Robust Minimum Weight Perfect Matchings

Monja Raschke
monja.raschke@rwth-aachen.de
Teaching and Research Area
Combinatorial Optimization
RWTH Aachen, Germany

Felix Engelhardt
engelhardt@combi.rwth-aachen.de
Teaching and Research Area
Combinatorial Optimization
RWTH Aachen, Germany

Christina Büsing
buesing@combi.rwth-aachen.de
Teaching and Research Area
Combinatorial Optimization
RWTH Aachen, Germany

ABSTRACT

In this work, we investigate the theoretical complexity of the robust minimum weight perfect matching problem. We consider discrete, interval-based and budgeted uncertainty sets under min-max and min-max regret criteria, as well as two-stage robustness. We identify polynomial-time solvable special cases for paths, trees, complete, and series-parallel graphs. We also show that complexity bounds are tight for most problems, except for min-max regret and two-stage robustness with Γ as part of the input.

KEYWORDS

robust optimization, combinatorial optimization, complexity theory, uncertainty budget, reductions, dynamic programming

1 INTRODUCTION

Given a graph with costs for each edge, the *minimum weight perfect matching problem (MWPM)* consists of finding a subset of edges such that every vertex of the graph is incident to exactly one edge and the total cost of selected edges is minimal. In many applications, costs are not known in advance. One approach to anticipate this is *robust combinatorial optimisation (RCO)* [9, 13]. In this work, we investigate the theoretical complexity of the robust MWPM.

Uncertainty Sets and Decision Criteria. In RCO, different types of cost values are described via uncertainty sets $\mathcal{U}$. We consider three types: *discrete uncertainty (D)*, i.e. scenarios are explicitly listed; *interval-based uncertainty (I)*, i.e. upper and lower bounds for costs; and *budgeted uncertainty (B)* cf. [4], i.e. interval-based uncertainty where only a limited number of costs may deviate. Furthermore, we consider three robust decision criteria: *min-max (MiMa)* minimises the costs of a solution in the worst case, *min-max regret (MiMaR)* minimises the difference of costs of the chosen solution to the best solution of any scenario, and the *two-stage (TS)* allows selecting a partial solution at fixed costs, which has to be completed after the remaining cost are known. The combination of those cases defines nine robust optimisation problems. We denote those by subscript (D, I, B) and superscript (MiMa, MiMaR, TS), e.g. the min-max regret weighted perfect matching problem with discrete uncertainty is abbreviated as $MWPM_D^{MiMaR}$. We add k to denote the decision version of a problem, e.g. $MWPM_D^{MiMaR}$ becomes k-$MWPM_D^{MiMaR}$.

Related Work. In general, the MWPM is equivalent to finding a matching with maximum total weight [12, Ch. 11]; both problems can be solved in polynomial time [6]. For the robust version of the MWPM, the classic book about RCO by Kouvelis and Yu [13] as well as the more recent book by Goerigk and Hartisch [9] only dedicate a brief section to *robust assignment problems (RAPs)*, a special case of the robust MWPM on a bipartite graph. Goerigk and Hartisch [9, Ch. 4] proved that MiMa problems with interval uncertainty preserve the complexity of the nominal problem. Since the MWPM is solvable in polynomial time, these results mean that the *min-max weighted perfect matching problem with interval-based uncertainty ($MWPM_I^{MiMa}$)* is solvable in polynomial time as well. TS problems with interval uncertainty also preserve the complexity of the nominal problem [11], as do combinatorial optimisation problems with MiMa budgeted uncertainty [3]. Thus, the *two-stage minimum weighted perfect matching problem with interval-based uncertainty ($MWPM_I^{TS}$)* and the *min-max weighted perfect matching problem with budgeted uncertainty ($MWPM_B^{MiMa}$)* are also polynomial-time solvable.

Apart from these general results, the theoretical complexity of the robust MWPM is not yet sufficiently studied. The research on robust perfect matching (PM) problems mostly considers RAPs. Since the RAP is a special case of the MWPM, all **NP**-hardness results carry over: Aissi et al. [1] proved that the MiMa and MiMaR assignment problem with discrete uncertainty and the MiMaR assignment problem with interval uncertainty are **NP**-hard even on directed acyclic graphs. Goerigk and Hartisch [9, Ch. 9] proved that the TS assignment problem with discrete budgeted uncertainty is **NP**-hard. Kasperski and Zieliński [11] proved the **NP**-hardness of the *two-stage minimum weighted perfect matching problem with discrete uncertainty ($MWPM_D^{TS}$)* for bipartite graphs. We summarise complexity results on the robust MWPM in Table 1:

Table 1: Known results on the complexity of the MWPM with different uncertainty sets and robust decision criteria.

uncertainty	min-max	min-max regret	two-stage
interval	polynomial [9]	**NP**-hard [1]	polynomial [11]
budgeted	polynomial [3]	**NP**-hard	**NP**-hard [9]
discrete	**NP**-hard [1]	**NP**-hard [1]	**NP**-hard [11]

Note that since the *min-max regret weighted perfect matching problem with budgeted uncertainty ($MWPM_B^{MiMaR}$)* contains the min-max regret weighted perfect matching problem with interval-based uncertainty ($MWPM_I^{MiMaR}$) as a special case, it is **NP**-hard.

Polynomial Time Solvable Special Cases. The $MWPM_I^{MiMaR}$ is the only one of the considered **NP**-hard robust PM problems, for which polynomially solvable special cases have been identified: Averbakh et al. [2] proved that the MiMaR versions with interval uncertainty of polynomially solvable minisum network problems are polynomially solvable, if the number of non-degenerate intervals is fixed or bounded by the logarithm of a polynomial of m, the number of edges. Escoffier et al. [7, 8] proved that the $MWPM_I^{MiMaR}$ is solvable in pseudopolynomial time on bipartite graphs with bounded treewidth and bounded maximum degree, and solvable in polynomial time on bipartite graphs when the worst-case regret is bounded.

