

---

# Min-max-min Robust Combinatorial Optimization Subject to Discrete Uncertainty

Christoph Buchheim · Jannis Kurtz

Received: date / Accepted: date

**Abstract** We consider combinatorial optimization problems with uncertain objective functions. In the min-max-min robust optimization approach, a fixed number  $k$  of feasible solutions is computed such that the respective best of them is optimal in the worst case. The idea is to calculate a set of candidate solutions in a potentially expensive preprocessing and then select the best solution out of this set in real-time, once the actual scenario is known.

In this paper, we investigate the complexity of this min-max-min problem in the case of discrete uncertainty, as well as its connection to the classical min-max robust counterpart. Essentially, it turns out that the min-max-min problem is not harder to solve than the min-max problem, while producing much better solutions in general.

**Keywords** Robust Optimization ·  $k$ -Adaptability · Discrete Uncertainty

## 1 Introduction

Uncertain data can occur in many real-world optimization problems. Typical examples of parameters which can be uncertain are the costs and demands in the vehicle routing problem, the returns of assets in financial applications, arrival times of jobs in scheduling problems and many more. In this paper we

---

This work has partially been supported by the German Research Foundation (DFG) within the Research Training Group 1855.

Christoph Buchheim  
Vogelpothsweg 87, 44227 Dortmund, Germany  
E-mail: christoph.buchheim@math.tu-dortmund.de

Jannis Kurtz  
Vogelpothsweg 87, 44227 Dortmund, Germany  
E-mail: jannis.kurtz@math.tu-dortmund.de

study combinatorial problems of the form

$$\min_{x \in X} c^\top x \quad (\text{M})$$

where  $X \subseteq \{0, 1\}^n$  is the set of all feasible solutions and only the cost vector  $c$  is uncertain. Typical examples for the latter problem are the shortest-path problem, the spanning-tree problem or the matching problem with uncertain costs, which will be considered in the following, among others.

One approach to handle these uncertainties is robust optimization, which was first introduced by Soyster [15] in 1973 and has received increasing attention since the seminal works of Ben-Tal and Nemirovski [6], El Ghaoui et al. [10], and Kouvelis and Yu [13] in the late 1990s. In this approach, the uncertain objective function is replaced by a so-called uncertainty set  $U$ , which contains all scenarios that are likely enough to be taken into account. Then the task is to find the best solution concerning the worst objective value over all scenarios. The robust optimization idea thus leads to the well known min-max problem

$$\min_{x \in X} \max_{c \in U} c^\top x. \quad (\text{M}^2)$$

The latter problem is known to be too conservative for many practical applications, since considering all possible scenarios can lead to solutions which are far from optimal in many scenarios [8]. To tackle this problem, recently many new approaches have been developed in the robust optimization literature. One of these approaches is the so called *adjustable robustness* and was introduced by Ben-Tal et al. [5]. The authors propose a two-stage model where the set of variables is decomposed into *here and now* variables  $x$  and *wait and see* variables  $y$ . The objective is to find a solution  $x$  such that for all possible scenarios there exists a  $y$  such that  $(x, y)$  is feasible and minimizes the worst case. This problem is known to be hard theoretically and practically. Therefore Bertsimas and Caramanis [7] introduced the concept of  $k$ -adaptability to approximate the latter problem. The idea is to compute  $k$  second-stage policies here-and-now; the best of these policies is chosen once the scenario is revealed. This idea was later used by Hanasusanto et al. [12] to approximate two-stage robust binary programs. The authors showed that if the uncertainty only affects the objective function, it is sufficient to calculate  $n + 1$  solution to reach the exact optimal value of the two-stage problem.

