Thresholded Covering Algorithms for Robust and Max-min Optimization

Anupam Gupta[1,⋆], Viswanath Nagarajan[2], and R. Ravi[3,⋆⋆]

[1] Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA
[2] IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA
[3] Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Abstract. The general problem of robust optimization is this: one of several possible scenarios will appear tomorrow and require coverage, but things are more expensive tomorrow than they are today. What should you anticipatorily buy today, so that the worst-case covering cost (summed over both days) is minimized? We consider the *k-robust model* [6,15] where the possible scenarios tomorrow are given by all demand-subsets of size k.

We present a simple and intuitive template for k-robust problems. This gives improved approximation algorithms for the k-robust Steiner tree and set cover problems, and the first approximation algorithms for k-robust Steiner forest, minimum-cut and multicut. As a by-product of our techniques, we also get approximation algorithms for *k-max-min* problems of the form: "*given a covering problem instance, which k of the elements are costliest to cover?*"

1 Introduction

Consider the following *k-robust* set cover problem: we are given a set system $(U, \mathcal{F} \subseteq 2^U)$. Tomorrow some set of k elements $S \subseteq U$ will want to be covered; however, today we don't know what this set will be. One strategy is to wait until tomorrow and buy an $O(\log n)$-approximate set cover for this set. However, sets are cheaper today: they will cost λ times as much tomorrow as they cost today. Hence, it may make sense to buy some anticipatory partial solution today (i.e. in the first-stage), and then complete it tomorrow (i.e. second-stage) once we know the actual members of the set S. Since we do not know anything about the set S (or maybe we are risk-averse), we want to plan for the worst-case, and minimize

$$\text{(cost of anticipatory solution)} + \lambda \cdot \max_{S:|S|\leq k} \text{(additional cost to cover } S\text{)}.$$

Early approximation results for robust problems [4,10] had assumed that the collection of possible sets S was explicitly given (and the performance guarantee

⋆ Supported in part by NSF awards CCF-0448095 and CCF-0729022, and an Alfred P. Sloan Fellowship.
⋆⋆ Supported in part by NSF grant CCF-0728841.

S. Abramsky et al. (Eds.): ICALP 2010, Part I, LNCS 6198, pp. 262–274, 2010.

depended logarithmically on the size of this collection). Since this seemed quite restrictive, Feige et al. [6] proposed this k-robust model where any of the $\binom{n}{k}$ subsets S of size k could arrive. Though this collection of possible sets was potentially exponential sized (for large values of k), the hope was to get results that did not depend polynomially on k. To get such an algorithm, Feige et al. considered the k-max-min set-cover problem ("which subset $S \subseteq U$ of size k requires the largest set cover value?"), and used the online algorithm for set cover to get a greedy-style $O(\log m \log n)$ approximation for this problem; here m and n are the number of sets and elements in the set system. They then used this max-min problem as a separation oracle in an LP-rounding-based algorithm (à la [20]) to get the same approximation guarantee for the k-robust problem. They also showed the max-min and k-robust set cover problems to be $\Omega(\frac{\log m}{\log \log m} + \log n)$ hard—which left a logarithmic gap between the upper and lower bounds. However, an online algorithm based approach is unlikely to close this gap, since the online algorithm for set cover is necessarily a log-factor worse that its offline counterparts [2].

Apart from the obvious goal of improving the result for this particular problem, one may want to develop algorithms for other k-robust problems. E.g., for the *k-robust min-cut* problem, some set S of k sources will want to be separated from the sink vertex tomorrow, and we want to find the best way to cut edges to minimize the total cost incurred (over the two days) for the worst-case k-set S. Similarly, in the *k-robust Steiner forest*, we are given a metric space and a collection of source-sink pairs, and any set S of k source-sink pairs may desire pairwise connection tomorrow; what should we do to minimize the sum of the costs incurred over the two days? One can extend the Feige et al. framework to first solve max-min problems using online algorithms, but in all cases we seem to lose extra logarithmic factors. Moreover, for the above two problems (and others), the LP-rounding-based framework does not seem to extend directly. The latter obstacle was also observed by Khandekar et al. [15], who gave constant-factor algorithms for k-robust versions of Steiner tree and facility location.