INOC '26, 22–24 April 2026, Liège (Belgium)

10.48786/inoc.2026.13

Summary of Contributions. For the the robust PM problems for which **NP**-hardness results exist, i.e. the *two-stage minimum weighted perfect matching problem with budgeted uncertainty ($MWPM_B^{TS}$)*, the $MWPM_B^{MiMaR}$ and the $MWPM_D^{MiMaR}$, we prove that their decision version is either in **NP** or the next higher complexity class $\Sigma_2^{\mathbf{P}}$. We furthermore examine the complexity of these problems when restricted to paths, trees, series-parallel graphs (SPGs), and complete graphs and provide dynamic programming algorithms based on SP-trees that solve the $MWPM_I^{MiMaR}$, the $MWPM_B^{MiMaR}$, and the $MWPM_B^{TS}$ on SPGs in polynomial time.

2 COMPLEXITY ANALYSIS

2.1 Preliminaries

We assume all graphs to be undirected and simple. For any $n \in \mathbb{N}$, we use $[n]$ as a shorthand for $\{1, ..., n\}$ and $[n]_0$ for $[n] \cup \{0\}$.

Definition 1. Given a graph $G = (V, E)$ with edge weights $c_e \in \mathbb{R}$ for all $e \in E$ the *minimum weight perfect matching problem (MWPM)* consists of finding a PM $M \subseteq E$ in G such that the total weight $\sum_{e \in M} c_e$ of M is minimal or decide that no PM exists in G.

We denote uncertainty sets with $\mathcal{U}$ and consider the following three types:

Definition 2 ([9, Ch. 3])**.** An *interval-based uncertainty set* is given by

$$\mathcal{U} = [\underline{c}_1, \overline{c}_1] \times [\underline{c}_2, \overline{c}_2] \times ... \times [\underline{c}_n, \overline{c}_n],$$

where $\underline{c}_i \in \mathbb{R}_{\geq 0}$ is the lower and $\overline{c}_i \in \mathbb{R}_{\geq 0}$ the upper bound for the costs of the i-th element.

Definition 3 ([9, Ch. 3])**.** A *budgeted uncertainty set* is given by

$$\mathcal{U} = \left\{ c \in \mathbb{R}^n_{\geq 0} \;\middle|\; c_i = \underline{c}_i + \delta_i \hat{c}_i \quad \forall i \in, \delta_i \in \{0,1\}, \sum_{i \in [n]} \delta_i \leq \Gamma \right\},$$

where $\underline{c}_i$ is the lower bound and $\underline{c}_i + \hat{c}_i$ the upper bound of the costs of the i-th element. The parameters $\hat{c}_i$ are the *deviations* and Γ is called the *uncertainty budget.*

Definition 4 ([9, Ch. 3])**.** A *discrete uncertainty set* consists of $K \in \mathbb{N}$ cost scenarios and has the form

$$\mathcal{U} = \left\{ c^1, ..., c^K \right\} \subsetneq \mathbb{R}^n_{\geq 0},$$

where c^j is the j-th cost scenario and c_i^j denotes the costs of the i-th element in the j-th scenario.

Furthermore, recall the following three robust decision criteria, where $\mathcal{X}$ denotes the set of feasible solutions:

Definition 5 ([9, Ch. 3])**.** Given an uncertainty set $\mathcal{U}$, the min-max problem is to solve

$$\min_{x \in \mathcal{X}} \max_{c \in \mathcal{U}} \sum_{i \in [n]} c_i x_i.$$

Definition 6 ([9, Ch. 3])**.** Given an uncertainty set $\mathcal{U}$, the min-max regret problem is to solve

$$\min_{x \in \mathcal{X}} \max_{c \in \mathcal{U}} \left(\sum_{i \in [n]} c_i x_i - \min_{y \in \mathcal{X}} \sum_{i \in [n]} c_i y_i \right).$$

We refer to the optimal solution of the worst-case scenario, i.e. the solution our solution is compared to, as the *adversarial solution.*

Definition 7 ([9, Ch. 3])**.** Given an uncertainty set $\mathcal{U}$ and first stage costs C, the two-stage problem is to solve

$$\min_{x \in \mathcal{X}'} \max_{c \in \mathcal{U}} \min_{y \in \mathcal{X}(x)} \left(\sum_{i \in [n]} C_i x_i + \sum_{i \in [n]} c_i x_i \right),$$

where $\mathcal{X}'$ is the set of partial solutions, that can be completed to a feasible solution, and $\mathcal{X}(x)$ is the set of feasible solutions that the partial solution $x \in \mathcal{X}'$ can be completed to.

W.l.o.g., we restrict proofs to non-negative costs, because we can raise the costs of all edges in every scenario by $F = \left|\min_{c \in \mathcal{U}, e \in E}\{c_e\}\right|$ without changing the structure of solutions. We also restrict all considerations to minimisation problems, since maximisation problems of matchings are equivalent to the MWPM with negative costs.

2.2 NP–Completeness Results

Recall, that the decision version of a problem is to decide whether a given instance has a solution with an objective value not worse than some specified number k. Clearly, the decision versions of the min-max weighted perfect matching problem with discrete uncertainty ($MWPM_D^{MiMa}$), $MWPM_D^{TS}$, $MWPM_I^{MiMaR}$ and $MWPM_D^{MiMaR}$ are **NP**-complete. For a given Γ, there are $\binom{\Gamma}{|E|}$ scenarios. For constant Γ, this number of scenarios is polynomial in $|E|$. Thus, we state without proof:

Theorem 1. *k-$MWPM_B^{MiMaR}$ and k-$MWPM_B^{TS}$ with constant* Γ *are in* **NP**.