For the special case where no first stage exists, this idea was investigated in [9] in order to solve combinatorial problems with uncertainty in the objective function. The resulting optimization problem is

$$\min_{x^{(1)}, \dots, x^{(k)} \in X} \max_{c \in U} \min_{i=1, \dots, k} c^\top x^{(i)}. \quad (\text{M}^3)$$

The authors prove that (M<sup>3</sup>) can be solved in polynomial time for  $k \geq n + 1$  and for a convex uncertainty set  $U$  if the deterministic problem (M) can be solved in polynomial time. This result however cannot be extended to the case of discrete uncertainty, i.e., to the case that  $U$  is finite: different from (M<sup>2</sup>),

the uncertainty set  $U$  cannot be replaced by its convex hull in  $(M^3)$  without changing the problem.

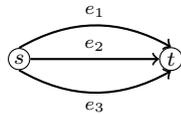
In this paper, we consider discrete uncertainty sets  $U = \{c_1, \dots, c_m\}$ . It was shown in [13,3] that  $(M^2)$  with discrete uncertainty is NP-hard for many combinatorial problems, even if the number  $m$  of scenarios is fixed, and that it is even strongly NP-hard if the number of scenarios is part of the input. On the other hand, for most of these problems pseudo-polynomial algorithm were found [13,2] for the case of fixed  $m$ . By reducing the min-max problem  $(M^2)$  to multicriteria optimization, it was further shown that many such problems admit an FPTAS [1].

In this paper, we investigate the complexity of  $(M^3)$  and show that, in spite of the larger generality, essentially the same results hold: in Section 3, we prove that for several classical combinatorial optimization problems the min-max-min problem  $(M^3)$  is NP-hard for fixed  $m$  and strongly NP-hard otherwise for any fixed  $k$ . In Section 4, we propose a pseudopolynomial algorithm for the case of fixed  $m$  which reduces  $(M^3)$  to  $(M^2)$ . Finally, we consider approximation schemes for  $(M^3)$  in Section 5.

## 2 Preliminaries

Let  $U = \{c_1, \dots, c_m\}$  be finite. First we show by a simple example that Problem  $(M^3)$  can yield a strictly better optimal value than Problem  $(M^2)$  even for  $k = 2$ . In fact, the improvement can be arbitrarily large.

*Example 1* Consider the graph  $G = (V, E)$  with  $V = \{s, t\}$  and  $E = \{e_1, e_2, e_3\}$  where  $e_i = (s, t)$  for each  $i$ :



Define the following scenarios  $U = \{c_1, c_2, c_3\}$ , where the  $j$ -th component of  $c_i$  is the cost of edge  $e_j$  in scenario  $i$ ,

$$c_1 = \begin{pmatrix} 1 \\ m \\ m+1 \end{pmatrix}, \quad c_2 = \begin{pmatrix} m+1 \\ m \\ 1 \end{pmatrix}, \quad c_3 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

where  $m$  is any positive integer. The optimal solution of Problem  $(M^2)$  is  $e_2$  with objective value  $m$ . Problem  $(M^3)$  with  $k = 2$  yields the optimal solution  $\{e_1, e_3\}$  and an optimal value of 1.  $\square$

Note that in the case of discrete uncertainty, the set  $U$  is not convex (unless it contains a single scenario), so that the results of [9] are not applicable. In

fact, different from the situation for Problem (M<sup>2</sup>), replacing  $U$  by its convex hull does not yield an equivalent problem in general, as the following example shows.

*Example 2* Consider again Example 1, where the optimal value of the min-max-min problem (M<sup>3</sup>) for  $k = 2$  was 1. If we replace  $U$  by its convex hull and consider the scenario

$$\frac{1}{2}(c_1 + c_2) = \begin{pmatrix} \frac{1}{2}m + 1 \\ m \\ \frac{1}{2}m + 1 \end{pmatrix} \in \text{conv}(U) ,$$

then we derive

$$\max_{c \in \text{conv}(U)} \min\{c^\top x^{(1)}, c^\top x^{(2)}\} \geq \frac{1}{2}m + 1$$

for all  $x^{(1)}, x^{(2)} \in X$ . □

### 3 Complexity

We now investigate the complexity of Problem (M<sup>3</sup>) for finite  $U = \{c_1, \dots, c_m\}$ . First note that the case  $k \geq m$  is trivial in this situation: an optimal solution for (M<sup>3</sup>) is then obtained by computing an optimal solution for each scenario  $c_i$  separately. The problem thus reduces to solving  $m$  certain linear optimization problems over  $X$ .

