Na prawach rękopisu
bez ograniczeń

INSTYTUT MATEMATYKI
POLITECHNIKA WROCŁAWSKA

Raport serii: PREPRINTY nr 007

Minimizing maximal regret in the linear assignment problems with interval costs

Adam Kasperski
Paweł Zieliński

Słowa kluczowe:
Optymalizacja dyskretna,
Zagadnienie przydziału,
Liczba przedziałowa,
Złożoność obliczeniowa

Wrocław 2004

Minimizing maximal regret in the linear assignment problems with interval costs

Adam Kasperski*
Institute of Industrial Engineering and Management, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
adam.kasperski@pwr.wroc.pl

Paweł Zieliński
Institute of Mathematics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
pziel@im.pwr.wroc.pl

Abstract

In this paper the linear bottleneck assignment problem and the linear sum assignment problem with interval costs are examined. In order to calculate the optimal solution to these problems one of the robust criteria, called the minimax regret, is adopted. It is shown that the robust linear bottleneck assignment problem can be solved in polynomial time while the robust linear sum assignment problem is $\mathcal{NP}$-hard. For the robust linear sum assignment problem a mixed integer model is proposed and some computational experiments are presented.

Keywords: Combinatorial optimization; Robust optimization; Assignment problem; Interval data; Computational complexity

1 Introduction

Assignment problem belongs to the class of *matching problems* in which we wish to find the best way to pair objects to achieve a desired goal. For a full review of assignment models we refer the reader to [9]. In the assignment problem the objects are separated into two groups W and V and we wish to assign the objects from V to the objects from W. A typical example is the assignment of n jobs to n machines or n tasks to n workers. For each $i \in V$ and $j \in W$ there is given a cost c_{ij} of assigning i to j. We seek an assignment for which the sum of costs or the maximal cost is minimal. The first objective function corresponds to the *linear sum assignment* problem, while the second one corresponds to the *linear bottleneck assignment* problem. The assignment problem can be modeled as the weighted matching problem in a bipartite graph. Due to its special structure, some polynomial algorithms have been developed, which can solve the problem very efficiently. Several algorithms have been proposed for the linear sum assignment problem. The oldest one is the Hungarian algorithm with time complexity $O(n^3)$ developed by Kuhn [16]. A wide review of different algorithms for that problem

*Corresponding author

can be found in [1] and [9]. The linear bottleneck assignment problem was introduced by Fulkerson et al. [10]. Gabow and Tarjan [11] constructed an algorithm for that problem with time complexity $O(m\sqrt{n\log n})$, where m is the number of edges in the corresponding bipartite graph.

Linear assignments problems establish a class of well known and efficiently solvable problems. In the classical problems it is assumed that all the costs are precisely known. This assumption may be often a serious restriction since for many real-world processes the exact values of the costs are not known in advance. In the industrial applications the costs often vary and it is more convenient to define them as intervals (ranges of possible values) rather than exact values. If the costs are not precisely known, then the value of the objective function is also imprecise and an additional criterion is required to calculate the optimal solution. In the paper we adopt an approach to modeling the imprecision, called the *robust* approach. A comprehensive treatment of the state of art in robust optimization can be found in [15]. In the robust approach the set of all the possible realizations of costs, called *scenarios*, is given. The objective is to find a solution which minimizes the "worst-case" performance over all scenarios. When choosing among robust solutions, several criteria can be used. Kouvelis and Yu [15] studied three robustness criteria, namely *absolute robustness*, *robust deviation* and *relative robustness*. In this paper we apply the robust deviation criterion called also the *minimax regret*. Using this criterion we seek a solution for which the maximal possible deviation from optimum is minimal.

The minimax regret approach has been already applied to some combinatorial optimization problems with interval parameters. In [2], [3] and [20] it has been applied to the minimal spanning tree problem, in [18] to the shortest path problem and in [7] and [8] to some location problems. There are also some results on sequencing (see [5], [14] and [15]). In [13] the minimax regret approach to the linear programming problem with interval cost coefficients is presented. Most of these papers have appeared recently so the minimax regret approach to optimization is relatively new. It seems to be an interesting alternative to some other methods of dealing with imprecision, especially to the stochastic approach. In the stochastic approach the parameters of the problem are modeled as random variables. But it is often expensive or even impossible to assume any specific probability distribution for the unknown parameters. In such a situation it may be more convenient to define the parameters as intervals and apply the minimax regret criterion, which has simple and natural interpretation.

This paper is organized as follows. In Section 2 we present the formulation of the problem. In Section 3 we show that the linear bottleneck assignment problem with the minimax regret criterion can by solved in a polynomial time. In Section 4 we study the linear sum assignment problem with the minimax regret criterion. We prove that this problem is $\mathcal{NP}$-hard. A consequence of this result is that the linear sum transportation problem with the minimax regret criterion is also $\mathcal{NP}$-hard. In the next part of this section we propose the mixed integer model for the robust linear sum assignment problem and we carry out some computational experiments.

2 Problem formulation

Let $G = (V,W;E)$ be a bipartite graph, where the vertex sets V and W have n vertices and $|E| = m$. An *assignment* $\boldsymbol{A}$ is a *perfect matching* in G, i.e. $\boldsymbol{A}$ is a subset of the edges of G such that each element in V is paired with exactly one element in W (see Figure 1). We will denote by Φ the set of all the assignments in G. For each edge

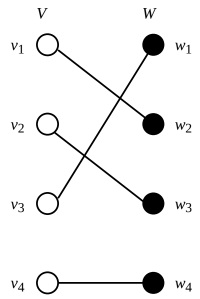

Figure 1: An example of the assignment

$e \in E$ there is given an interval $[\underline{c}_e, \overline{c}_e]$, which expresses an uncertain cost associated with the edge e. A *cost scenario* (shortly *scenario*) S is a particular realization of the costs $c_e^S \in [\underline{c}_e, \overline{c}_e]$, $e \in E$. We will denote by Γ the set of all the possible scenarios in the problem. Let $F(\boldsymbol{A},S)$ denote the cost of an assignment $\boldsymbol{A}$ under scenario $S \in \Gamma$.