However, if Γ is part of the input, the enumeration above is no longer polynomial in Γ. At most, this raises the complexity by one level in polynomial hierarchy, i.e. to $\Sigma_2^{\mathbf{P}}$ [15]:

Theorem 2. *k-$MWPM_B^{MiMaR}$ with* Γ *as part of the input is in* $\Sigma_2^{\mathbf{P}}$.

Proof. It suffices to show that k-$MWPM_B^{MiMaR}$ can be expressed in the form of a standard $\Sigma_2^{\mathbf{P}}$-problem [15], i.e.

$$\exists M \in \mathcal{M}\, \forall c \in \mathcal{U} : \sum_{e \in M} c_e - \sum_{e \in M_c} c_e \leq k,$$

where $\mathcal{M}$ is the set of PMs in G and M_c denotes a minimum weight PM in G for costs c. Since M_c can be computed in polynomial time, $\sum_{e \in M} c_e - \sum_{e \in M_c} c_e \leq k$ can be evaluated in polynomial time for any PM M. Thus, k-$MWPM_B^{MiMaR}$ is in $\Sigma_2^{\mathbf{P}}$. □

Theorem 3. *k-$MWPM_B^{TS}$ with* Γ *as part of the input is in* $\Sigma_2^{\mathbf{P}}$.

Proof. Let $\mathcal{U}$ be the budgeted uncertainty set defined by lower bounds $\underline{c}$ and maximum deviations $\hat{c}$ for each edge, and Γ. It suffices to show that k-$MWPM_B^{TS}$ can be expressed in the form of a standard $\Sigma_2^{\mathbf{P}}$-problem, i.e.

$$\exists M_F \in \mathcal{M} \forall c \in \mathcal{U} : \sum_{e \in M_F} C_e + \min \left\{ \sum_{e \in M_S} c_e \;\middle|\; M_S \in \mathcal{S}(M_F) \right\} \leq k,$$

where for any matching M_F the set $\mathcal{S}(M_F)$ is the set of all matchings that complete M_F to a PM in G, and $\mathcal{M}$ is the set of all matchings M_F in G, for which $\mathcal{S}(M_F)$ is not empty. Whether the costs of completing a given matching M_F in a scenario c exceeds k can be checked in polynomial time. □

The complexity classes of the robust MWPMs considered in this work are summarised in Table 2. Note that it is open whether the $MWPM_B^{TS}$ and the $MWPM_B^{MiMaR}$ are $\Sigma_2^{\mathbf{P}}$-hard.

Table 2: Complexity classes. Results that leave a gap to the strongest known hardness result are marked grey.

uncertainty	min-max	min-max regret	two-stage
interval	**P**	**NP**	**P**
budgeted (incl. Γ)	**P**	Σ_2^p	Σ_2^p
budgeted (const. Γ)	**P**	**NP**	**NP**
discrete	**NP**	**NP**	**NP**

2.3 Paths, Trees and Complete Graphs

Robust MWPMs can be solved in polynomial time if enumerating their entire solution space and determining an objective value, i.e. solving the adversarial problem, of any solution is possible in polynomial time. This is the case for $MWPM_D^{MiMa}$ and $MWPM_D^{MiMaR}$ on trees and cycles with at most 1 and 2 PMs, respectively.

In contrast, the $MWPM_D^{TS}$ is **NP**-hard, even if the underlying graph is a path. We prove this using a reduction from the two-stage selection problem with discrete uncertainty (SP_D^{TS}):

Theorem 4. *The $MWPM_D^{TS}$ is weakly **NP**-hard on paths, if the number of scenarios is constant, even for $K = 2$ scenarios, and strongly **NP**-hard if K is part of the input.*

Proof. Given n numbers $c_1, ..., c_n \in \mathbb{N}$ and a budget $p \in \mathbb{N}$ the *selection problem (SP)* consists of selecting p of these numbers $\{c_{i_1}, ..., c_{i_p}\} \subseteq \{c_1, ..., c_n\}$, such that their sum $\sum_{j\in[p]} c_{i_j}$ is minimal.

Consider the SP_D^{TS} where the budget p of items to be selected equals the total number n of available items. This special case of the robust selection problem is known to be weakly **NP**-hard when the number K of scenarios is fixed, even for $K = 2$ scenarios, and strongly **NP**-hard, when K is part of the input [10].

Let $\mathcal{I}$ be an SP_D^{TS} instance with n items $\{1, ..., n\}$, first stage costs C_i for all items $i \in [n]$, K second stage cost scenarios $\mathcal{U} = \{c^1, ..., c^K\}$, and $p = n$. We construct an instance $\mathcal{I}'$ of the $MWPM_D^{TS}$ from $\mathcal{I}$: the graph is the path with $2n$ vertices $G = (V, E)$, where $V = [2n]$ and $E = \{\{i, i+1\} \mid i \in [2n-1]\}$. Figure 1 shows the construction of the graph.

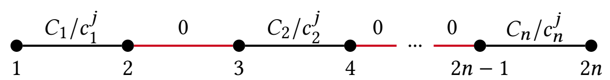

Figure 1: Reduction graph. Black edges correspond to items. Edges with one cost have that cost in all scenarios & stages.