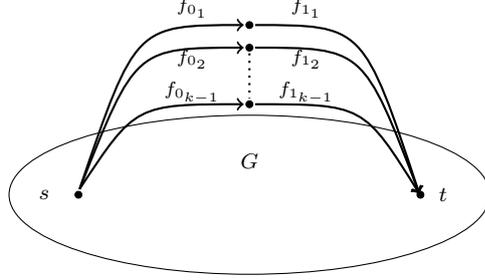
For  $k < m$ , this approach is not feasible. In fact, in this section we show that Problem (M<sup>3</sup>) becomes NP-hard for  $k < m$  in the discrete scenario case for the shortest path problem, the spanning tree problem and the matching problem. To this end, we will polynomially reduce the min-max problem (M<sup>2</sup>) to the min-max-min problem (M<sup>3</sup>) for the latter problems.

**Theorem 1** *For the shortest path problem on a graph  $G$  and for a discrete uncertainty set  $U = \{c_1, \dots, c_m\}$ , we can polynomially reduce Problem (M<sup>2</sup>) to Problem (M<sup>3</sup>) with at least  $m + k - 1$  scenarios, for any fixed  $k < m$ .*

*Proof* For any given instance of the min-max shortest path problem, i.e., a graph  $G = (V, E)$  and an uncertainty set  $U = \{c_1, \dots, c_m\}$  with  $c_i \geq 0$  for all  $i$ , we define the graph  $G_k^{sp}$  as follows: Add  $k - 1$  nodes  $v_1, \dots, v_{k-1}$  to  $G$  and add edges  $f_{0_i} = (s, v_i)$  and  $f_{1_i} = (v_i, t)$  for all  $i \in \{1, \dots, k - 1\}$ ; see Figure 1. The idea of the proof is to define scenarios on  $G_k^{sp}$  that force the min-max-min problem to choose each new path as one solution. The remaining  $k$ -th solution then must be the optimal min-max solution in  $G$ . To this end, we define scenarios  $\bar{c}_1, \dots, \bar{c}_m$  on  $G_k^{sp}$  by extending every scenario  $c_i \in U$  by  $M$  on the edges  $f_{1_1}, \dots, f_{1_{k-1}}$ , where  $M$  is a sufficiently large number that can be chosen as

$$M := \sum_{i=1}^m \sum_{j=1}^{|E|} (c_i)_j + 1.$$

Then we add scenarios  $d_1, \dots, d_{k-1}$  such that in scenario  $d_i$  all edges in  $G$  and all edges  $f_{1_1}, \dots, f_{1_{k-1}}$  have costs  $M$ , except for edge  $f_{1_i}$  which has cost zero. The edges  $f_{0_1}, \dots, f_{0_{k-1}}$  have zero costs in all scenarios. Note that we only added the  $k-1$  new nodes and therefore the edges  $f_{0_1}, \dots, f_{0_{k-1}}$  to avoid multigraphs.