In this paper we consider the following two cost functions:

$$\textbf{BA:} \quad F(\boldsymbol{A},S) = \max_{e\in\boldsymbol{A}}\{c_e^S\}, \tag{1}$$

$$\textbf{LA:} \quad F(\boldsymbol{A},S) = \sum_{e\in\boldsymbol{A}} c_e^S. \tag{2}$$

The cost function **BA** corresponds to the linear bottleneck assignment problem, while the cost function **LA** corresponds to the linear sum assignment problem. Let us define:

$$F^*(S) = \min_{\boldsymbol{A}\in\Phi} F(\boldsymbol{A},S), \tag{3}$$

thus $F^*(S)$ is the cost of an optimal assignment under a fixed scenario S. Let us define:

$$Z(\boldsymbol{A}) = \max_{S\in\Gamma}\{F(\boldsymbol{A},S) - F^*(S)\}. \tag{4}$$

The value of $Z(\boldsymbol{A})$ is called *the maximal regret* for the assignment $\boldsymbol{A}$. The scenario $S^0 \in \Gamma$ which maximizes the right hand side of (4) is called *the worst case scenario* for $\boldsymbol{A}$ and the optimal assignment $\boldsymbol{A}^* \in \Phi$ for the scenario S^0 is called *the worst case alternative* for $\boldsymbol{A}$. Thus, the maximal regret for $\boldsymbol{A}$ can be also expressed as follows:

$$Z(\boldsymbol{A}) = F(\boldsymbol{A},S^0) - F(\boldsymbol{A}^*,S^0). \tag{5}$$

In the paper we focus on the following optimization problem:

$$\min\{Z(\boldsymbol{A}) : \boldsymbol{A} \in \Phi\}, \tag{6}$$

so we look for an assignment for which the maximal regret is minimal. The optimization problem (6) is called *the robust assignment problem* and the optimal solution of (6) is called *the optimal robust assignment*. The problem with the cost function of form (1) is called *the robust linear bottleneck assignment problem (RLBA)* and the problem with the cost function of form (2) is called *the robust linear sum assignment problem (RLSA)*. The robust assignment problem can be viewed as a generalization of the classical assignment problem. If all costs are deterministic (i.e. $\underline{c}_e = \overline{c}_e$ for all $e \in E$), then there is only one scenario S and problem (6) reduces itself to the calculation of the optimal assignment for scenario S.

3 The robust linear bottleneck assignment problem

In this section we propose a polynomial time algorithm for solving the problem *RLBA*. Let $\boldsymbol{A} \in \Phi$ be a given assignment and let $S \in \Gamma$ be a given scenario. An edge $f \in \boldsymbol{A}$ is called *critical* in $\boldsymbol{A}$ under S if

$$c_f^S = \max_{e \in \boldsymbol{A}} c_e^S = F(\boldsymbol{A}, S).$$

Thus, the edge f has the greatest cost in $\boldsymbol{A}$ under S. Let S^e, $e \in E$, denote the scenario in which the cost of the edge e is equal to $\overline{c}_e$ and the costs of all the other edges $g \in E \setminus \{e\}$ are equal to $\underline{c}_g$. The following proposition characterizes the worst case scenario for a given assignment $\boldsymbol{A}$:

Proposition 1. *For every assignment $\boldsymbol{A}$ there exists an edge $f \in \boldsymbol{A}$ such that*

1. *scenario S^f is the worst case scenario for $\boldsymbol{A}$,*
2. *the edge f is critical in $\boldsymbol{A}$ under S^f.*

Proof. Let S^0 be a worst case scenario for $\boldsymbol{A}$ and let $f \in \boldsymbol{A}$ be a critical edge in $\boldsymbol{A}$ under S^0. Let $\boldsymbol{A}^*$ be the worst case alternative for $\boldsymbol{A}$ under S^0. From (5) we have the maximal regret for $\boldsymbol{A}$

$$Z(\boldsymbol{A}) = F(\boldsymbol{A}, S^0) - F(\boldsymbol{A}^*, S^0) = c_f^{S^0} - F(\boldsymbol{A}^*, S^0). \tag{7}$$

Consider the scenario S^f. We see at once that the edge f is also critical in $\boldsymbol{A}$ under S^f, since we increase the cost of the critical edge f and decrease the costs of all the other edges in S^0. It remains to show that S^f is the worst case scenario for $\boldsymbol{A}$. This follows immediately from the fact that value $F(\boldsymbol{A}, S^0) - F(\boldsymbol{A}^*, S^0)$ cannot decrease if we replace S^0 with S^f (see (7)). $\square$

Let $\boldsymbol{A}$ be a given assignment and let S^f, $f \in E$, be the worst case scenario for $\boldsymbol{A}$, which satisfies Conditions 1–2 of Proposition 1. Since f is critical in $\boldsymbol{A}$ under S^f we have $F(\boldsymbol{A}, S^f) = \overline{c}_f$, and since S^f is the worst case scenario for $\boldsymbol{A}$ we obtain:

$$Z(\boldsymbol{A}) = F(\boldsymbol{A}, S^f) - F^*(S^f) = \overline{c}_f - F^*(S^f). \tag{8}$$

Proposition 2. *The maximal regret for $\boldsymbol{A}$ can be expressed as follows:*

$$Z(\boldsymbol{A}) = \max_{e \in \boldsymbol{A}} \{\overline{c}_e - F^*(S^e)\}.$$

Proof. From (8) we get:

$$Z(\boldsymbol{A}) \leq \max_{e\in\boldsymbol{A}}\{\overline{c}_e - F^*(S^e). \tag{9}$$

Suppose by contradiction that there exists $g \in \boldsymbol{A}$ such that $Z(\boldsymbol{A}) < \overline{c}_g - F^*(S^g)$. Since $F(\boldsymbol{A}, S^g) \geq \overline{c}_g$ we see that $Z(\boldsymbol{A}) < F(\boldsymbol{A}, S^g) - F^*(S^g)$ which contradicts the definition of $Z(\boldsymbol{A})$. We thus get that in (9) the equality must hold and the proof is completed. □

Accordingly, let us define $c^*_e = \overline{c}_e - F^*(S^e)$, $e \in E$. Observe that the value of c^*_e does not depend on assignment $\boldsymbol{A}$. Thus, the optimal robust assignment can be determined by solving the following optimization problem:

$$\min\{\max_{e\in\boldsymbol{A}}\{c^*_e\} : \boldsymbol{A} \in \Phi\}. \tag{10}$$

Problem (10) is the classical linear bottleneck assignment problem. For the given costs c^*_e, $e \in E$, it can be solved in $O(m\sqrt{n \log n})$ time [11]. Hence, the calculation of all the cost c^*_e, $e \in E$, is more time consuming since we have to execute the algorithm calculating $F^*(S^e)$ for each $e \in E$, which takes $O(m^2\sqrt{n \log n})$ time.

