Randomized Minmax Regret for Combinatorial Optimization Under Uncertainty*

Andrew Mastin[†] Patrick Jaillet[‡] Sang Chin[§]

January 29, 2014

Abstract

The minmax regret problem for combinatorial optimization under uncertainty can be viewed as a zero-sum game played between an optimizing player and an adversary, where the optimizing player selects a solution and the adversary selects costs with the intention of maximizing the regret of the player. Existing minmax regret models consider only deterministic solutions/strategies, and minmax regret versions of most polynomial solvable problems are NP-hard. In this paper, we consider a randomized model where the optimizing player selects a probability distribution (corresponding to a mixed strategy) over solutions and the adversary selects costs with knowledge of the player's distribution, but not its realization. We show that under this randomized model, the minmax regret version of any polynomial solvable combinatorial problem becomes polynomial solvable. This holds true for both the interval and discrete scenario representations of uncertainty. We also show that the maximum expected regret value under the randomized model is upper bounded by the regret under the deterministic model.

1 Introduction

Many optimization applications involve cost coefficients that are not fully known. When distributional information on cost coefficients is available (e.g. from historical data or other estimates), stochastic programming is often an appropriate modeling choice [12, 22]. In other cases, costs may only be known to be contained in intervals (i.e. each cost has a known lower and upper bound), or to be a member of a finite set of scenarios, and one is more interested in worst-case performance. Robust optimization formulations are desirable here as they employ a minmax-type objective and do not require knowledge of cost distributions [19, 10, 17].

In a general robust optimization problem with cost uncertainty, one must select a set of items from some feasible *solution set*, such that item costs are unknown but must be contained in a known *uncertainty set*. Under the well known *minmax* objective (also referred to as absolute robustness), the goal is to select a solution that gives the best upper bound on objective cost over all possible costs from the uncertainty set [23]. That is, one must select the solution that, when item costs are chosen to maximize the cost of the selected solution, is minimum. Under the *minmax regret* objective

*Research supported in part by NASA ESTOs Advanced Information System Technology (AIST) program under grant number NNX12H81G. Also supported by NSF grant 1029603 and ONR grant N00014-12-1-0033.

†Laboratory for Information and Decision Systems, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, USA; mastin@mit.edu

‡Laboratory for Information and Decision Systems, Department of Electrical Engineering and Computer Science and Operations Research Department, Massachusetts Institute of Technology, Cambridge, MA 02139, USA; jaillet@mit.edu

§Draper Laboratory, 555 Technology Square, Cambridge, MA 02139; schin@draper.com

(sometimes called the robust deviation model), the goal is instead to select the solution that minimizes the maximum possible regret, defined as the difference between the cost of the selected solution and the optimal solution [21].

A problem under the minmax regret objective can be viewed as a two stage game where in the first stage the optimizing player selects a deterministic solution, and in the second stage an adversary observes the selected solution and chooses costs from the uncertainty set with the intention of maximizing the player's regret. The goal of the optimizing player is thus to select a solution that least allows the adversary to generate regret. For both interval and discrete scenario representations of cost uncertainty, the minmax regret versions of most polynomial solvable problems are NP-hard [3]. A variation on this model, first suggested by Bertsimas et al. [8] for minmax robust optimization, is to allow the optimizing player to select a probability distribution over solutions and require the adversary to select costs based only on knowledge of the players distribution, but not its realization. In this paper, we show that under this randomized model, the minmax regret version of any polynomial solvable 0-1 integer programming problem becomes polynomial solvable. This holds true for both the interval and discrete scenario representations of uncertainty. We also show that the maximum expected regret under the randomized model is no greater than the corresponding regret under the deterministic model.

The paper is structured as follows. In the remainder of this section we review related work; Section 2 introduces notation and definitions. Section 3 presents the analysis for the interval representation of uncertainty and Section 4 considers the discrete scenario representation of uncertainty. A conclusion is given in Section 5.

1.1 Related work. One of the first studies of minmax regret from both an algorithmic and complexity perspective was that of Averbakh [4]. He looked at the minmax regret version of the simple problem of selecting k items out of n total items where the cost of each item is uncertain, and the goal is to select the set of items with minimum total cost. For interval uncertainty, he derived a polynomial time algorithm based on interchange arguments. He demonstrated that for the discrete scenario representation of uncertainty, however, the minmax regret problem becomes NP-hard, even for the case of only two scenarios. It is interesting to contrast these results with the case of general minmax regret linear programming, which as shown by Averbakh and Lebedev [6], is NP-hard for interval uncertainty but polynomial solvable for discrete scenario uncertainty.