The edges $E_M = \{\{2i-1, 2i\} \mid i \in [n]\}$ form a PM in G. Since G is a path, E_M is the only PM. In order to define the costs, we split E into E_M and the remaining edges $E_0 = \{\{2i, 2i+1\} \mid i \in [n-1]\}$ (marked red in Fig. 1). For any $e \in E$ the first stage costs are

$$C'_e = \begin{cases} 0, & \text{if } e \in E_0 \\ C_i, & \text{if } e \in E_M, e = \{2i-1, 2i\} \end{cases},$$

and there are K second stage cost scenarios, i.e.

$$c'^j_e = \begin{cases} 0, & \text{if } e \in E_0 \\ c_i^j, & \text{if } e \in E_M, e = \{2i-1, 2i\}, \end{cases}$$

where c_i^j is the second stage cost of item i in scenario $j \in [K]$.

Any solution of $\mathcal{I}'$ can be transformed into a solution of $\mathcal{I}$, since using the edge $\{2i-1, 2i\}$ corresponds to picking item i. If an edge is part of the first stage solution, the corresponding item has to be selected in the first stage, otherwise in the second stage. Due to the structure of G any PM has to contain exactly the edges E_M which corresponds to selecting all n items.

Any solution of $\mathcal{I}$ can be transformed into a solution of $\mathcal{I}'$ analogously. Since first and second stage costs in $\mathcal{I}$ and $\mathcal{I}'$ are equal, the reduction preserves the objective value of the solutions and an optimal solution to $\mathcal{I}$ corresponds to an optimal solution to $\mathcal{I}'$. This reduction is computable in polynomial time and preserves the number of scenarios K. This proves the theorem. □

The proof of Theorem 4 can easily be extended to cycles.

Corollary 1. *The $MWPM_D^{TS}$ on cycles is weakly **NP**-hard, if the number of scenarios is constant, even for $K = 2$ scenarios, and strongly **NP**-hard, if K is part of the input.*

Finally, for all considered robust MWPM problems the complexity remains the same when they are restricted to complete graphs.

Theorem 5. *For $CR \in \{MiMa, MiMaR, TS\}$ and $US \in \{I, B, D\}$ the problem $MWPM_{US}^{CR}$ on complete graphs is **NP**-hard, if the $MWPM_{US}^{CR}$ is **NP**-hard on general graphs.*

Proof. Given an instance $\mathcal{I}$ of the $MWPM_{US}^{CR}$ on a graph $G = (V, E)$ we construct an instance $\mathcal{I}'$ of the $MWPM_{US}^{CR}$ on a complete graph $G' = (V, E \cup \overline{E})$, where $\overline{E} = \{\{v, w\} \mid v, w \in V, \{v, w\} \notin E\}$ contains the missing edges of G. We assign each edge in $\overline{E}$ costs of $2Q$ in all scenarios of the respective uncertainty set, where $Q \geq \sum_{e\in E} \max_{c\in\mathcal{U}} c_e + 1$ (for TS problems $Q \geq \sum_{e\in E} \max\{\max_{c\in\mathcal{U}} c_e, C_e\} + 1$) is a sufficiently large number. All edges in E have the same costs as specified in instance $\mathcal{I}$. Note that the optimal solutions of the instances $\mathcal{I}$ and $\mathcal{I}'$ are identical, unless $\mathcal{I}$ has no solution, in which case $\mathcal{I}'$ will contain at least one edge from $\overline{E}$. Obviously, the instance $\mathcal{I}'$ can be constructed in polynomial time. Thus, if the problem $MWPM_{US}^{CR}$ is **NP**-hard in general, it is still **NP**-hard when restricted to complete graphs. □

2.4 Series-Parallel Graphs

For interval and budgeted uncertainty, we show that the robust problems are solvable in polynomial time on SPGs.

Definition 8 ([5, Ch. 11]). A graph $G = (V, E)$ with a *source* $s \in V$ and a *target* $t \in V$ is called *series-parallel* if either:

(1) G consists of the single edge $\{s, t\}$
(2) G can be constructed from two SPG G_1 with source s_1 and target t_1 and G_2 with source s_2 and target t_2 using one of the following two operations:
  (a) *Series composition*: identify t_1 with s_2.
  (b) *Parallel composition*: identify s_1 with s_2 and t_1 with t_2.

We refer to s, t as the *terminal vertices* of the SPG.

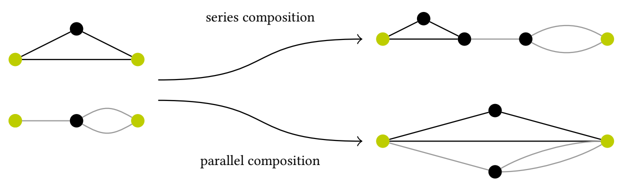

Figure 2: Construction of SPGs. Terminal vertices are green.

The construction of SPGs as described in Def. 8 is illustrated in Fig. 2. SPGs may contain parallel edges, and can be represented by so-called *SP-trees*. These are binary decomposition trees, that show how an SPG is built from its edges using series and parallel compositions. Fig. 3 shows an example of an SPG and its SP-tree.

Definition 9 ([5, Ch. 11])**.** The *SP-tree* of an SPG $G = (V, E)$ is a binary tree. We refer to the tree's vertices as *nodes*. Each node i is labelled with (v, w), with $v, w \in V$, and corresponds to a series-parallel subgraph G_i of G with source v and target w. There are three types of nodes:

(1) *leaves*: the leaves of the SP-tree represent the edges of G.
(2) *s-nodes*: represent the series composition of their children (by convention, the left child is considered first).
(3) *p-nodes*: represent the parallel composition of their children.

The root of the tree is labelled with (s, t) and corresponds to G.

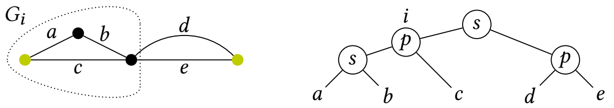

Figure 3: An SPG and its SP-tree. Node labels have been omitted. The dashed line marks the subgraph associated with node i of the SP-tree.