**Fig. 1** The graph  $G_k^{sp}$ .

Now if we choose a solution  $\{x^{(1)}, \dots, x^{(k)}\}$  of the min-max-min shortest-path problem on  $G_k^{sp}$  where  $x^{(1)}, \dots, x^{(k-1)}$  are the added paths  $\{f_{0_i}, f_{1_i}\}$  and  $x^{(k)}$  is any feasible solution in  $G$  then the objective value

$$\max_{c \in \{\bar{c}_1, \dots, \bar{c}_m, d_1, \dots, d_{k-1}\}} \min_{i=1, \dots, k} c^\top x^{(i)}$$

must be strictly lower than  $M$  since on  $\bar{c}_i$  the minimum is attained by  $x^{(k)}$  and on  $d_i$  the minimum is attained by  $x^{(i)}$  and all values are strictly lower than  $M$  by definition of  $M$ . So we know that every optimal solution of the min-max-min problem on the graph  $G_k^{sp}$  must contain the paths  $\{f_{0_i}, f_{1_i}\}$  since otherwise, if for any  $i_0$  the path is not contained, then in scenario  $d_{i_0}$  the minimum  $\min_{i=1, \dots, k} c^\top x^{(i)}$  is greater than  $M$  and therefore

$$\max_{c \in \{\bar{c}_1, \dots, \bar{c}_m, d_1, \dots, d_{k-1}\}} \min_{i=1, \dots, k} c^\top x^{(i)} \geq M$$

which cannot be optimal. On the other hand, by the same reasoning applied to the scenarios  $\bar{c}_i$ , in every optimal solution there must be contained a solution which only uses edges in  $G$ . So let  $\{x^{(1)}, \dots, x^{(k)}\}$  be an optimal solution where w.l.o.g.  $x^{(k)}$  is the path in  $G$  and  $x^{(i)}$  is the path  $\{f_{0_i}, f_{1_i}\}$  for  $i = 1, \dots, k-1$ . Then we have

$$\max_{c \in \{d_1, \dots, d_{k-1}\}} \min_{i=1, \dots, k} c^\top x^{(i)} = 0$$

by definition of the scenarios  $\{d_1, \dots, d_{k-1}\}$ . On the other hand

$$\max_{c \in \{\bar{c}_1, \dots, \bar{c}_m\}} \min_{i=1, \dots, k} c^\top x^{(i)} = \max_{c \in \{c_1, \dots, c_m\}} c^\top x^{(k)} > 0$$

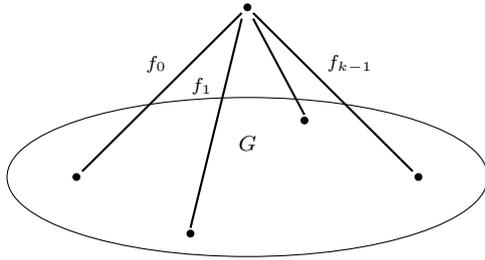
and therefore

$$\min_{x^{(1)}, \dots, x^{(k)}} \max_{c \in \{\bar{c}_1, \dots, \bar{c}_m, d_1, \dots, d_{k-1}\}} \min_{i=1, \dots, k} c^\top x^{(i)} = \min_x \max_{c \in \{c_1, \dots, c_m\}} c^\top x.$$

So the min-max optimal solution must be contained in  $\{x^{(1)}, \dots, x^{(k)}\}$ .  $\square$

**Theorem 2** *The assertion of Theorem 1 also holds for the minimum spanning tree problem on graphs with at least  $k$  nodes.*

*Proof* The proof is similar to the proof of Theorem 1. Let a graph  $G = (V, E)$  with  $|V| \geq k$  and an uncertainty set  $U = \{c_1, \dots, c_m\}$  be given. We then define the graph  $G_k^{st}$  as follows: Add one node  $w$  to  $G$  and add edges  $f_i = \{v_i, w\}$  for all  $i \in \{0, \dots, k-1\}$ , where the  $v_i$  are arbitrary pairwise different nodes in  $G$ ; see Figure 2.