Apart from the item selection problem, most polynomial solvable minmax regret combinatorial problems are NP-hard, both for interval and discrete scenario uncertainty. This is true for the shortest path, minimum spanning tree, assignment, and minimum s-t cut problems [24, 19, 5, 1, 2]. One exception is the minimum cut problem, the minmax regret version of which is polynomial solvable both for interval and discrete scenario uncertainty [2]. The survey paper of Aissi et al. [3] provides a comprehensive summary of results related to both minmax and minmax regret combinatorial problems. For problems that are already NP-complete, most of their minmax regret versions are Σ_2^p-complete (meaning that they are at the second level of the polynomial hierarchy) [16]. To solve minmax regret problems in practice, the book by Kasperski reviews standard mixed integer program (MIP) formulations for both interval and discrete scenario uncertainty [17].

The application of a game theoretic model with mixed strategies to robust optimization problems was introduced by Bertsimas et al. [8]. They focused on the minmax robust model, and their analysis was motivated by adversarial models used for online optimization algorithms. As described by Ben-David et al. [7] (see also Borodin and El-Yaniv [13]), the three types of adversaries are the *oblivious adversary*, the *adaptive online adversary*, and the *adaptive offline adversary*. The adaptive offline adversary is the analog of the conventional deterministic minmax regret problem, while the adaptive online adversary corresponds to our randomized model. The analog of the oblivious adversary, which we do study, is the model where the adversary first selects costs, and the optimizing player then selects the solution after viewing these costs.

For the randomized (corresponding to the adaptive online adversary) minmax problem, Bertsimas et al. [8] showed that if it is possible to optimize over both the solution set and the uncertainty set in polynomial time, then an optimal mixed strategy solution can be calculated in polynomial time, and that the expected regret under the randomized model is no greater than the regret for the deterministic model. This holds despite the fact that solving the minmax version of many polynomial solvable problems is NP-hard for the deterministic case [11]. They also bound the improvement gained from randomization for various uncertainty sets. Our work is similar to theirs, but we focus on the minmax regret objective instead of the minmax objective.

Another line of research that is related to ours is in security applications, where the adversarial model is realistically motivated. Korzhyk, et al. [18] consider assignment-type problems where defensive resources, such as security guards, must be assigned to valued targets. They follow a Stackelberg model where the defending player has the power to commit to a mixed strategy; the attacker then observes this mixed strategy (though not the realization) and decides which targets to attack. They use linear programming formulations along with the Birkhoff-von Neumann theorem to find polynomial-sized optimal mixed strategies. Another similar work is that of Bertismas et al. which looks at randomized strategies for network interdiction [9].

2 Definitions

We consider a general combinatorial optimization problem where we are given a set of n items $E = \{e_1, e_2, \ldots, e_n\}$ and a set $\mathcal{F}$ of feasible subsets of E. Each item $e \in E$ has a cost $c_e \in \mathbb{R}_+$. Given the vector $c = (c_1, \ldots, c_n)$, the goal of the optimization problem is to select the feasible subset of items that minimizes the total cost; we refer to this as the *nominal problem*:

$$F^*(c) := \min_{T \in \mathcal{F}} \sum_{e \in T} c_e. \tag{2.1}$$

Let $x = (x_1, \ldots, x_n)$ be a characteristic vector for some set T, so that $x_e = 1$ if $e \in T$ and 0 otherwise. Also let $\mathcal{X} \subseteq \{0,1\}^n$ denote the set of all characteristic vectors corresponding to feasible sets $T \in \mathcal{F}$. We assume that $\mathcal{X}$ is described in size m (e.g. with m linear inequalities). We can equivalently write the nominal problem with a linear objective function:

$$F^*(c) = \min_{x \in \mathcal{X}} c^\top x, \tag{2.2}$$

where $c^\top x$ denotes the inner product of c and x. The cost vector c is unknown, but assumed to be an element of some uncertainty set $\mathcal{C}$.