The SP-tree of an SPG can be constructed in $O(m)$ time and has $O(m)$ nodes, where m is the number of edges of the SPG [14].

Next, we present a dynamic programming approach, which assembles an optimal solution for the $MWPM_B^{MiMaR}$ from optimal partial solutions in the subgraphs associated with the nodes of the SP-tree. Since the MiMaR criterion cannot be applied to partial solutions, we use the following modified objective.

Definition 10. Let $G = (V, E)$ be a graph with lower bounds $\underline{c}_e$, edge cost deviations $\hat{c}_e$, and an uncertainty budget Γ. Let $G' = (V', E')$ be a subgraph of G. The *worst-case partial regret* of a matching $M \subseteq E'$ with respect to a set $V_A \subseteq V'$ and a *partial budget* $\gamma \in [\Gamma]$ is then we define $PRB_{G'}(M, V_A, \gamma) =$

$$\max_{M', M_\gamma} \left\{ \sum_{e \in M} c_e(M_\gamma) - \sum_{e \in M'} c_e(M_\gamma) \;\middle|\; \begin{array}{l} M' \text{ PM in } G'(V_A) \\ M_\gamma \subseteq M \\ |M_\gamma| \leq \gamma \end{array} \right\},$$

where the costs c are defined by the set M_γ as

$$c_e(M_\gamma) = \begin{cases} \underline{c}_e + \hat{c}_e, & \text{if } e \in M_\gamma \\ \underline{c}_e, & \text{else} \end{cases}.$$

Thus, $PRB_{G'}(M, V_A, \gamma)$ is the largest possible difference between the costs of M and those of a matching M' in G' covering exactly the vertices V_A, while at most γ edges in G' have deviating costs.

Note, that for $G' = G$, $V_A = V$, $\gamma = \Gamma$, and a PM M in G the definition of $PRB_{G'}(M, V_A, \gamma)$ is equivalent to the worst-case regret of M in defined by the $MWPM_B^{MiMaR}$.

Theorem 6. *$MWPM_B^{MiMaR}$ is solvable in polynomial time on SPGs.*

Proof. Any matching in an SPG with at least two edges decomposes into matchings in the subgraphs the SPG was composed from. Since SPGs are recursive, this decomposition can be extended until the considered subgraphs are the edges of the original graph, for which the worst-case partial regret is trivial to compute.

Note, that if a non-terminal vertex of any subgraph is exposed by the respective part of the matching, it is also exposed by the complete matching, since the composition of SPGs never adds edges to non-terminal vertices. Thus, any perfect matching decomposes into matchings which cover all non-terminal vertices of their respective subgraphs and for each component the set of covered terminal vertices suffices to define the set of all covered vertices.

Consider a matching M in an SPG G with $PRB_G(M, V_A, \gamma)$ minimal among all matchings in G covering the same set of vertices as M, for some set of vertices V_A and budget γ. The worst-case regret of M is defined by the adversarial solution M_A and the set M_γ of edges with deviating costs. For each of the matchings M_{comp}, that M decomposes into, $PRB_{G'}(M_{comp}, V'_A, \gamma')$ is minimal among all matchings in the respective subgraph G' covering the same set of vertices as M_{comp}, where γ' is the number of edges in M_γ in G' and V'_A are the vertices covered by the edges of M_A that are in G', as a simple exchange argument proves.

Our algorithm reverses the decomposition process. For each node i of the SP-tree we compute a table T_i with entries $T_i(S, S', \gamma)$ for $S, S' \subseteq \{s, t\}$, where $\{s, t\}$ are the terminal vertices of the subgraph G_i associated with i, and $\gamma \in [\Gamma]_0$. We use $V_i^S, V_i^{S'}$ to denote the union of all non-terminal vertices of G_i and S or S', respectively. The entries are defined as

$$T_i\left(S, S', \gamma\right) = \min\left\{ PRB_{G_i}\left(M, V_i^{S'}, \gamma\right) \middle| M \text{ is a PM in } G_i\left(V_i^S\right)\right\},$$

where the sets S, S' denote the terminal vertices covered by the solution and adversarial solution component, respectively, and γ is the partial budget. The tables are necessary, because in advance we only know that each component of the optimal solution has minimum worst-case partial regret in its respective subgraph, but not w.r.t. which vertex set and partial budget or which vertices the component covers. Thus, we compute a table with all possible configurations. The tables have linear size, as only subsets of the terminal vertices have to be enumerated.

The nodes are processed bottom-up, so the tables for the children can be used to compute the table for the parent. For each of the $O(m)$ nodes of the SP-tree, the corresponding table has $O(\Gamma)$ entries. We show for leaves, s-nodes, and p-nodes that each entry can be computed in polynomial time. We use $\perp$ to denote that for a given set of parameters no solution exists.

leaf-nodes: A leaf node i of the SP-tree represents an edge e connecting the terminal vertices s and t of the associated subgraph G_i. Compute the entries of T_i as:

$$T_i(S, S', \gamma) = \begin{cases} 0, & \text{if } S = S' = \emptyset \text{ or } S = S' = \{s, t\} \\ \underline{c}_e, & \text{if } S = \{s, t\}, S' = \emptyset, \gamma = 0 \\ \underline{c}_e + \hat{c}_e, & \text{if } S = \{s, t\}, S' = \emptyset, \gamma > 1 \\ -\underline{c}_e, & \text{if } S = \emptyset, S' = \{s, t\} \\ \perp, & \text{else} \end{cases}.$$

Since G_i consists only of the edge e, any matching in G_i either consists of that edge and hence covers both terminal vertices or is empty and covers no vertices. The cases covered by the above equation follow: if solution and adversarial solution cover the same vertices, they also have the same costs, resulting in a regret of 0. Otherwise, the regret depends on the costs of e, which will only deviate, when e is part of the solution and they are allowed to, i.e. $\gamma > 0$. Clearly, each entry can be computed in $O(1)$ time.