Now we show how to reduce the complexity of the calculations. Let $\underline{S}$ be the scenario in which the costs of all the edges $e \in E$ are equal to their lower bounds, i.e. $c^{\underline{S}}_e = \underline{c}_e$ for $e \in E$. Let $\underline{\boldsymbol{A}}$ be the optimal assignment under $\underline{S}$ i.e. $F^*(\underline{S}) = F(\underline{\boldsymbol{A}}, \underline{S})$. The following proposition is true:

Proposition 3. *For every edge $e \in E \setminus \underline{\boldsymbol{A}}$ equality $F^*(S^e) = F^*(\underline{S})$ holds.*

Proof. From the definitions of $\underline{S}$ and S^e and the fact that the costs of all the edges in $\underline{S}$ are not greater than in S^e, $e \in E$, we get $F^*(S^e) \geq F^*(\underline{S})$. Since $e \notin \underline{\boldsymbol{A}}$, we conclude that $F^*(S^e) \leq F(\underline{\boldsymbol{A}}, S^e) = F(\underline{\boldsymbol{A}}, \underline{S}) = F^*(\underline{S})$, and thus $F^*(S^e) = F^*(\underline{S})$. This completes the proof. □

Proposition 3 allows to decrease the complexity of the algorithm for solving the problem *RLBA*. In order to determine c^*_e, for each $e \in E$, one need not calculate the value of $F^*(S^e)$ for each $e \in E$. It is enough to calculate the value of $F^*(\underline{S})$ and the values of $F^*(S^e)$ for $e \in \underline{\boldsymbol{A}}$. Therefore, the algorithm for solving problem *RLBA* requires $O(nm\sqrt{n \log n})$ time instead of $O(m^2\sqrt{n \log n})$. Its implementation is presented in the form of Algorithm 1.

4 The robust linear sum assignment problem

In this section we consider the problem *RLSA*, i.e. the robust assignment problem with the cost function of form (2). The following proposition characterizes the worst case scenario for a given assignment $\boldsymbol{A}$ in this problem:

Proposition 4. *For every assignment $\boldsymbol{A}$ there exists a worst case scenario, such that under this scenario:*

1. *the costs of all the edges $e \in \boldsymbol{A}$ are equal to $\overline{c}_e$,*
2. *the cost of all the edges $e \in E \setminus \boldsymbol{A}$ are equal to $\underline{c}_e$.*

Algorithm 1 Algorithm for solving the problem *RLBA*

Require: $G = (V, W; E)$, $[\underline{c}_e, \overline{c}_e]$ for each $e \in E$
Ensure: The optimal robust assignment $\boldsymbol{A}$

```
Solve the bottleneck assignment problem for scenario S̲. Denote the optimal solu-
  tion as A̲ and denote the cost of A̲ under S̲ by C.
for all e ∈ E \ A̲ do
    c*_e ← c̄_e − C
end for
for all e ∈ A̲ do
    compute F*(S^e) by solving the bottleneck assignment for scenario S^e.
    c*_e ← c̄_e − F*(S^e)
end for
Solve the bottleneck assignment problem for G = (V, W; E) with the costs c*_e, e ∈
  E. Denote the optimal solution by A.
return A
```

Proof. Let S^0 be a worst case scenario for a given assignment $\boldsymbol{A}$. Let $\boldsymbol{A}^*$ be the worst case alternative for $\boldsymbol{A}$ under S^0. It holds:

$$Z(\boldsymbol{A}) = \sum_{e \in \boldsymbol{A} \setminus \boldsymbol{A}^*} c_e^{S^0} - \sum_{e \in \boldsymbol{A}^* \setminus \boldsymbol{A}} c_e^{S^0}. \tag{11}$$

Consider the scenario S in which $c_e^S = \overline{c}_e$ for all $e \in \boldsymbol{A}$ and $c_e^S = \underline{c}_e$ for all $e \in E \setminus \boldsymbol{A}$. Note, that S fulfills Conditions 1 and 2 of the proposition. What is left is to show that S is the worst case scenario for $\boldsymbol{A}$. From (11) we obtain:

$$Z(\boldsymbol{A}) \leq \sum_{e \in \boldsymbol{A} \setminus \boldsymbol{A}^*} \overline{c}_e - \sum_{e \in \boldsymbol{A}^* \setminus \boldsymbol{A}} \underline{c}_e = F(\boldsymbol{A}, S) - F(\boldsymbol{A}^*, S) \leq F(\boldsymbol{A}, S) - F^*(S),$$

and thus S is the worst case scenario for $\boldsymbol{A}$. $\square$

Let $\boldsymbol{A}$ be a given assignment and let $S^{\boldsymbol{A}}$ be the worst case scenario for $\boldsymbol{A}$ such that $S^{\boldsymbol{A}}$ satisfies Conditions 1 and 2 of Proposition 4. The maximal regret for $\boldsymbol{A}$ can be expressed as follows:

$$Z(\boldsymbol{A}) = F(\boldsymbol{A}, S^{\boldsymbol{A}}) - F^*(S^{\boldsymbol{A}}) = \sum_{e \in \boldsymbol{A}} \overline{c}_e - F^*(S^{\boldsymbol{A}}). \tag{12}$$

Equality (12) allows to calculate the value of the maximal regret for a given assignment $\boldsymbol{A}$ in polynomial time. The value of $F^*(S^{\boldsymbol{A}})$ can be determined by solving the classical linear sum assignment problem for the fixed scenario $S^{\boldsymbol{A}}$.

4.1 The computational complexity of *RLSA*

In this section, we prove that the problem *RLSA* is $\mathcal{NP}$-hard by showing that a decision problem related to it is $\mathcal{NP}$-complete. The robust linear sum assignment decision problem, Decision-*RLSA*, is defined as follows:
INPUT: A bipartite graph $G = (V, W, E)$, $|V| = |W| = n$, costs associated with the edges $e \in E$ are determined by means of intervals $[\underline{c}_e, \overline{c}_e]$ (with integer bounds), $\underline{c}_e \geq 0$, and a nonnegative integer D.

QUESTION: Is there an assignment $\boldsymbol{A}$ in G such that $Z(\boldsymbol{A}) \leq D$?

Our proof of the $\mathcal{NP}$-completeness of Decision-*RLSA* consists in reducing to it a certain modified PARTITION problem, called MPARTITION.

The MPARTITION problem is defined as follows:

INPUT: A finite set $\mathcal{A}$ of positive integers, $\mathcal{A} = \{a_1, \ldots, a_q\}$, having the overall sum of $2b$ and a positive integer $K < q$.

QUESTION: Is there a subset $\mathcal{A}^{'} \subset \mathcal{A}$ that sums up exactly to b and $|\mathcal{A}^{'}| = K$?

It is well known that MPARTITION is $\mathcal{NP}$-complete (see for instance [12] and comments on PARTITION given there).

Theorem 1. *Decision-RLSA is $\mathcal{NP}$-complete.*

Proof. Suppose that we are given an instance of MPARTITION. We show a polynomial time reduction to an instance of Decision-*RLSA*, such that the answer for the instance of Decision-*RLSA* is "yes" if and only if the answer for the corresponding instance of MPARTITION is also "yes".