We will review the conventional regret definitions for the deterministic minmax regret framework, and then present the analogous definitions for our randomized model. For some $x \in \mathcal{X}$ and a cost vector $c \in \mathcal{C}$, the deterministic cost of a solution is

$$F(x, c) := \sum_{e \in E} c_e x_e. \tag{2.3}$$

The regret of a solution x under some cost vector c is the difference between the cost of the solution and the optimal cost:

$$R(x, c) := F(x, c) - F^*(c). \tag{2.4}$$

The *maximum regret* problem for a solution x is

$$R_{\max}(x) := \max_{c \in C} R(x, c) = \max_{c \in C} \left(F(x, c) - F^*(c) \right). \tag{2.5}$$

The *deterministic minmax regret* problem is then

$$Z_{\text{D}} := \min_{x \in \mathcal{X}} R_{\max}(x) = \min_{x \in \mathcal{X}} \max_{c \in \mathcal{C}} (F(x, c) - F^*(c)). \tag{2.6}$$

We now move to the randomized framework, where solutions are selected according to a probability distribution. For some set $T \in \mathcal{F}$, let y_T denote the probability that the subset T is selected. Let $y = (y_T)_{T \in \mathcal{F}}$ be the vector of length $|\mathcal{F}|$ specifying the subset selection distribution; we will refer to y simply as a solution. Define the feasible region for y as

$$\mathcal{Y} := \{y | y \geq \mathbf{0}, \mathbf{1}^\top y = 1\}, \tag{2.7}$$

where the notation $\mathbf{0}$ and $\mathbf{1}$ indicates a full vector of zeros and ones, respectively. For some solution y and cost vector c, the expected cost of the solution is

$$\overline{F}(y, c) := \sum_{T \in \mathcal{F}} y_T \sum_{e \in T} c_e. \tag{2.8}$$

The expected regret is then

$$\overline{R}(y, c) := \overline{F}(y, c) - F^*(c). \tag{2.9}$$

The *maximum expected regret* problem for a solution y is

$$\overline{R}_{\max}(y) := \max_{c \in \mathcal{C}} \overline{R}(y, c) = \max_{c \in \mathcal{C}} (\overline{F}(y, c) - F^*(c)). \tag{2.10}$$

Finally, the minmax expected regret problem, which we refer to as the *randomized minmax regret* problem, is stated as

$$Z_{\text{R}} := \min_{y \in \mathcal{Y}} \overline{R}_{\max}(y) = \min_{y \in \mathcal{Y}} \max_{c \in \mathcal{C}} (\overline{F}(y, c) - F^*(c)). \tag{2.11}$$

3 Interval uncertainty

In this section we assume that the set $\mathcal{C}$ is characterized by interval uncertainty, that is

$$c_e \in [c_e^-, c_e^+], \quad \forall e \in E. \tag{3.12}$$

Define the region

$$\mathcal{I} := \{c | c_e \in [c_e^-, c_e^+], e \in E\}. \tag{3.13}$$

The resulting deterministic minmax regret problem problem has been well studied and can be solved with a MIP [17]. We use the following unconventional formulation, which has an exponential number of constraints. We will show that the randomized minmax regret problem corresponds to the linear programming relaxation of this formulation.

Lemma 1. *For interval uncertainty, the deterministic minmax regret problem* (2.6) *is equivalent to the following integer program.*

$$\begin{aligned} Z_{\mathrm{D}} \;=\; \min \quad & z \\ \text{s.t.} \quad & \sum_{e \in E \setminus T} c_e^+ x_e - \sum_{e \in T} c_e^- \left(1 - x_e\right) \le z, \qquad \forall T \in \mathcal{F}, \\ & x \in \mathcal{X}. \end{aligned} \tag{3.14}$$

Proof. From the maximum regret definition (2.5),

$$\begin{aligned} R_{\max}(x) &= \max_{c \in \mathcal{I}} \left(F(x,c) - F^*(c)\right) \\ &= \max_{c \in \mathcal{I}} \left(\sum_{e \in E} c_e x_e - \min_{T \in \mathcal{F}} \sum_{e \in T} c_e \right) \\ &= \max_{T \in \mathcal{F}} \max_{c \in \mathcal{I}} \left(\sum_{e \in E} c_e x_e - \sum_{e \in T} c_e \right) \\ &= \max_{T \in \mathcal{F}} \max_{c \in \mathcal{I}} \left(\sum_{e \in E \setminus T} c_e x_e - \sum_{e \in T} c_e (1 - x_e) \right) \\ &= \max_{T \in \mathcal{F}} \left(\sum_{e \in E \setminus T} c_e^+ x_e - \sum_{e \in T} c_e^- (1 - x_e) \right), \end{aligned} \tag{3.15}$$

where in the third equality we have used that the expression $\sum_{e \in E} c_e x_e$ is not a function of T, and the last equality follows since $x_e \in \{0, 1\}$. The program is then valid by the definition of the maximum. □