**Fig. 2** The graph  $G_k^{st}$ .

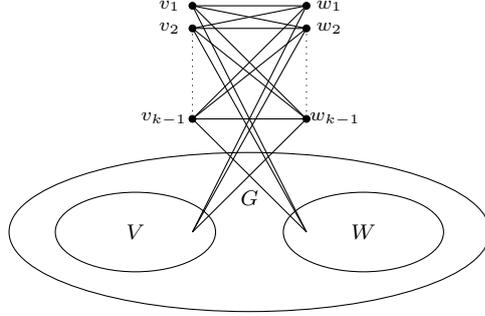
Again the idea of the proof is to define scenarios on  $G_k^{st}$  that force the min-max-min problem to choose exactly one new edge for each solution contained in the optimal solution. To this end, we define scenarios  $\bar{c}_1, \dots, \bar{c}_m$  on  $G_k^{st}$  by extending every scenario  $c_i \in U$  by zero on  $f_0$  and by  $M$  on the edges  $f_1, \dots, f_{k-1}$ , where  $M$  can be chosen as

$$M := \sum_{i=1}^m \sum_{j=1}^{|E|} |(c_i)_j| + 1.$$

Then we add scenarios  $d_1, \dots, d_{k-1}$  such that in scenario  $d_i$  all edges in  $G$  have costs  $M$  and all edges  $f_0, \dots, f_{k-1}$  have costs  $(|V| + 1)M$ , except for edge  $f_i$  which has costs  $-|V|M$ . Now by the same reasoning as in the proof of Theorem 1 every optimal solution of min-max-min on  $G_k^{st}$  consists of solutions  $x^{(0)}, \dots, x^{(k-1)}$  where  $x^{(i)}$  uses only edge  $f_i$  of the new edges. Then the projection of  $x^{(0)}$  on  $G$  is the optimal min-max solution.  $\square$

**Theorem 3** *The assertion of Theorem 1 also holds for the matching problem.*

*Proof* The proof is similar to the proof of Theorem 1. For a given instance of the min-max assignment problem, i.e., a bipartite graph  $G = (V, W, E)$  and an uncertainty set  $U = \{c_1, \dots, c_m\}$ , we define the graph  $G_k^{as}$  as follows: Add nodes  $v_1, \dots, v_{k-1}$  and  $w_1, \dots, w_{k-1}$  to  $G$ . Moreover, add edges  $f_{v_i w} = \{v_i, w\}$  for all  $v_i$  and all  $w \in W \cup \{w_1, \dots, w_{k-1}\}$ , as well as edges  $f_{v w_i} = \{v, w_i\}$  for each  $w_i$  and all  $v \in V$ ; see Figure 3.



**Fig. 3** The graph  $G_k^{as}$ .

Again the idea of the proof is to define scenarios on  $G_k^{as}$  that force the min-max-min problem to choose exactly one of the edges  $f_{v_i w_i}$  for each solution contained in the optimal solution. To this end, we define scenarios  $\bar{c}_1, \dots, \bar{c}_m$  on  $G_k^{as}$  by extending every scenario  $c_i \in U$  by zero on  $f_{v_i w_i}$  for  $i = 1, \dots, k-1$  and by  $2M$  on all other new edges, where  $M$  can be chosen as

$$M := \sum_{i=1}^m \sum_{j=1}^{|E|} |(c_i)_j| + 1.$$

Then we add scenarios  $d_1, \dots, d_{k-1}$  such that in scenario  $d_i$  edge  $f_{v_i w_i}$  has cost  $-2M$  while the edges  $f_{v_i w}$  and  $f_{v w_i}$  for all  $w \neq w_i$  and  $v \neq v_i$  and all edges  $f_{v_j w_j}$  with  $j \neq i$  have costs  $3M$ . All remaining edges have costs  $\frac{M}{n+k-2}$ . Now by the same reasoning as in the proof of Theorem 1, every optimal solution of min-max-min on  $G_k^{as}$  consists of  $x^{(1)}, \dots, x^{(k)}$  where for each  $i = 1, \dots, k-1$  solution  $x^{(i)}$  uses edge  $f_{v_i w_i}$  and none of the edges  $f_{v_j w_j}$  with  $j \neq i$ . Moreover  $x^{(k)}$  uses all edges  $f_{v_i w_i}$  for  $i = 1, \dots, k-1$  and coincides with an optimal solution of the min-max problem on  $G$ .  $\square$

**Corollary 1** For any fixed  $k \in \mathbb{N}$  and fixed  $m > k$ , Problem (M<sup>3</sup>) is NP-hard for the shortest path problem, the minimum spanning tree problem and the matching problem for uncertainty sets  $U$  with  $|U| = m$ .