The reduction proceeds as follows. To each instance of MPARTITION, we associate a graph $G^{'} = (V^{'}, W^{'}, E^{'})$. The graph $G^{'}$ consists of one "gadget" $G^{(i)} = (V^{(i)}, W^{(i)}, E^{(i)})$, $V^{(i)} \subset V^{'}$, $W^{(i)} \subset W^{'}$, $E^{(i)} \subset E^{'}$, per each element $a_i \in \mathcal{A}$, $i = 1, \ldots, q$. See Figure 2a for gadget $G^{(i)}$ corresponding to $a_i \in \mathcal{A}$. The edge costs are intervals.

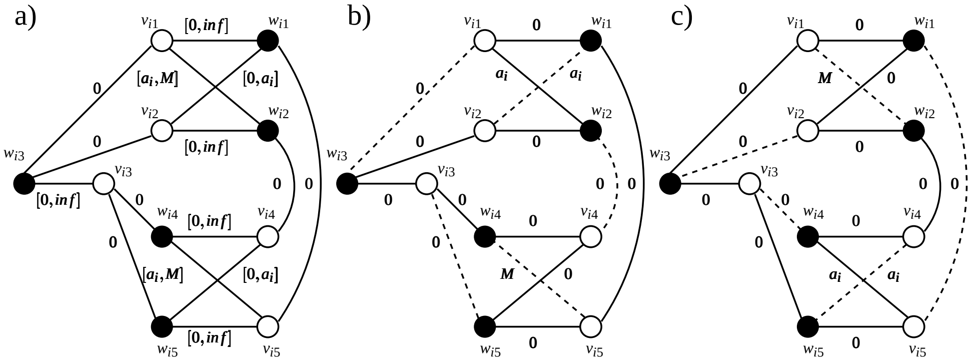

Figure 2: Gadget $G^{(i)}$ corresponding to element $a_i \in \mathcal{A}$.

The one-point intervals are presented in Figure 2a as precise numbers. M is a number such that $M > 2b$, set $M = 2b + 1$, inf is a large number such that $M \ll inf$, set $inf = q^4 M$. The white nodes belong to $V^{(i)}$ and black ones belong to $W^{(i)}$. It is clear that the gadget $G^{(i)}$ is a bipartite graph. The interval costs are chosen in such a way that $G^{(i)}$ has the following property.

Property 1. *Let $\boldsymbol{A}^{(i)}$ be an assignment in $G^{(i)}$. If $\boldsymbol{A}^{(i)}$ contains an edge with $\overline{c}_e = inf$ then $Z(\boldsymbol{A}^{(i)}) \geq inf$. Otherwise $Z(\boldsymbol{A}^{(i)}) = a_i + M$.*

Moreover, there exist only two assignments in $G^{(i)}$ with the maximal regret equal to $a_i + M$. See Figure 2b, c for these two assignments marked with a dashed line and their worst case scenarios.

Graph $G^{'}$ consists of nested gadgets $G^{(i)}$, $i = 1, \ldots, q$. We denote by v_{ij} node j in set $V^{(i)}$ and by w_{ij} node j in set $W^{(i)}$ in gadget $G^{(i)}$. To each element $a_i \in \mathcal{A}$, we associate gadget $G^{(i)}$, $i = 1, \ldots, q$ (see Figure 3) and then we link them together as follows. For

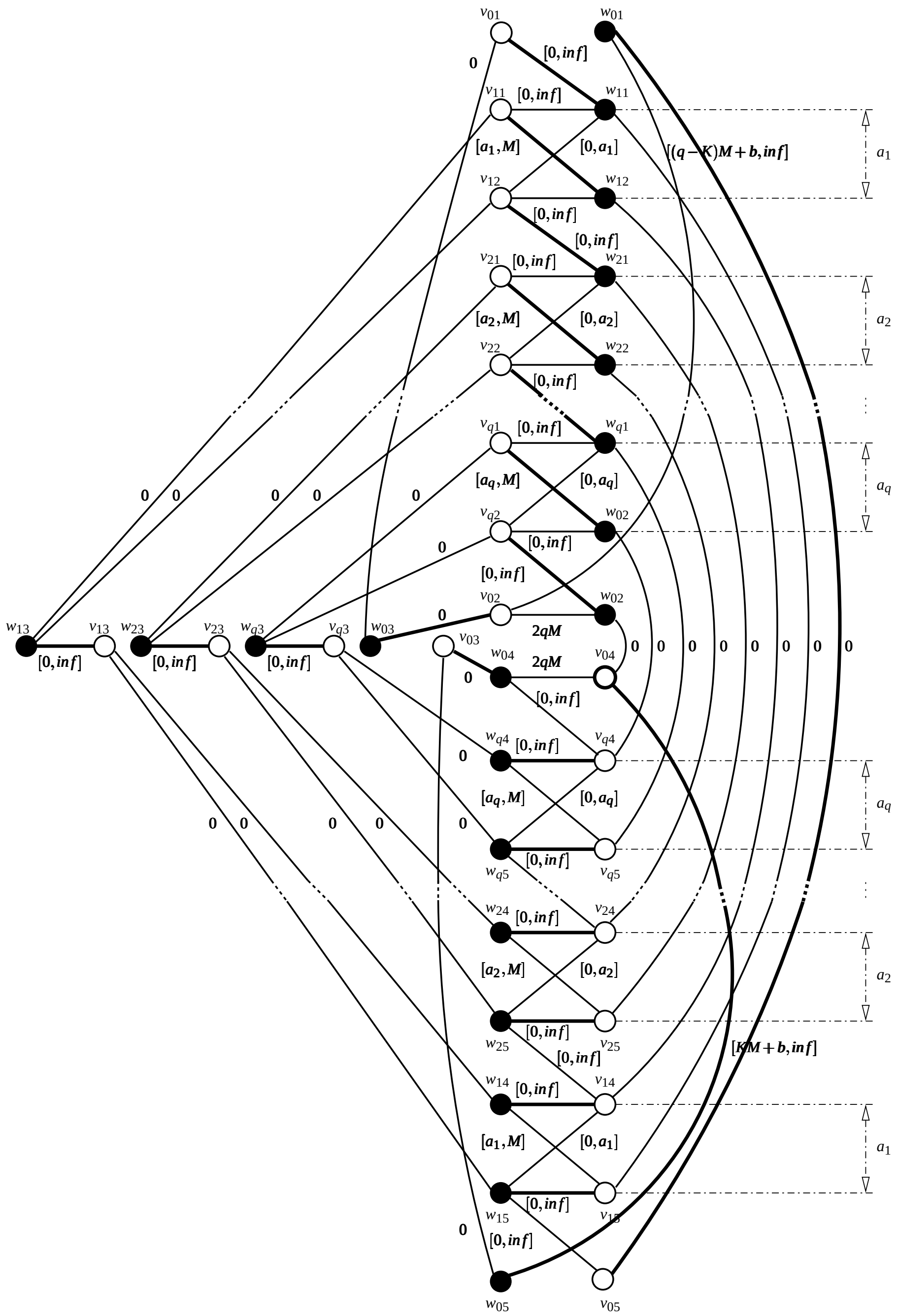

Figure 3: Graph G'