In the randomized framework, we begin by considering the maximum expected regret problem (2.10). This problem is interpreted as the problem solved by the adversary for a given distribution y; that is, the adversary must select a *best response* for y. It is a well known result from game theory that there always exists a pure strategy best response for an opponent strategy, so we can consider deterministic solutions to this problem without loss of generality [20].

Lemma 2. *For interval uncertainty, the maximum expected regret problem* (2.10) *is equivalent to the problem*

$$\max_{T \in \mathcal{F}} \left(\sum_{e \in E \setminus T} c_e^+ \sum_{U \in \mathcal{F} : e \in U} y_U - \sum_{e \in T} c_e^- \left(1 - \sum_{U \in \mathcal{F} : e \in U} y_U \right) \right). \tag{3.16}$$

Proof. Starting with (2.10), we have

$$\begin{aligned}
\overline{R}_{\max}(y) &= \max_{c\in\mathcal{I}}(\overline{F}(y,c) - F^*(c)) \\
&= \max_{c\in\mathcal{I}}\left(\sum_{U\in\mathcal{F}} y_U \sum_{e\in U} c_e - \min_{T\in\mathcal{F}}\left(\sum_{e\in T} c_e\right)\right) \\
&= \max_{c\in\mathcal{I}}\left(\sum_{e\in E} c_e \sum_{U\in\mathcal{F}:e\in U} y_U - \min_{T\in\mathcal{F}}\left(\sum_{e\in T} c_e\right)\right) \\
&= \max_{c\in\mathcal{I}}\max_{T\in\mathcal{F}}\left(\sum_{e\in E} c_e \sum_{U\in\mathcal{F}:e\in U} y_U - \sum_{e\in T} c_e\right) \\
&= \max_{T\in\mathcal{F}}\max_{c\in\mathcal{I}}\left(\sum_{e\in E\setminus T} c_e \sum_{U\in\mathcal{F}:e\in U} y_U - \sum_{e\in T} c_e\left(1 - \sum_{U\in\mathcal{F}:e\in U} y_U\right)\right),
\end{aligned} \tag{3.17}$$

where in the third equality we have used that the expression $\sum_{e\in E} c_e \sum_{U\in\mathcal{F}:e\in U} y_U$ is not a function of T, and the other equalities follow from rearranging terms. Notice in (3.17) that $\sum_{U\in\mathcal{F}:e\in U} y_U$ is simply the total probability that item e is selected, so for $y \in \mathcal{Y}$, we must have $\sum_{U\in\mathcal{F}:e\in U} y_U \in [0,1]$. This makes it easy to see that for a given $T \in \mathcal{F}$,

$$\begin{aligned}
&\max_{c\in\mathcal{I}}\left(\sum_{e\in E\setminus T} c_e \sum_{U\in\mathcal{F}:e\in U} y_U - \sum_{e\in T} c_e\left(1 - \sum_{U\in\mathcal{F}:e\in U} y_U\right)\right) \\
&\quad= \sum_{e\in E\setminus T} c_e^+ \sum_{U\in\mathcal{F}:e\in U} y_U - \sum_{e\in T} c_e^-\left(1 - \sum_{U\in\mathcal{F}:e\in U} y_U\right).
\end{aligned} \tag{3.18}$$

Substituting (3.18) into (3.17) then gives an optimization problem with a finite number of feasible solutions,

$$\overline{R}_{\max}(y) = \max_{T\in\mathcal{F}}\left(\sum_{e\in E\setminus T} c_e^+ \sum_{U\in\mathcal{F}:e\in U} y_U - \sum_{e\in T} c_e^-\left(1 - \sum_{U\in\mathcal{F}:e\in U} y_U\right)\right), \tag{3.19}$$

which completes the proof. □

An immediate corollary of Lemma 2 is that we can solve the maximum expected regret problem for a given y by enumerating all $|\mathcal{F}|$ subsets (potentially an exponential number of them) and choosing the one that maximizes the argument of (3.19). The entire randomized minmax regret problem is now

$$\min_{y\in\mathcal{Y}} \overline{R}_{\max}(y) = \min_{y\in\mathcal{Y}}\max_{T\in\mathcal{F}}\left(\sum_{e\in E\setminus T} c_e^+ \sum_{U\in\mathcal{F}:e\in U} y_U - \sum_{e\in T} c_e^-\left(1 - \sum_{U\in\mathcal{F}:e\in U} y_U\right)\right). \tag{3.20}$$