*Proof* The min-max variants of these problems are NP-hard for  $m \geq 2$  by [13]. From Theorems 1, 2, and 3, we derive that the min-max-min variants of the same problems are NP-hard if the number of scenarios is at least  $k+1$ .  $\square$

**Corollary 2** For any fixed  $k \in \mathbb{N}$ , Problem (M<sup>3</sup>) for finite  $U$  is strongly NP-hard for the shortest path problem and the minimum spanning tree problem.

*Proof* The min-max variants of both problems are strongly NP-hard by [13] and the constructions in Theorems 1 and 2 are polynomial in  $|U|$ .  $\square$

#### 4 Pseudopolynomial Algorithms

In [13], pseudo-polynomial algorithms are given for many combinatorial min-max problems with a bounded number of scenarios. In the following, we will show how to polynomially reduce Problem (M<sup>3</sup>) to Problem (M<sup>2</sup>) for discrete uncertainty and a bounded number of scenarios. This implies that Problem (M<sup>3</sup>) is in fact only weakly NP-hard for many combinatorial problems. At the same time, the reduction shows that we can solve Problem (M<sup>3</sup>) in polynomial time if we can solve Problem (M<sup>2</sup>) in polynomial time, which is the case, e.g., for the minimum cut problem.

---

**Algorithm 1** Reduction from Problem (M<sup>3</sup>) to Problem (M<sup>2</sup>) for  $k < m$

---

**Input:**  $U = \{c_1, \dots, c_m\}$ ,  $X \subseteq \{0, 1\}^n$ ,  $k < m$

**Output:** optimal solution  $\{x^{(1)}, \dots, x^{(k)}\}$  of Problem (M<sup>3</sup>)

1:  $v := \infty$

2: **for** all  $k$ -partitions  $U_1, \dots, U_k$  of  $U$  **do**

3:   **if**  $\max\{\min_{x \in X} \max_{c \in U_1} c^\top x, \dots, \min_{x \in X} \max_{c \in U_k} c^\top x\} < v$  **then**

4:      $v = \max\{\min_{x \in X} \max_{c \in U_1} c^\top x, \dots, \min_{x \in X} \max_{c \in U_k} c^\top x\}$

5:      $x^{(i)} = \arg \min_{x \in X} \max_{c \in U_i} c^\top x \quad \forall i = 1, \dots, k$

6:   **end if**

7: **end for**

8: **return**  $\{x^{(1)}, \dots, x^{(k)}\}$

---

**Theorem 4** Algorithm (1) calculates an optimal solution for Problem (M<sup>3</sup>).

*Proof* Let  $\{\bar{x}^{(1)}, \dots, \bar{x}^{(k)}\} \subseteq X$  be an optimal solution of Problem (M<sup>3</sup>). Choose a partition  $\bar{U}_1, \dots, \bar{U}_k$  of  $U$  such that

$$\bar{U}_i \subseteq \{c \in U \mid c^\top \bar{x}^{(i)} \leq c^\top \bar{x}^{(j)} \quad \forall j \in \{1, \dots, k\}\} \quad \forall i = 1, \dots, k;$$

thus  $\bar{U}_i$  is the set of scenarios covered by the solution  $\bar{x}^{(i)}$ . Then