$i = 1,\dots,q-1$, we add edges (v_{i2}, w_{i+11}), (v_{i4}, w_{i+15}) with costs of $[0, inf]$. Finally, we add edges (v_{01}, w_{11}), (v_{q2}, w_{02}), (v_{q4}, w_{04}), (v_{05}, w_{15}) with interval costs $[0, inf]$, edges (v_{01}, w_{03}), (v_{02}, w_{03}), (v_{03}, w_{04}), (v_{03}, w_{05}) with interval cost $[0,0]$, edges (v_{02}, w_{02}), (v_{04}, w_{04}) with interval cost $[2qM, 2qM]$, and edges (v_{02}, w_{01}) and (v_{04}, w_{05}) having interval costs $[(q-K)M+b, inf]$ and $[KM+b, inf]$, respectively. This completes the definition of the graph $G^{'} = (V^{'}, W^{'}, E^{'})$. The white and black nodes belong to $V^{'}$ and $W^{'}$, respectively. It is easily seen that $G^{'}$ is a bipartite graph. The construction of $G^{'}$ is done in time bounded by a polynomial in the size of MPARTITION. The parameter D is set to be $4qM$.

Note that only nodes w_{i1}, v_{i2}, v_{i4} and w_{i5} of gadget $G^{(i)}$ are connected with the other nodes of graph $G^{'}$ by only four edges having interval costs $[0, inf]$ (see Figure 3). This and Property 1 lead to the following property.

Property 2. *Any assignment $\boldsymbol{A}$ in $G^{'}$ such that $Z(\boldsymbol{A}) < inf$ must use either edges (v_{i1}, w_{i3}), (v_{i2}, w_{i1}), (v_{i3}, w_{i5}) (v_{i4}, w_{i2}), (v_{i5}, w_{i4}) (see Figure 2b) or edges (v_{i1}, w_{i2}), (v_{i2}, w_{i3}), (v_{i3}, w_{i4}), (v_{i4}, w_{i5}), (v_{i5}, w_{i1}) (see Figure 2c), for $i = 1,\dots,q$.*

Now we prove that there exists a subset $\mathcal{A}^{'} \subset \mathcal{A}$ which sums up exactly to b and $|\mathcal{A}^{'}| = K$ if and only if there exists an assignment $\boldsymbol{A}$ in $G^{'}$ such that $Z(\boldsymbol{A}) \leq 4qM$.
$\Longrightarrow$ Assume that there exists a subset $\mathcal{A}^{'} \subset \mathcal{A}$ that sums up exactly to b and $|\mathcal{A}^{'}| = K$. We will construct an assignment $\boldsymbol{A}$ with $Z(\boldsymbol{A}) = 4qM$.

From the construction of $G^{'}$, it follows that each element $a_i \in \mathcal{A}$, $i = 1,\dots,q$, corresponds to gadget $G^{(i)}$ (see Figure 3). If $a_i \in \mathcal{A}^{'}$, then we include edges (v_{i1}, w_{i3}), (v_{i2}, w_{i1}), (v_{i3}, w_{i5}) (v_{i4}, w_{i2}), (v_{i5}, w_{i4}) in assignment $\boldsymbol{A}$ (see Figure 2b). Otherwise $(a_i \in \mathcal{A} \setminus \mathcal{A}^{'})$, we include edges (v_{i1}, w_{i2}), (v_{i2}, w_{i3}), (v_{i3}, w_{i4}), (v_{i4}, w_{i5}), (v_{i5}, w_{i1}) in the assignment $\boldsymbol{A}$ (see Figure 2c). To complete $\boldsymbol{A}$ we add edges (v_{02}, w_{02}), (v_{04}, w_{04}), (v_{05}, w_{01}), (v_{01}, w_{03}), (v_{03}, w_{05}).

Consider scenario S^0 determined according to Proposition 4. We claim that cost $F(\boldsymbol{A}, S^0)$ of assignment $\boldsymbol{A}$ is $5qM + 2b$. Indeed, each element a_i, $i = 1,\dots,q$, belongs either to $\mathcal{A}^{'}$ or to $\mathcal{A} \setminus \mathcal{A}^{'}$. Subsets $\mathcal{A}^{'}$ and $\mathcal{A} \setminus \mathcal{A}^{'}$ sum up exactly to b and $|\mathcal{A}^{'}| = K$ and $|\mathcal{A} \setminus \mathcal{A}^{'}| = q - K$. So, the constructed assignment $\boldsymbol{A}$ uses K times the dashed edges shown in Figure 2b, this gives $KM + b$ and $q - K$ times the dashed edges shown in Figure 2c, this gives $(q-K)M + b$. The cost of edges (v_{02}, w_{02}), (v_{04}, w_{04}), (v_{05}, w_{01}), (v_{01}, w_{03}), (v_{03}, w_{05}) under scenario S^0 is equal to $4qM$. Hence, cost $F(\boldsymbol{A}, S^0) = 5qM + 2b$.

Now we show that the rest of assignments in $G^{'}$ have costs at least $qM + 2b$ under S^0. To do this, assume to the contrary that there exists an assignment $\boldsymbol{A}^{'}$ such that $F(\boldsymbol{A}^{'}, S^0) < qM + 2b$. Therefore, $\boldsymbol{A}^{'}$ does not contain edges (v_{02}, w_{02}) and (v_{04}, w_{04}) having costs $2qM$ $(M > 2b)$ and in consequence it contains either (v_{02}, w_{03}), (v_{03}, w_{04}), (v_{04}, w_{05}) and (v_{05}, w_{01}) or (v_{01}, w_{03}), (v_{02}, w_{01}), (v_{03}, w_{05}) and (v_{04}, w_{02}).

Consider the first case. The costs of edges (v_{02}, w_{03}), (v_{03}, w_{04}), (v_{04}, w_{05}) and (v_{05}, w_{01}) sum up to $KM + b$ under S^0. From the construction of $G^{'}$ and the fact that $\boldsymbol{A}^{'}$ is an assignment in $G^{'}$, it follows that $\boldsymbol{A}^{'}$ must use edges (v_{01}, w_{11}), (v_{q2}, w_{02}), edges (v_{i3}, w_{i3}), (v_{i4}, w_{i4}), (v_{i5}, w_{i5}), for $i = 1,\dots,q$, and edges $(v_{i-1\,2}, w_{i1})$ for $i = 2,\dots q$, with the costs of zero at their lower bounds ($\boldsymbol{A}^{'}$ is marked with a bold line in Figure 3). Assignment $\boldsymbol{A}^{'}$ must also contain edges (v_{i1}, w_{i2}), $i = 1,\dots,q$, with the edge costs K times at their lower bounds equal to a_i, when the corresponding element a_i belongs to $\mathcal{A}^{'}$, and $q - K$ times at their upper bounds equal to M, when the corresponding element

a_i belongs to $\mathcal{A} \setminus \mathcal{A}'$. Since $\mathcal{A}'$ sums up to b, the sum of these edges costs equals $(q-K)M+b$. Hence, the cost of $\boldsymbol{A}'$ is $qM+2b$. This contradicts our assumption.