The above expression motivates the substitution

$$p_e := \sum_{U\in\mathcal{F}:e\in U} y_U. \tag{3.21}$$

Let $p = (p_1, \ldots p_n)$. The following is the minmax regret analog of an observation made by Bertsimas et al. [8].

Lemma 3. *For interval uncertainty, the objective value* Z_{R} *of the randomized minmax regret problem* (2.11) *is equal to that of the problem*

$$\min_{p \in \mathrm{CH}(\mathcal{X})} \max_{T \in \mathcal{F}} \left(\sum_{e \in E \setminus T} c_e^+ p_e - \sum_{e \in T} c_e^- (1 - p_e) \right), \tag{3.22}$$

where $\mathrm{CH}(\mathcal{X})$ *denotes the convex hull of* $\mathcal{X}$*.*

Proof. We use the same arguments presented in [8]. The objective value of (3.22) is clearly equal to the objective value of (3.20). Note that p must lie in the convex hull of $\mathcal{X}$. By Carathéodory's Theorem [14], any $p \in \mathrm{CH}(\mathcal{X})$ can be represented by a convex combination of at most $n+1$ points in $\mathcal{X}$, so there exists a surjective mapping from $\mathcal{Y}$ to $\mathrm{CH}(\mathcal{X})$. □

Since we will ultimately use the simplified formulation given in Lemma 3 to solve the randomized minmax regret problem, we address the problem of recovering a vector y given a solution p. In the proof of the lemma, we have used Carathéodory's Theorem, which proves existence of such a mapping, but not its construction. To this end, we define a *mixed strategy encoding* $\mathcal{M} = (X, Y)$ as a set of deterministic solutions $X = \{x^{T_i} \in \mathcal{X} : \ i = 1, \dots, \mu\}$ that should be selected with nonzero probability and the corresponding probabilities $Y = \{y_{T_i} \in [0, 1] : \ i = 1, \dots, \mu\}$ that satisfy $\sum_{i=1}^{\mu} y_{T_i} = 1$. Here μ is the support size of the mixed strategy (i.e. the number of deterministic solutions with nonzero probablity). For a given vector p, we are interested in solving the following constraint satisfaction program:

$$\begin{aligned} \min \quad & 0 \\ \text{s.t.} \quad & \sum_{T \in \mathcal{F}: e \in T} y_T = p_e, \qquad \forall e \in E, \\ & \sum_{T \in \mathcal{F}} y_T = 1, \\ & y \geq \mathbf{0}. \end{aligned} \tag{3.23}$$

Consider the dual program of (3.23), which has variables $u = (u_1, \dots, u_e)$ and w:

$$\max \quad w - \sum_{e \in E} p_e u_e \tag{3.24}$$

$$\text{s.t.} \quad w - \sum_{e \in T} u_e \leq 0, \qquad \forall T \in \mathcal{F}, \tag{3.25}$$

$$u, w \text{ free.}$$

Recall that the region $\mathcal{X}$ is described in size m.

Lemma 4. *For any given* $p \in \mathrm{CH}(\mathcal{X})$*, a corresponding mixed strategy encoding* $\mathcal{M}$ *of size polynomial in* n *can be found via the linear programming formulation* (3.24) - (3.25)*. Furthermore, if the nominal problem* $F^*(c)$ *can be solved in time polynomial in* n *and* m*, then* $\mathcal{M}$ *can be found in time polynomial in* n *and* m*.*

Proof. Notice that while the primal program has an exponential number of variables and a linear number of constraints, the opposite holds true for the dual. The primal program is bounded since all objective coefficients are equal to zero, and is feasible due to Carathéodory's Theorem. Therefore the dual program must be feasible and bounded.