p-nodes: The subgraph G_i associated with a p-node i with children j and k results from the parallel composition of the subgraphs G_j and G_k. We refer to the terminal vertices of G_i, G_j, G_k as s, t (see Fig. 4). The entries $T_i(S, S', \gamma)$ for a p-node are computed as the optimal objective value of the following bi-level mathematical program (BMP):

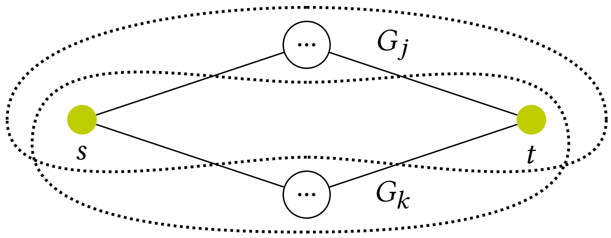

Figure 4: Sketch of the subgraph G_i associated with a p-node i and the subgraphs G_j and G_k it is composed of. The terminal vertices (green) are the only shared vertices of G_j and G_k and can thus be covered by matchings in either of them.

$$\begin{aligned}
\min \quad & T_j(S_j, S'_j, \gamma_j) + T_k(S_k, S'_k, \gamma_k) \\
s.t. \quad & S_j \cap S_k = \emptyset, S_j \cup S_k = S, S_j, S_k \subseteq S \\
& S'_j, S'_k, \gamma_j, \gamma_k \in \arg\max \quad T_j(S_j, S'_j, \gamma_j) + T_k(S_k, S'_k, \gamma_k) \\
& \qquad s.t. \quad S'_j \cap S'_k = \emptyset, S'_j \cup S'_k = S', S'_j, \\
& \qquad\qquad S'_k \subseteq S', \gamma_j + \gamma_k = \gamma, \gamma_j, \gamma_k \in [\gamma]_0
\end{aligned}$$

where the variables S_j, S_k denote the set of terminal vertices covered by the solution component in G_j and G_k, respectively. Analogously, S'_j, S'_k denote the set of terminal vertices covered by the solution component in G_j and G_k, respectively, and γ_j, γ_k describe how the partial budget γ is distributed between G_j and G_k. The bi-level structure captures that the adversarial solution and the cost scenario including the distribution of the budget are selected after the solution is known. The BMP above can be solved in $O(\Gamma)$ via enumeration, since there are only 4 subsets of terminal vertices and $\gamma + 1$ possible distributions of γ.

s-nodes: The subgraph G_i associated with an s-node i with children j and k results from the series composition of the subgraphs G_j and G_k. Note, that G_i, G_j, G_k have different terminal vertices. We denote the terminal vertices of G_i with s, t, and the vertex at which G_j and G_k are connected with l (see Fig. 5).
The entry $T_i(S, S', \gamma)$ is computed as the optimal objective value of the following BMP:

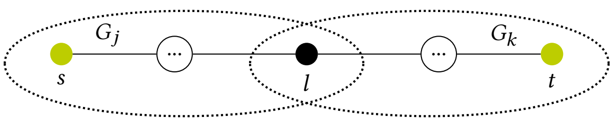

Figure 5: Sketch of the subgraph G_i associated with an s-node i and the subgraphs G_j and G_k it is composed of. Each of the terminal vertices (green) only belongs to one of the subgraphs. The only shared vertex of G_j and G_k is their former terminal vertex l.

$$\begin{aligned}
\min \quad & T_j(S_j \cup L_j, S'_j \cup L'_j, \gamma_j) + T_k(S_k \cup L_k, S'_k \cup L'_k, \gamma_k) \\
s.t. \quad & L_j \cap L_k = \emptyset, L_j \cup L_k = \{l\}, L_j, L_k \subseteq \{l\} \\
& L'_j, L'_k, \gamma_j, \gamma_k \in \arg\max \quad \textstyle\sum_{X \in \{j,k\}} T_X(S_X \cup L_X, S'_X \cup L'_X, \gamma_X) \\
& \qquad s.t. \quad L'_j \cap L'_k = \emptyset, L'_j \cup L'_k = \{l\}, L'_j, \\
& \qquad\qquad L'_k \subseteq \{l\}, \gamma_j + \gamma_k = \gamma, \gamma_j, \gamma_k \in [\gamma]_0
\end{aligned}$$

where the variables L_j, L_k express whether the vertex l is covered by the solution component in G_j or G_k. Analogously, L'_j, L'_k express whether l is covered by the adversarial solution component in G_j or G_k. Note, that l has to be covered by solution and adversarial solution, since in G_i it is no longer a terminal vertex. The variables γ_j, γ_k and the sets S_j, S_k, S'_j, S'_k have the same meaning as for p-nodes. However, S_j, S_k, S'_j, S'_k are no longer variables, since their values are predefined by the structure of G_i and the sets S, S' (recall Fig. 5): S_j and S_k are given by the intersection of S with $\{s\}, \{t\}$, and S'_j and S'_k by the intersection of S' with $\{s\}, \{t\}$, respectively. Again, the BMP above can be solved in $O(\Gamma)$ time, by enumerating all $O(\Gamma)$ feasible solutions.
Since all tables can be computed in polynomial time, the entire algorithm has a polynomial runtime. □

Since Theorem 6 also holds for $\Gamma = |E|$, it follows that:

Corollary 2. *$MWPM_I^{MiMaR}$ is solvable in polynomial time on SPGs.*

Similarly to the $MWPM_B^{MiMaR}$, we design a dynamic program for $MWPM_B^{TS}$ on SPGs. For that, define an alternative objective function to evaluate partial solutions:

Definition 11. Let $G = (V, E)$ be a graph with first-stage costs C_e and lower bounds $\underline{c}_e$, deviations $\hat{c}_e$, and uncertainty budget Γ for the second-stage costs. Let $G' = (V', E')$ be a subgraph of G. The *worst-case partial completion cost* of the matching M_F in G' w.r.t. the set $V_S \subseteq V'$ and the partial budget $\gamma \in [\Gamma]_0$ is given by the optimal objective value of $WPC_{G'}(M_F, V_S, \gamma)$:

$$\begin{aligned}
\max \quad & \sum_{e \in M_F} C_e + \sum_{e \in M_S} c_e \\
s.t. \quad & |E_\gamma| \leq \gamma, E_\gamma \subseteq E' \\
& M_S \in \arg\min \left\{ \sum_{e \in M_F} C_e + \sum_{e \in M_S} c_e \,\middle|\, M_F \cup M_S \text{ PM in } G'(V_S) \right\}
\end{aligned}$$

where c is the cost scenario in which the edge costs in E_γ deviate. Thus, $WPC_{G'}(M_F, V_S, \gamma)$ is the cost of a PM M in $G'(V_S)$, where $M_F \subseteq M$ has certain first-stage costs and $M \setminus M_F$ is cost minimal for the worst possible cost scenario with at most γ deviations.

Solving the problem above for $G' = G$, $V_S = V$, and $\gamma = \Gamma$ is equivalent to solving the adversarial problem for M. Using this, we can prove that the $MWPM_B^{TS}$ is solvable in polynomial time on SPGs, analogously to the proof for $MWPM_B^{MiMaR}$. Thus, we state:

Theorem 7. *The $MWPM_B^{TS}$ is solvable in polynomial time on SPGs.*

Building and iteratively expanding the worst-case scenario for a partial solution is impossible for discrete uncertainty sets. Instead, we prove that the $MWPM_D^{MiMa}$ is **NP**-hard on SPG:

Theorem 8. *The $MWPM_D^{MiMa}$ on simple SPGs is weakly* **NP***-hard, if the number of scenarios is constant, even for $K = 2$ scenarios, and strongly* **NP***-hard, if K is part of the input.*

Proof. Consider an instance $\mathcal{I}$ of the SP_D^{TS}, with a set $I = [n]$ of items, $p = n$ of which have to be selected, first stage costs C_i for each item i, and K second stage cost scenarios $\mathcal{U} = \{c^1, ..., c^K\} \subsetneq \mathbb{R}^n$. We construct an instance $\mathcal{I}'$ of the $MWPM_D^{MiMa}$ from $\mathcal{I}$ as follows: First, we construct the graph $G = (V, E)$, where $V = [4n]$ and the edge set $E = E_I \cup E_0$ is composed of $E_I = \bigcup_{i=1}^n \{\{4i, 4i-1\}, \{4i-1, 4i-2\}, \{4i-2, 4i-3\}, \{4i-3, 4i\}\}$, which contains a cycle for each item of $\mathcal{I}$, and $E_0 = \{\{4i, 4i{+}2\} \mid i \in [n{-}1]\}$, which connects the cycles from E_I. The construction of G is illustrated in Fig. 6. Edges in E_0 are marked red.

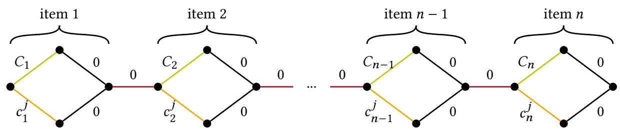

Figure 6: Graph for the reduction from the SP_D^{TS} to the $MWPM_D^{MiMa}$. Each cycle represents an item from the SP_D^{TS} instance. The green and orange edges correspond to selecting an item in the first or second stage, respectively.

Clearly, G is series-parallel and simple. The uncertainty set $\mathcal{U}'$ consists of K cost scenarios, which are defined as follows:

$$c_e'^j = \begin{cases} C_i, & \text{if } e = \{4i-1, 4i-2\} \\ c_i^j, & \text{if } e = \{4i-2, 4i-3\} \\ 0, & \text{else} \end{cases} \quad \forall e \in E, j \in [K].$$

Analogously to the proof of Thm. 4, any solution X for $\mathcal{I}$ can be transformed into a solution M for $\mathcal{I}'$ and vice versa, such that X and M have the same objective value. Here, the edge $\{4i-1, 4i-2\}$ corresponds to selecting item i in the first stage, while the edge $\{4i-2, 4i-3\}$ corresponds to selecting item i in the second stage. It follows that a solution M for $\mathcal{I}'$ is optimal if and only if the corresponding solution X is optimal for $\mathcal{I}$. This reduction is computable in polynomial time and preserves the number of scenarios. □

The $MWPM_D^{MiMaR}$ is **NP**-hard on SPGs as well. We omit the proof of the following theorem, as it is very similar to the proof of Thm. 8.

Theorem 9. *The $MWPM_D^{MiMaR}$ on simple SPG is weakly* **NP**-*hard, if the number of scenarios is constant, even for $K = 2$ scenarios, and strongly* **NP**-*hard, if K is part of the input.*

3 CONCLUSION

The decision versions of the **NP**-hard MWPM problems are in **NP**, except for the $MWPM_B^{MiMaR}$ and the $MWPM_B^{TS}$. For both, it remains open whether we can solve the adversarial problem in polynomial time, but we do know that k-$MWPM_B^{MiMaR}$ and k-$MWPM_B^{TS}$ are at most in $\Sigma_2^{\mathbf{P}}$. If the uncertainty budget Γ is not part of the input, both problems' decision versions are in **NP**, because all worst-case scenarios for a solution can be enumerated and evaluated in polynomial time. Second, we analysed the complexity of the **NP**-hard problem variations when restricted to certain graph classes. The results of this analysis are summarised in Table 3.