$$\begin{aligned} \min_{x \in X} \max_{c \in \bar{U}_i} c^\top x &\leq \min_{j=1, \dots, k} \max_{c \in \bar{U}_i} c^\top \bar{x}^{(j)} \\ &\leq \max_{c \in \bar{U}_i} c^\top \bar{x}^{(i)} \\ &= \max_{c \in \bar{U}_i} \min\{c^\top \bar{x}^{(1)}, \dots, c^\top \bar{x}^{(k)}\} \end{aligned}$$

and hence

$$\max \left\{ \min_{x \in X} \max_{c \in \bar{U}_1} c^\top x, \dots, \min_{x \in X} \max_{c \in \bar{U}_k} c^\top x \right\} \leq \max_{c \in U} \min\{c^\top \bar{x}^{(1)}, \dots, c^\top \bar{x}^{(k)}\},$$

which shows the result.  $\square$

**Corollary 3** *There exist pseudo-polynomial algorithms for Problem (M<sup>3</sup>) with fixed number of scenarios for the shortest path problem, the spanning tree problem, the perfect matching problem in planar graphs, and the knapsack problem.*

*Proof* Algorithm (1) needs to solve an instance of Problem (M<sup>2</sup>) a polynomial number of times if the number of scenarios is fixed. Thus using the pseudo-polynomial algorithms devised in [13] and [2] together with Algorithm (1) proves the result.  $\square$

**Corollary 4** *There exists a polynomial time algorithm for Problem (M<sup>3</sup>) with fixed number of scenarios for the minimum cut problem.*

*Proof* It was shown in [4] that the min-max version of the minimum cut problem can be solved in polynomial time for a fixed number of scenarios.  $\square$

## 5 Approximation Complexity

In this section, we show that Problem (M<sup>3</sup>) admits an FPTAS for many combinatorial optimization problems if the number of scenarios is fixed. For this, we prove that any min-max-min instance with uncertainty set  $U = \{c_1, \dots, c_m\}$  and feasible set  $X$  can be approximated using any approximation algorithm of the related multi-objective problem

$$\min_{x \in X} \begin{pmatrix} c_1^\top x \\ \vdots \\ c_m^\top x \end{pmatrix}. \quad (1)$$

It was shown in [14] and [11] that Problem (1) admits an FPTAS for the shortest path problem, the minimum spanning tree problem, and the knapsack problem, provided that the number  $m$  of criteria is fixed. More precisely, this means that for any  $\varepsilon > 0$  there exists a polynomial time algorithm which calculates a set  $F \subseteq X$  of polynomial size such that for every efficient solution  $x$ , the set  $F$  contains a feasible solution  $y$  such that  $c_i^\top y \leq (1 + \varepsilon)c_i^\top x$  for each scenario  $c_i \in U$ .

The following proof is based on the idea of the corresponding result for the min-max problem presented in [?].

**Theorem 5** *Given any function  $f : \mathbb{N} \rightarrow [1, \infty)$ , if Problem (1) admits a polynomial-time  $f(n)$ -approximation algorithm, then also Problem (M<sup>3</sup>) for fixed  $k$  admits a polynomial-time  $f(n)$ -approximation algorithm.*

*Proof* Let  $F \subseteq X$  be an  $f(n)$ -approximation of the set of efficient solutions, i.e., for any feasible solution  $x \in X$  there exists a feasible solution  $y \in F$  such that  $c_i^\top y \leq f(n)c_i^\top x$  for all  $i = 1, \dots, m$ . Note that at least one optimal solution  $\{x^{(1)}, \dots, x^{(k)}\}$  of Problem (M<sup>3</sup>) consists of only efficient solutions of

Problem (1), as each non-efficient solution can be replaced by a dominating solution without increasing the objective function value of  $(M^3)$ .

