pp. 117–122, 2003.

[10] P. Dominguez-Marin, *The Discrete Ordered Median Problem: Models and Solution Methods*. Springer, 2003.

Włodzimierz Ogryczak is a Professor and Deputy Director for Research in the Institute of Control and Computation Engineering (ICCE) at the Warsaw University of Technology, Poland. He received both his M.Sc. (1973) and Ph.D. (1983) in Mathematics from Warsaw University, and D.Sc. (1997) in Computer Science from Polish Academy of Sciences. His research interests are focused on models, computer solutions and interdisciplinary applications in the area of optimization and decision making with the main stress on: multiple criteria optimization and decision support, decision making under risk, location and distribution problems. He has published three books and numerous research articles in international journals.

E-mail: wogrycza@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland



Paweł Olender received the M.Sc. in Computer Science from the Warsaw University of Technology, Poland, in 2008. Currently, he is a Ph.D. student in computer science at the Institute of Control and Computation Engineering at the Warsaw University of Technology. He is employed by the National Institute of Telecommunications in Warsaw. He has participated in projects related to data warehousing and analysis for a telecommunication operator. His research interests are focused on modeling, decision support, optimization, data mining and multi-agent systems.

E-mail: P.Olender@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

E-mail: P.Olender@itl.waw.pl
National Institute of Telecommunications
Szachowa st 1
04-894 Warsaw, Poland