To guarantee a polynomial sized solution, note that the separation problem for the constraints (3.25) is simply the nominal problem with costs u, so the dual program can be solved via the ellipsoid method. If the nominal problem can be solved in polynomial time, then the constraints (3.25) can be generated in polynomial time, giving a polynomial time solution for the entire dual program (3.24) - (3.25).

From a practical perspective, a separation oracle for (3.25) gives an efficient method for performing row generation with the simplex method. Each row i generated while solving the dual problem gives a solution $x^{T_i} \in \mathcal{X}$, and its dual variable is the corresponding probability y_{T_i}. □

Using Lemma 3, we can now formulate a linear program to solve the randomized minmax regret problem.

$$
\begin{aligned}
Z_{\mathrm{R}} = \min\quad & z && (3.26)\\
\text{s.t.}\quad & \sum_{e\in E\setminus T} c_e^+ p_e - \sum_{e\in T} c_e^- (1-p_e) \le z, \qquad \forall T \in \mathcal{F}, && (3.27)\\
& p \in \mathrm{CH}(\mathcal{X}). && (3.28)
\end{aligned}
$$

While the above problem may have an exponential number of constraints, it can be solved efficiently via the ellipsoid algorithm if a separation oracle is available for the constraints (3.27) and (3.28). This brings us to our main result.

Theorem 1. *For interval uncertainty, if the nominal problem $F^*(c)$ can be solved in time polynomial in n and m, then the corresponding randomized minmax regret problem $\min_{y\in\mathcal{Y}} \max_{c\in\mathcal{I}}(\overline{F}(y,c) - F^*(c))$ can be solved in time polynomial in n and m.*

Proof. Consider the linear program (3.26) - (3.28). Note that if we can optimize over $\mathcal{X}$ in polynomial time, then we can separate over $\mathrm{CH}(\mathcal{X})$ in polynomial time via the result of [15]. This gives a separation oracle for the constraint (3.28). To see the separation oracle for the constraint (3.27), we define the item cost vector $d = (d_1, \ldots, d_n)$ where

$$d_e := c_e^- + p_e(c_e^+ - c_e^-), \quad e \in E, \tag{3.29}$$

and then solve

$$z_d = \min_{T\in\mathcal{F}} \sum_{e\in T} d_e. \tag{3.30}$$

Let T_d be the set that minimizes the above expression. If $\sum_{e\in E} c_e^+ p_e - z_d \le z$, then we are guaranteed feasibility, otherwise the separating hyperplane (3.27) is generated where $T = T_d$. To see the validity of this approach, we have

$$
\begin{aligned}
\sum_{e\in E} c_e^+ p_e - z_d &= \sum_{e\in E} c_e^+ p_e - \min_{T\in\mathcal{F}} \sum_{e\in T} d_e\\
&= \sum_{e\in E} c_e^+ p_e - \min_{T\in\mathcal{F}} \sum_{e\in T} (c_e^- + p_e(c_e^+ - c_e^-))\\
&= \max_{T\in\mathcal{F}} \left(\sum_{e\in E} c_e^+ p_e - \sum_{e\in T} (c_e^- + p_e(c_e^+ - c_e^-)) \right)\\
&= \max_{T\in\mathcal{F}} \left(\sum_{e\in E\setminus T} c_e^+ p_e - \sum_{e\in T} c_e^- (1-p_e) \right). \qquad (3.31)
\end{aligned}
$$

The solution to the linear program (3.26) - (3.28) is a vector p, which can then be used to find a mixed strategy y in polynomial time using Lemma 4. $\square$

THEOREM 2. *For interval uncertainty,* $Z_{\mathrm{R}} \leq Z_{\mathrm{D}}$.

Proof. This follows simply by noting that the program (3.26) - (3.28) is the linear programming relaxation of (3.14). $\square$

4 Discrete scenario uncertainty

We now consider the discrete scenario representation of uncertainty. We are given a finite set $\mathcal{S}$ of $|\mathcal{S}| = k$ scenarios, where for each $S \in \mathcal{S}$, there exists a cost vector $c^S = (c_e^S)_{e\in E}$ where for all elements and scenarios, $c_e^S \in \mathbb{R}$. With this representation, the deterministic maximum regret problem is

$$R_{\max}(x) = \max_{S\in\mathcal{S}} R(x, c^S) = \max_{S\in\mathcal{S}} \left(F(x, c^S) - F^*(c^S)\right) \tag{4.32}$$

and the deterministic minmax regret problem is

$$Z_{\mathrm{D}} = \min_{x\in\mathcal{X}} R_{\max}(x) = \min_{x\in\mathcal{X}} \max_{S\in\mathcal{S}} (F(x, c^S) - F^*(c^S)). \tag{4.33}$$