So let  $\{x^{(1)}, \dots, x^{(k)}\}$  be an optimal solution where each  $x^{(i)}$  is efficient. Thus for every  $x^{(i)}$  there exists  $y^{(i)} \in F$  with  $c^\top y^{(i)} \leq f(n)c^\top x^{(i)}$  for all  $c \in U$ . Therefore

$$\min_{i=1, \dots, k} c^\top y^{(i)} \leq f(n) \min_{i=1, \dots, k} c^\top x^{(i)}$$

for all  $c \in U$ . The desired algorithm for Problem  $(M^3)$  first computes  $F$  and then chooses a solution  $\{z^{(1)}, \dots, z^{(k)}\} \subseteq F$  which minimizes the objective function of Problem  $(M^3)$ . This can be done in polynomial time by the polynomial size of  $F$  and since  $k$  is fixed. In summary, we have

$$\max_{c \in U} \min_{i=1, \dots, k} c^\top z^{(i)} \leq \max_{c \in U} \min_{i=1, \dots, k} c^\top y^{(i)} \leq f(n) \max_{c \in U} \min_{i=1, \dots, k} c^\top x^{(i)}$$

which proves the result.  $\square$

**Corollary 5** *Problem  $(M^3)$  admits an FPTAS for the shortest path problem, the minimum spanning tree problem, and the knapsack problem if the number of scenarios is fixed.*

*Proof* It was shown in [14] and [11] that Problem (1) for the shortest path problem, the minimum spanning tree problem, and the knapsack problem admits an FPTAS, if the number of objective criteria is fixed. The result thus follows from Theorem 5.  $\square$

**Acknowledgements** We would like to thank David Adjiashvili for fruitful discussions.

## References

1. Aissi, H., Bazgan, C., Vanderpooten, D.: Approximation complexity of min-max (regret) versions of shortest path, spanning tree, and knapsack. In: Algorithms – ESA 2005, *Lecture Notes in Computer Science*, vol. 3669, pp. 862–873. Springer (2005)
2. Aissi, H., Bazgan, C., Vanderpooten, D.: Pseudo-polynomial algorithms for min-max and min-max regret problems. In: 5th International Symposium on Operations Research and its Applications (ISORA 2005), pp. 171–178 (2005)
3. Aissi, H., Bazgan, C., Vanderpooten, D.: Min-max and min-max regret versions of combinatorial optimization problems: A survey. *European Journal of Operational Research* **197**(2), 427–438 (2009)
4. Armon, A., Zwick, U.: Multicriteria global minimum cuts. *Algorithmica* **46**(1), 15–26 (2006)
5. Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A.: Adjustable robust solutions of uncertain linear programs. *Mathematical Programming* **99**(2), 351–376 (2004)
6. Ben-Tal, A., Nemirovski, A.: Robust convex optimization. *Mathematics of Operations Research* **23**(4), 769–805 (1998)
7. Bertsimas, D., Caramanis, C.: Finite adaptability in multistage linear optimization. *Automatic Control, IEEE Transactions on* **55**(12), 2751–2766 (2010)
8. Bertsimas, D., Sim, M.: The price of robustness. *Operations Research* **52**(1), 35–53 (2004)
9. Buchheim, C., Kurtz, J.: Min-max-min robust combinatorial optimization. *Optimization Online* (2016)

10. El Ghaoui, L., Lebret, H.: Robust solutions to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications* **18**(4), 1035–1064 (1997)
11. Erlebach, T., Kellerer, H., Pferschy, U.: Approximating multiobjective knapsack problems. *Management Science* **48**(12), 1603–1612 (2002)
12. Hanasusanto, G.A., Kuhn, D., Wiesemann, W.: K-adaptability in two-stage robust binary programming. *Optimization Online* (2015)
13. Kouvelis, P., Yu, G.: *Robust Discrete Optimization and Its Applications*. Springer (1996)
14. Papadimitriou, C.H., Yannakakis, M.: On the approximability of trade-offs and optimal access of web sources. In: *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pp. 86–92. IEEE (2000)
15. Soyster, A.L.: Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research* **21**(5), 1154–1157 (1973)