The complexity of all considered problems remains unchanged when they are restricted to complete graphs, negative edge costs are included or the maximisation versions are considered.

Third, we presented algorithms to solve the $MWPM_I^{MiMaR}$, the $MWPM_B^{MiMaR}$, and the $MWPM_B^{TS}$ on SPGs in polynomial time by dynamic programming on the SP-tree. In comparison to that, the $MWPM_D^{MiMa}$, the $MWPM_D^{MiMaR}$, and the $MWPM_D^{TS}$ are only solvable in polynomial time on graphs that allow the enumeration of all feasible solutions in polynomial time. Interestingly, Aissi et al. [1] found three of the problems we consider, i.e. the $MWPM_I^{MiMaR}$, the $MWPM_D^{MiMaR}$ and the $MWPM_D^{MiMa}$, to be hard when restricted to bipartite directed acyclic graphs. Two of those are strongly **NP**-hard on SPGs, one, $MWPM_I^{MiMaR}$, is solvable in polynomial time.

REFERENCES

[1] Hassene Aissi, Cristina Bazgan, and Daniel Vanderpooten. 2005. Complexity of the min–max and min–max regret assignment problems. *Oper. Res. Letters* 33, 6 (2005), 634–640. doi:10.1016/j.orl.2004.12.002

[2] Igor Averbakh and Vasilij Lebedev. 2004. Interval data minmax regret network optimization problems. *Discret. Appl. Math.* 138, 3 (2004), 289–301. doi:10.1016/S0166-218X(03)00462-1

[3] Dimitris Bertsimas and Melvyn Sim. 2003. Robust discrete optimization and network flows. *Math. Prog.* 98, 1 (2003), 49–71. doi:10.1007/s10107-003-0396-4

[4] Dimitris Bertsimas and Melvyn Sim. 2004. The Price of Robustness. *Oper. Res.* 52, 1 (2004), 35–53. doi:10.1287/opre.1030.0065

[5] Andreas Brandstädt, Van Bang Le, and Jeremy P. Spinrad. 1999. *Graph Classes: A Survey.* SIAM.

[6] Jack Edmonds. 1965. Maximum matching and a polyhedron with 0,1-vertices. *J. Res. Nat. Bur. Standards Section B Math. and Math. Physics* 69B, 1 and 2 (1965), 125. doi:10.6028/jres.069B.013

[7] Bruno Escoffier, Jérôme Monnot, and Olivier Spanjaard. 2008. Some tractable instances of interval data minmax regret problems. *Oper. Res. Letters* 36, 4 (2008), 424–429. doi:10.1016/j.orl.2007.12.004

[8] Bruno Escoffier, Jérôme Monnot, and Olivier Spanjaard. 2008. Some Tractable Instances of Interval Data Minmax Regret Problems: Bounded Distance from Triviality. In *SOFSEM 2008: Theory and Practice of Computer Science.* Lecture Notes in Computer Science, Vol. 4910. Springer Berlin Heidelberg, Berlin, Heidelberg, 280–291. doi:10.1007/978-3-540-77566-9_24

[9] Marc Goerigk and Michael Hartisch. 2024. *An Introduction to Robust Combinatorial Optimization.* International Series in Operations Research & Management Science, Vol. 361. Springer.

[10] Adam Kasperski and Paweł Zieliński. 2017. Robust recoverable and two-stage selection problems. *Discret. Appl. Math.* 233 (2017), 52–64. doi:10.1016/j.dam.2017.08.014

[11] Adam Kasperski and Paweł Zieliński. 2017. Robust Two-Stage Network Problems. In *Oper. Res. Proceedings 2015.* Springer International Publishing, Cham, 35–40. doi:10.1007/978-3-319-42902-1_5

[12] Bernhard Korte and Jens Vygen (Eds.). 2012. *Combinatorial Optimization.* Springer Berlin Heidelberg, Berlin, Heidelberg. doi:10.1007/978-3-642-24488-9

[13] Panos Kouvelis and Gang Yu. 1997. *Robust discrete optimization and its applications.* Nonconvex Optimization and Its Applications, Vol. 14. Kluwer Acad. Publ, Dordrecht.

[14] Jacobo Valdes, Robert E. Tarjan, and Eugene L. Lawler. 1979. The recognition of Series Parallel digraphs. In *Proceedings of the STOC '79.* ACM Press, New York, New York, USA, 1–12. doi:10.1145/800135.804393

[15] Gerhard J. Woeginger. 2021. The trouble with the second quantifier. *4OR* 19, 2 (2021), 157–181. doi:10.1007/s10288-021-00477-y

Table 3: Summary of the complexity results. Previously known results are marked grey.

	min-max			min-max regret			two-stage		
	budgeted	interval	discrete	interval	budgeted	discrete	interval	budgeted	discrete
Paths	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	**NP**-hard
Trees	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	?	**NP**-hard
Cycles	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	polynomial	**NP**-hard
SPGs	polynomial	polynomial	**NP**-hard	polynomial	polynomial	**NP**-hard	polynomial	polynomial	**NP**-hard
bipartite	polynomial	polynomial	**NP**-hard	**NP**-hard	**NP**-hard	**NP**-hard	polynomial	**NP**-hard	**NP**-hard
complete	polynomial	polynomial	**NP**-hard	**NP**-hard	**NP**-hard	**NP**-hard	polynomial	**NP**-hard	**NP**-hard
general	polynomial	polynomial	**NP**-hard	**NP**-hard	**NP**-hard	**NP**-hard	polynomial	**NP**-hard	**NP**-hard