The second case, when $\boldsymbol{A}'$ uses (v_{01},w_{03}), (v_{02},w_{01}), (v_{03},w_{05}) and (v_{04},w_{02}) is similar. Now $\boldsymbol{A}'$ is symmetrical to the assignment considered in the first case and also its cost is $qM+2b$. This contradicts our assumption.

We have thus proved that all the assignments in G' have costs at least $qM+2b$ under S^0. In particular, the assignment marked with a bold line in Figure 3, let us denote it by $\boldsymbol{A}^*$, has cost of exactly $qM+2b$ under S^0, i.e. $F(\boldsymbol{A}^*,S^0) = qM+2b$. Thus, $Z(\boldsymbol{A}) = F(\boldsymbol{A},S^0) - F(\boldsymbol{A}^*,S^0) = 4qM$.
$\Longleftarrow$ Let $\boldsymbol{A}$ be an assignments such that $Z(\boldsymbol{A}) = F(\boldsymbol{A},S^0) - F(\boldsymbol{A}^*,S^0) \leq 4qM$. The scenario S^0 is the worst case scenario for $\boldsymbol{A}$ determined by means of Proposition 4. We determine a subset $\mathcal{A}' \subset \mathcal{A}$, $|\mathcal{A}'| = K$, and show that it sums up exactly to b. From Property 2 and the construction of G', we deduce that the assignment $\boldsymbol{A}$ uses edges (v_{02},w_{02}), (v_{04},w_{04}), (v_{05},w_{01}), (v_{01},w_{03}), (v_{03},w_{05}) and either edges (v_{i1},w_{i3}), (v_{i2},w_{i1}), (v_{i3},w_{i5}) (v_{i4},w_{i2}), (v_{i5},w_{i4}) or (v_{i1},w_{i2}), (v_{i2},w_{i3}), (v_{i3},w_{i4}), (v_{i4},w_{i5}), (v_{i5},w_{i1}), for $i = 1,\dots,q$. So, under scenario S^0, the assignment $\boldsymbol{A}$ has cost $F(\boldsymbol{A},S^0)$ equal to $5qM+2b$. Since $F(\boldsymbol{A},S^0) - F(\boldsymbol{A}^*,S^0) \leq 4qM$, the cost of the optimal assignment $\boldsymbol{A}^*$ under S^0 is at least $qM+2b$.

Now we give a property of assignment $\boldsymbol{A}$.

Property 3. *The costs of edges* (v_{i1},w_{i2}) *and* (v_{i2},w_{i1}), $i = 1,\dots,q$, *such that either* (v_{i1},w_{i2}) *or* (v_{i2},w_{i1}) *belong to* $\boldsymbol{A}$ *have the overall sum of* $(q-K)M+b$ *under* S^0.

In order to prove Property 3, suppose to the contrary that the sum of costs is less than $(q-K)M+b$. Therefore, the sum of costs of edges (v_{i5},w_{i4}), if $(v_{i2},w_{i1}) \in \boldsymbol{A}$, or (v_{i4},w_{i5}), if $(v_{i1},w_{i2}) \in \boldsymbol{A}$, $i = 1,\dots,q$, is greater than $KM+b$. This implies the existence of an assignment $\boldsymbol{A}'$ such $F(\boldsymbol{A}',S^0) < qM+2b$. See Figure 3 for such assignment marked with a bold line. In consequence $Z(\boldsymbol{A}) > 4qM$. This contradicts our assumption. Similarly, assume that the costs of edges (v_{i1},w_{i2}) and (v_{i2},w_{i1}), $i = 1,\dots,q$, such that either (v_{i1},w_{i2}) or (v_{i2},w_{i1}) belong to $\boldsymbol{A}$ is greater than $KM+b$. The sum of costs of edges (v_{i5},w_{i4}), if $(v_{i2},w_{i1}) \in \boldsymbol{A}$, or (v_{i4},w_{i5}), if $(v_{i1},w_{i2}) \in \boldsymbol{A}$, $i = 1,\dots,q$, is less than $KM+b$. So, it follows that there exists an assignment $\boldsymbol{A}''$ such $F(\boldsymbol{A}'',S^0) < qM+2b$. This assignment $\boldsymbol{A}''$ is symmetrical to $\boldsymbol{A}'$. We arrive to a contradiction , since $Z(\boldsymbol{A}) > 4qM$ and the proof of Property 3 is complete.

We return to the main proof. Let us determine a subset $\mathcal{A}'$. If assignment $\boldsymbol{A}$ uses (v_{i1},w_{i3}), (v_{i2},w_{i1}), (v_{i3},w_{i5}) (v_{i4},w_{i2}), (v_{i5},w_{i4}) (see Figure 2b) then a_i is included in $\mathcal{A}'$, $i = 1,\dots,q$. Otherwise (it uses (v_{i1},w_{i2}), (v_{i2},w_{i3}), (v_{i3},w_{i4}), (v_{i4},w_{i5}), (v_{i5},w_{i1}), see Figure 2c) a_i is included in $\mathcal{A} \setminus \mathcal{A}'$. From Property 3, we obtain that the costs of edges (v_{i1},w_{i2}) and (v_{i2},w_{i1}), $i = 1,\dots,q$, such that either (v_{i1},w_{i2}) or (v_{i2},w_{i1}) belong to $\boldsymbol{A}$ sum up to $(q-K)M+b$, under S^0, M is a number such that $M > \sum_{i=1}^{q} a_i$, and only edges (v_{i1},w_{i2}) have costs M, $i = 1,\dots,q$. Thus, the assignment $\boldsymbol{A}$ must use $q-K$ times (v_{i1},w_{i2}) with the cost of M and K times (v_{i2},w_{i1}) with the cost of a_i. This gives $|\mathcal{A}'| = K$ and $\sum_{\{i \mid a_i \in \mathcal{A}', 1 \leq i \leq q\}} a_i = b$.