Likewise for the randomized model, the maximum expected regret is

$$\overline{R}_{\max}(y) = \max_{S\in\mathcal{S}} \overline{R}(y, c^S) = \max_{S\in\mathcal{S}} (\overline{F}(y, c^S) - F^*(c^S)) \tag{4.34}$$

and the randomized minmax regret problem is

$$Z_{\mathrm{R}} = \min_{y\in\mathcal{Y}} \overline{R}_{\max}(y) = \min_{y\in\mathcal{Y}} \max_{S\in\mathcal{S}} (\overline{F}(y, c^S) - F^*(c^S)). \tag{4.35}$$

It is easy to see that the deterministic model admits an integer programming solution.

LEMMA 5. *The deterministic minmax regret problem with discrete scenario uncertainty is equivalent to the following integer program.*

$$\begin{aligned} Z_{\mathrm{D}} \;=\; \min \quad & z \\ \text{s.t.} \quad & \sum_{e\in E} c_e x_e - F^*(c^S) \leq z, \qquad \forall S \in \mathcal{S}, \\ & x \in \mathcal{X}. \end{aligned} \tag{4.36}$$

Proof. This follows by the definition of the maximum. $\square$

We now state and prove the main results of this section.

THEOREM 3. *For discrete scenario uncertainty, if the nominal problem $F^*(c)$ can be solved in time polynomial in n and m, then the corresponding randomized minmax regret problem $\min_{y\in\mathcal{Y}} \max_{S\in\mathcal{S}} (\overline{F}(y, c^S) - F^*(c^S))$ can be solved in time polynomial in n and m.*

Proof. Proceeding similarly as we did for the interval representation,

$$\begin{aligned} \overline{R}_{\max}(y) &= \max_{S\in\mathcal{S}} \left(\sum_{T\in\mathcal{F}} y_T \sum_{e\in T} c_e^S - F^*(c^S) \right) \\ &= \max_{S\in\mathcal{S}} \left(\sum_{e\in E} c_e^S \sum_{T\in\mathcal{F}: e\in T} y_T - F^*(c^S) \right). \end{aligned} \tag{4.37}$$

We again use the substitution

$$p_e := \sum_{T \in \mathcal{F}: e \in T} y_T, \tag{4.38}$$

so that we can optimize over the convex hull of $\mathcal{X}$ and recover the vector y using Lemma 4. For the randomized minmax regret problem, we then have the linear program

$$\begin{aligned} \min \quad & z && (4.39)\\ \text{s.t.} \quad & \sum_{e \in E} c_e^S p_e - F^*(c^S) \le z, \qquad \forall S \in \mathcal{S}, && (4.40)\\ & p \in \mathrm{CH}(\mathcal{X}). && (4.41) \end{aligned}$$

Since for all $S \in \mathcal{S}$, the value $F^*(c^S)$ is polynomial solvable, each constraint (4.40) can be enumerated in polynomial time. The separation oracle for the constraints (4.41) is given by the equivalence of optimization and separation [15]. □

THEOREM 4. *For discrete scenario uncertainty,* $Z_{\mathrm{R}} \le Z_{\mathrm{D}}$.

Proof. The program (4.39) - (4.41) is the linear programming relaxation of (4.36). □

5 Conclusion

We have shown that for both the interval and discrete scenario representations of uncertainty, the randomized minmax regret version of any polynomial solvable combinatorial problem is polynomial solvable. Furthermore, the maximum expected regret in the randomized model is upper bounded by the maximum regret of the deterministic model. These results, including the fact that there always exists a polynomial-sized optimal solution for randomized minmax regret, are at first glance somewhat surprising. Intuitively, the polynomial solvability of the randomized model results from the fact that a linear program must be solved instead of the integer program or mixed integer program (which is required for the deterministic model). The improvement in performance holds because the adversary has less power in the randomized model than the deterministic model.