1.1 Main Results

In this paper, we present a general template to design algorithms for k-robust problems. We improve on previous results, by obtaining an $O(\log m + \log n)$ factor for k-robust set cover, and improving the constant in the approximation factor for Steiner tree. We also give the first algorithms for some other standard covering problems, getting constant-factor approximations for both k-robust Steiner forest—which was left open by Khandekar et al.—and for k-robust min-cut, and an $O(\frac{\log^2 n}{\log \log n})$ approximation for k-robust multicut. Our algorithms do not use a max-min subroutine directly: however, our approach ends up giving us approximation algorithms for k-max-min versions of set cover, Steiner forest, min-cut and multicut; all but the one for multicut are best possible under standard assumptions; the ones for Steiner forest and multicut are the first known algorithms for their k-max-min versions.

An important contribution of our work (even more than the new/improved approximation guarantees) is the simplicity of the algorithms, and the ideas in their analysis. The following is our actual algorithm for k-robust set cover.

> Suppose we "guess" that the maximum second-stage cost in the optimal solution is T. Let $A \subseteq U$ be all elements for which the cheapest set covering them costs more than $\beta \cdot T/k$, where $\beta = O(\log m + \log n)$. We build a set cover on A as our first stage. (Say this cover costs C_T.)

To remove the guessing, try all values of T and choose the solution that incurs the least total cost $C_T + \lambda\beta T$. Clearly, by design, no matter which k elements arrive tomorrow, it will not cost us more than $\lambda \cdot k \cdot \beta T/k = \lambda\beta T$ to cover them, which is within β of what the optimal solution pays.

The key step of our analysis is to argue why C_T is close to optimum. We briefly describe the intuition; details appear in Section 3. Suppose $C_T \gg \beta\mathsf{Opt}$: then the fractional solution to the LP for set cover for A would cost $\gg \frac{\beta}{\ln n}\mathsf{Opt} \gg \mathsf{Opt}$, and so would its dual. Our key technical contribution is to show how to *"round" this dual LP* to find a "witness" $A' \subseteq A$ with only k elements, and also a corresponding feasible dual of value $\gg \mathsf{Opt}$—i.e., the dual value does not decrease much in the rounding (This rounding uses the fact that the algorithm only put those elements in A that were expensive to cover). Using duality again, this proves that the optimal LP value, and hence the optimal set cover for these k elements A', would cost much more than Opt—a contradiction!

In fact, our algorithms for the other k-robust problems are almost identical to this one; indeed, the only slightly involved algorithm is that for k-robust Steiner forest. Of course, the proofs to bound the cost C_T need different ideas in each case. For example, directly rounding the dual for Steiner forest is difficult, so we give a primal-dual argument to show the existence of such a witness $A' \subseteq A$ of size at most k. For the cut-problems, one has to deal with additional issues because Opt consists of two stages that have to be charged to separately, and this requires a Gomory-Hu-tree-based charging. Even after this, we still have to show that if the cut for a set of sources A is large then there is a witness $A' \subseteq A$ of at most k sources for which the cut is also large—i.e., we have to somehow aggregate the flows (i.e. dual-rounding for cut problems). We prove new flow-aggregation lemmas for single-sink flows using Steiner-tree-packing results, and for multiflows using oblivious routing [18]; both proofs are possibly of independent interest.

Paper Outline. In Sections 2 and 2.1 we present the formal framework for k-robust problems, and abstract out the properties that we'd like from our algorithms. Section 3 contains such an algorithm for k-robust set cover, and k-robust min-cut appears in Section 4. We refer the reader to the full version of the paper [13] for missing proofs and results on k-robust Steiner forest and multicut. There we also present a general reduction from robust problems to the corresponding max-min problems, which has implications for uncertainty sets specified by matroid and knapsack type constraints.

1.2 Related Work

Approximation algorithms for robust optimization was initiated by Dhamdhere et al. [4]: they study the case when the scenarios were explicitly listed, and gave constant-factor approximations for Steiner tree and facility location, and logarithmic approximations to mincut/multicut problems. Golovin et al. [10] improved the mincut result to a constant factor approximation, and also gave an $O(1)$-approximation for robust shortest-paths. The k-robust model was introduced in Feige et al. [6], where they gave an $O(\log m \log n)$-approximation for set cover. Khandekar et al. [15] noted that the techniques of [6] did not give good results for Steiner tree, and developed new constant-factor approximations for k-robust versions of Steiner tree, Steiner forest on trees and facility location. Using our framework, the algorithm we get for Steiner tree can be viewed as a rephrasing of their algorithm—our proof is arguably more transparent and results in a better bound. Our approach can also be used to get a slightly better ratio than [15] for the Steiner forest problem on trees.