It remains to prove that Decision-*RLSA* is in $\mathcal{NP}$. A nondeterministic Turing machine first “guesses” a matching $\boldsymbol{A}$ and then verifies $\boldsymbol{A}$, i.e. it checks: whether $\boldsymbol{A}$ is a perfect matching (assignment) in G and $F(S^0,\boldsymbol{A}) - F(S^0,\boldsymbol{A}^*) \leq D$ ($Z(\boldsymbol{A}) \leq D$) in the worst case scenario S^0 determined as in Proposition 4. The verification can be done in time bounded by a polynomial in the size of G as it amounts to solving the classical assignment problem in G where the edge costs are determined by scenario S^0. Thus, Decision-*RLSA* is $\mathcal{NP}$-Complete. $\square$

From Theorem 1 we immediately obtain the computational complexity of *RLSA*.

Corollary 1. *RLSA is $\mathcal{NP}$-hard.*

4.2 The robust linear sum transportation problem – a generalization of *RLSA*

The problem *RLSA* is a basic combinatorial problem which turns out to be $\mathcal{NP}$-hard. It follows immediately that all the generalizations of *RLSA* are also $\mathcal{NP}$-hard. Consider the following *linear sum transportation problem*. We are given m suppliers $V_1,\ldots,V_m$. Each supplier V_i has s_i units of goods. These goods must be delivered to n customers $W_1,\ldots,W_n$, where each customer W_j requires d_j units of goods. For each supplier V_i and customer W_j there is given a cost c_{ij} of sending one unit of good from V_i to W_j. The solution to this problem can be represented by a matrix $\boldsymbol{X} = [x_{ij}]_{m\times n}$, where x_{ij} denotes the amount of goods sent from V_i to W_j. We seek a solution $\boldsymbol{X} = [x_{ij}]_{m\times n}$ for which the value of $F(\boldsymbol{X}) = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$ is minimal. The problem can be efficiently solved and some polynomial algorithms can be found in [1].

The linear sum transportation problem can be modeled as a network flow problem in a bipartite graph $G = (V,W;E)$, where V represents the set of suppliers and W represents the set of customers. With each edge $e = (i,j) \in E$ we associate cost $c_e = c_{ij}$. Note that the linear sum assignment problem can be viewed as a special case of the linear sum transportation problem in which $n = m$, $s_i = 1$ and $d_i = 1$ for all $i = 1,\ldots,n$.

Assume now, that the costs in the problem may be imprecise and they are given as intervals $[\underline{c}_e, \overline{c}_e]$, $e \in E$. For a given cost scenario $S \in \Gamma$ the value of $F(S,\boldsymbol{X})$ denotes the cost of solution $\boldsymbol{X}$ under S and the value of $F^*(S)$ denotes the cost of the optimal solution under S. The maximal regret for $\boldsymbol{X}$ is defined as follows:

$$Z(\boldsymbol{X}) = \max_{S\in\Gamma}\{F(S,\mathbf{X}) - F^*(S)\}.$$

The *robust linear sum transportation problem*, shortly *RLST*, consists in calculating solution $\boldsymbol{X}$ for which the value of maximal regret $Z(\boldsymbol{X})$ is minimal.

Theorem 2. *The problem RLST is $\mathcal{NP}$-hard*

Proof. If we set $n = m$, $s_i = 1$ and $d_i = 1$ for all $i = 1,\ldots,n$, in the problem *RLST*, then we get the problem *RLSA* as the special case. Since the problem *RLSA* is $\mathcal{NP}$-hard it follows immediately that the more general problem *RLST* is also $\mathcal{NP}$-hard. $\square$

4.3 The MIXIP model for *RLSA*

The problem *RLSA* can be formulated as a mixed integer model (MIXIP). Assume that all the elements in set W (and similarly in V) are numbered from 1 to n. Let us introduce binary variables $x_{ij} \in \{0,1\}$, $i = 1,\ldots,n$, $j = 1,\ldots,n$, such that $x_{ij} = 1$ if and only if element $i \in W$ is assigned to element $j \in V$. Each assignment $\boldsymbol{A}$ can be represented as a matrix $\boldsymbol{X} = [x_{ij}]_{n\times n}$, whose elements fulfill the assignment constraints.

For a given scenario $S \in \Gamma$ the optimal assignment corresponds to the optimal solution of the following problem:

$$\begin{array}{ll} \sum_{i=1}^n \sum_{j=1}^n c_{ij}^S x_{ij} \to \min & \\ \sum_{i=1}^n x_{ij} = 1 & \text{for} \quad j = 1,\dots,n, \\ \sum_{j=1}^n x_{ij} = 1 & \text{for} \quad i = 1,\dots,n, \\ x_{ij} \in \{0,1\} & \text{for} \quad i,j = 1,\dots,n. \end{array} \tag{13}$$

It is well known that the following model has the same minimal objective value as (13):

$$\begin{array}{ll} \sum_{i=1}^n \sum_{j=1}^n c_{ij}^S x_{ij} \to \min & \\ \sum_{i=1}^n x_{ij} = 1 & \text{for} \quad j = 1,\dots,n, \\ \sum_{j=1}^n x_{ij} = 1 & \text{for} \quad i = 1,\dots,n, \\ x_{ij} \geq 0 & \text{for} \quad i,j = 1,\dots,n. \end{array} \tag{14}$$

The dual model to (14) contains $2n$ unrestricted variables and takes the following form:

$$\begin{array}{ll} \sum_{i=1}^{2n} y_i \to \max & \\ y_i + y_{n+j} \leq c_{ij}^S & \text{for} \quad i,j = 1,\dots,n, \end{array} \tag{15}$$

The minimal objective value of (14) (as well as (13)) equals the maximal objective value of (15). Assume that an assignment $\boldsymbol{A}$ is represented by a given matrix $\boldsymbol{X} = [x_{ij}]_{n\times n}$. Let $S^{\boldsymbol{A}}$ be the worst case scenario for $\boldsymbol{A}$ (see Proposition 4). Then, it holds:

$$F(\boldsymbol{A}, S^{\boldsymbol{A}}) = \sum_{i=1}^n \sum_{j=1}^n \overline{c}_{ij} x_{ij}.$$

From (15) and the definition of $S^{\boldsymbol{A}}$ it follows that the value of $F^*(S^{\boldsymbol{A}})$ is the maximal objective value of the following problem:

$$\begin{array}{ll} \sum_{i=1}^{2n} y_i \to \max & \\ y_i + y_{n+j} \leq \overline{c}_{ij} x_{ij} + \underline{c}_{ij}(1 - x_{ij}) & \text{for} \quad i,j = 1,\dots,n. \end{array}$$

Assignment $\boldsymbol{A}$ which minimizes the value of $Z(\boldsymbol{A}) = F(\boldsymbol{A}, S^{\boldsymbol{A}}) - F^*(S^{\boldsymbol{A}})$ can be then obtained by means of the following MIXIP model:

$$\begin{array}{ll} \sum_{i=1}^n \sum_{j=1}^n \overline{c}_{ij} x_{ij} - \sum_{i=1}^{2n} y_i \to \min & \\ \sum_{i=1}^n x_{ij} = 1 & \text{for} \quad j = 1,\dots,n, \\ \sum_{j=1}^n x_{ij} = 1 & \text{for} \quad i = 1,\dots,n, \\ y_i + y_{n+j} \leq \overline{c}_{ij} x_{ij} + \underline{c}_{ij}(1 - x_{ij}) & \text{for} \quad i,j = 1,\dots,n, \\ x_{ij} \in \{0,1\} & \text{for} \quad i,j = 1,\dots,n. \end{array} \tag{16}$$