For many applications that are not adversarial in nature, the randomized minmax regret criteria is arguably a more appropriate model than the deterministic version. In particular, the deterministic solution may be overly conservative since costs are not truly chosen in an adversarial fashion in response to the selected solution. On the other hand, one must be willing to tolerate higher variance if randomization is used.

An important future step with this research is to develop approximation algorithms for dealing with nominal problems that are already NP-complete. This problem is non-trivial: an algorithm with an approximation factor α for a nominal problem does not immediately yield an algorithm to approximate the randomized minmax regret problem with a factor α. Another interesting topic to study from an experimental perspective is a hybrid approach that employs both deterministic and randomized minmax regret. For example, one could find a solution that minimizes the maximum expected regret, subject to the maximum regret being no greater than some constant M. The algorithm for this problem can be easily constructed by combining our results with existing work, but may no longer be polynomial solvable.

References

[1] H. Aissi, C. Bazgan, and D. Vanderpooten, *Complexity of the min–max and min–max regret assignment problems*, Operations Research Letters, 33 (2005), pp. 634–640.

[2] ———, *Complexity of the min-max (regret) versions of cut problems*, in Algorithms and Computation, Springer, 2005, pp. 789–798.

[3] ———, *Min–max and min–max regret versions of combinatorial optimization problems: A survey*, European Journal of Operational Research, 197 (2009), pp. 427–438.

[4] I. Averbakh, *On the complexity of a class of combinatorial optimization problems with uncertainty*, Mathematical Programming, 90 (2001), pp. 263–272.

[5] I. Averbakh and V. Lebedev, *Interval data minmax regret network optimization problems*, Discrete Applied Mathematics, 138 (2004), pp. 289–301.

[6] ———, *On the complexity of minmax regret linear programming*, European Journal of Operational Research, 160 (2005), pp. 227–231.

[7] S. Ben-David, A. Borodin, R. Karp, G. Tardos, and A. Wigderson, *On the power of randomization in on-line algorithms*, Algorithmica, 11 (1994), pp. 2–14.

[8] D. Bertsimas, E. Nasrabadi, and J. B. Orlin, *On the power of nature in robust discrete optimization*, In preparation, (2012).

[9] ———, *On the power of randomization in network interdiction*, In preparation, (2012).

[10] D. Bertsimas and M. Sim, *The price of robustness*, Operations research, 52 (2004), pp. 35–53.

[11] ———, *Robust discrete optimization under ellipsoidal uncertainty sets*, Technical report, Massachusetts Institute of Technology, (2004).

[12] J. R. Birge and F. V. Louveaux, *Introduction to stochastic programming*, Springer, 1997.

[13] A. Borodin and R. El-Yaniv, *Online computation and competitive analysis*, vol. 53, Cambridge University Press Cambridge, 1998.

[14] C. Carathéodory, *Über den variabilitätsbereich der fourierschen konstanten von positiven harmonischen funktionen*, Rendiconti del Circolo Matematico di Palermo (1884-1940), 32 (1911), pp. 193–217.

[15] M. Grötschel, L. Lovász, and A. Schrijver, *The ellipsoid method and its consequences in combinatorial optimization*, Combinatorica, 1 (1981), pp. 169–197.

[16] B. Johannes and J. B. Orlin, *Minimax regret problems are harder than minimax problems*, Submitted, (2012).

[17] A. Kasperski, *Discrete optimization with interval data: minmax regret and fuzzy approach*, vol. 228, Springer, 2008.

[18] D. Korzhyk, V. Conitzer, and R. Parr, *Complexity of computing optimal stackelberg strategies in security resource allocation games.*, in AAAI, 2010.

[19] P. Kouvelis and G. Yu, *Robust discrete optimization and its applications*, vol. 14, Springer, 1997.

[20] N. Nisan, *Algorithmic game theory*, Cambridge University Press, 2007.

[21] L. J. Savage, *The theory of statistical decision*, Journal of the American Statistical association, 46 (1951), pp. 55–67.

[22] A. Shapiro, D. Dentcheva, and A. P. Ruszczyński, *Lectures on stochastic programming: modeling and theory*, vol. 9, SIAM, 2009.

[23] A. Wald, *Contributions to the theory of statistical estimation and testing hypotheses*, The Annals of Mathematical Statistics, 10 (1939), pp. 299–326.

[24] G. Yu and J. Yang, *On the robust shortest path problem*, Computers & Operations Research, 25 (1998), pp. 457–468.