Assume that $(\boldsymbol{X},\boldsymbol{y})$ is an optimal solution of (16). Then, $\boldsymbol{X}$ represents the optimal robust assignment and the value of the objective function of (16) is equal to the value of the maximal regret for the optimal assignment. Model (16) contains n^2 binary variables and $2n$ unrestricted real variables. If we replace the constraints $x_{ij} \in \{0,1\}$ with $x_{ij} \geq 0$ in (16), then we get the relaxation of the problem, which can be efficiently solved. Solving the relaxation we get the lower bound for the optimal robust assignment.

4.4 Some computational experiments

In order to solve the mixed integer model (16) we have used the commercial solver *ILOG OPL Studio 3.6.1* (http://www.ilog.com) and a computer equipped with Pentium III 866MHz processor, 2GB RAM. We have computed the optimal robust assignment for n=20, 50, 80, 110, 140, 160, 180, 200. The interval costs are generated as follows: for each edge $e \in E$ the value of $\underline{c}_e$ is uniformly distributed in the interval $[0, C]$ and $\overline{c}_e$ is uniformly distributed in the interval $[\underline{c}_e, \underline{c}_e + C]$. The fixed parameter C specifies the variability of the costs in the problem. For each problem size n, we have generated six sets of problems S^n_C, in which the value of C is equal to 10, 30, 50, 70, 100 and 150 respectively. Each set S^n_C, $C \in \{10, 30, 50, 70, 100, 150\}$, $n \in \{20, 50, 80, 110, 140, 160, 180, 200\}$ contains five randomly generated problems. The average computational times in CPU seconds for each set S^n_C are presented in Table 1. It is easy to observe that the average computational time strongly depends on the value of C. The greater is the variability of the costs C the longer is the average computational time.

$C \setminus n$	20	50	80	110	140	160	180	200
10	1.3	10.1	4.7	13.5	23.1	42.2	55.7	65.9
30	1.6	37.0	137.9	242.6	299.8	443.4	419.8	400.0
50	1.2	43.6	203.8	519.0	1309.7	1495.5	1305.3	1316.5
70	1.9	54.0	255.9	512.9	2029.8	2005.5	2689.6	3978.1
100	1.5	61.4	195.7	490.3	1644.9	5133.8	5178.4	8689.2
150	1.4	68.1	245.4	937.9	1604.7	3747.7	7106.9	11665.2

Table 1: The results of the experiments (the average computational times in CPU seconds for each set of problems).

5 Conclusions

In this paper, we have examined the linear assignment problems with interval costs. In order to calculate the optimal solution we have applied the robust criterion (the minimax regret criterion). We have shown that the robust version of problem with the bottleneck objective function (*RLBA*) can be solved in $O(nm\sqrt{n \log n})$ time. We have also proven that the robust linear sum assignment problem (*RLSA*) is $\mathcal{NP}$-hard. As a consequence the robust linear sum transportation problem, being the generalization of *RLSA*, turns out to be $\mathcal{NP}$-hard. We have formulated a mixed integer model (MIXIP) for the problem *RLSA* and we have carried out some computational experiments. The next subject of research would involve the development of efficient algorithms for problem *RLSA*. The MIXIP model (16) can be a good starting point for some algorithms based on the branch and bound method.

References

[1] R. K. Ahuja, T. L. Magnanti, J. B. Orlin, *Network Flows, Theory, Algorithms and Applications*, Prentice Hall, New Jersey 1993.

[2] I. Aron, P. Van Hentenryck, A Constraint Satisfaction Approach to the Robust Spanning Tree with Interval Data, *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, Edmonton, Canada, August, 2002.

[3] I. Aron, P. Van Hentenryck, On the Complexity of the Robust Spanning Tree with Interval Data, *Operations Research Letters* 32 (2004) 36–140.

[4] I. Averbakh,O. Berman, Algorithms for the robust 1-center problem on a tree, *European Journal of Operational Research* 123 (2000) 292–302.

[5] I. Averbakh, Minimax regret solutions for minimax optimization problems with uncertainty, *Operations Research Letters* 27 (2000) 57–65.

[6] I. Averbakh, On the complexity of a class of combinatorial optimization problems with uncertainty, *Mathematical Programming* 25 (2001) 263–272.

[7] I. Averbakh, Complexity of robust single facility location problems on networks with uncertain edge lengths, *Discrete Applied Mathematics* 127 (2003) 505–522.

[8] I. Averbakh, Minimax regret linear resource allocation problems, *Operations Research Letters* 32 (2004) 174–180.

[9] R.E. Burkard, Selected topics on assignment problems *Discrete Applied Mathematics* 123 (2000) 257–302.

[10] R. Fulkerson, I. Glicksberg, O. Gross, A production line assignment problem, Technical Report RM-1102, The Rand Corporation, Sta. Monica, CA, 1953.

[11] H. N. Gabow, R. E. Tarjan, Algorithms for two bottleneck optimization problems, *J. Algorithms* 9 (1988), 411–417.

[12] M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freemam and Co., San Francisco, 1979.

[13] M. Inuiguchi, M. Sakawa, Minimax regret solution to linear programming problems with an interval objective function, *European Journal of Operational Research* 86 (1995) 526–536.

[14] A. Kasperski, Minimizing maximal regret in sequencing problem with the maximum lateness criterion, submitted to *Operations Research Letters*.

[15] P. Kouvelis, G. Yu, *Robust Discrete Optimization and its applications*, Kluwer Academic Publishers, Boston, 1997.

[16] H. W. Kuhn, The Hungarian method for the assignment problem, *Naval Research Logistics Quarterly* 2 (1955) 83–97.

[17] C.J. Lin, U.P. Wen, Sensitivity analysis of the optimal assignment, *European Journal of Operational Research* 149 (2003) 35–46.

[18] R. Montemanni, L.M Gambardella, A.V. Donati, A branch and bound algorithm for the robust shortest path problem with interval data, *Operations Research Letters* 32 (2004) 225–232.

[19] J.M. Mulvey, R.J. Vanderbei, S.A. Zenios, Robust Optimization of Large Scale Systems, *Operations Research* 43 (1995) 264–281.

[20] H. Yaman, O.E. Karaşan, M.C. Pinar, The robust spanning tree problem with interval data, *Operations Research Letters* 29 (2001) 31–40.