Constrained submodular maximization problems [17,7,23,3,25] appear very relevant at first sight: e.g., the k-max-min version of min-cut ("find the k sources whose separation from the sink costs the most") is precisely submodular maximization under a cardinality constraint, and hence is approximable to within $(1-1/e)$. But apart from min-cut, the other problems do not give us submodular functions to maximize, and massaging the functions to make them submodular seems to lose logarithmic factors. E.g., one can use tree embeddings [5] to reduce Steiner tree to a problem on trees and make it submodular; in other cases, one can use online algorithms to get submodular-like properties (we give a general reduction for covering problems that admit good offline and online algorithms in [13]). Eventually, it is unclear how to use existing results on submodular maximization in any general way.

Considering the *average* instead of the worst-case performance gives rise to the well-studied model of stochastic optimization [19,14]. Some common generalizations of the robust and stochastic models have been considered (see, e.g., Swamy [24] and Agrawal et al. [1]).

Feige et al. [6] also considered the k-max-min set cover—they gave an $O(\log m \log n)$-approximation algorithm for this, and used it in the algorithm for k-robust set cover. They also showed an $\Omega(\frac{\log m}{\log\log m})$ hardness of approximation for k-max-min (and k-robust) set cover. To the best of our knowledge, none of the k-max-min problems other than min-cut have been studied earlier.

The *k-min-min* versions of covering problems (i.e. "which k demands are the *cheapest* to cover?") have been extensively studied for set cover [21,8], Steiner tree [9], Steiner forest [12], min-cut and multicut [11,18]. However these problems seem to be related to the k-max-min versions only in spirit.

2 Notation and Definitions

Deterministic covering problems. A covering problem Π has a ground-set E of elements with costs $c : E \to \mathbb{R}_+$, and n covering requirements (often called demands or clients), where the solutions to the i-th requirement is specified—possibly implicitly—by a family $\mathcal{R}_i \subseteq 2^E$ which is upwards closed (since this is a covering problem). Requirement i is *satisfied* by solution $S \subseteq E$ iff $S \in \mathcal{R}_i$. The covering problem $\Pi = \langle E, c, \{\mathcal{R}_i\}_{i=1}^n \rangle$ involves computing a solution $S \subseteq E$ satisfying all n requirements and having minimum cost $\sum_{e \in S} c_e$. E.g., in set cover, "requirements" are items to be covered, and "elements" are sets to cover them with. In Steiner tree, requirements are terminals to connect to the root and elements are the edges; in multicut, requirements are terminal pairs to be separated, and elements are edges to be cut.

Robust covering problems. This problem, denoted $\mathsf{Robust}(\Pi)$, is a *two-stage optimization* problem, where elements are possibly bought in the first stage (at the given cost) or the second stage (at cost λ times higher). In the second stage, some subset $\omega \subseteq [n]$ of requirements (also called a *scenario*) materializes, and the elements bought in both stages must satisfy each requirement in ω. Formally, the input to problem $\mathsf{Robust}(\Pi)$ consists of (a) the covering problem $\Pi = \langle E, c, \{\mathcal{R}_i\}_{i=1}^n \rangle$ as above, (b) a set $\Omega \subseteq 2^{[n]}$ of scenarios (possibly implicitly given), and (c) an inflation parameter $\lambda \geq 1$. Since we deal with covering problems, it can be assumed WLOG that the uncertainty set Ω is downwards closed. A feasible solution to $\mathsf{Robust}(\Pi)$ is a set of *first stage elements* $E_0 \subseteq E$ (bought without knowledge of the scenario), along with an *augmentation algorithm* that given any $\omega \in \Omega$ outputs $E_\omega \subseteq E$ such that $E_0 \cup E_\omega$ satisfies all requirements in ω. The objective function is to minimize: $c(E_0) + \lambda \cdot \max_{\omega \in \Omega} c(E_\omega)$. Given such a solution, $c(E_0)$ is called the first-stage cost and $\max_{\omega \in \Omega} c(E_\omega)$ is the second-stage cost.

k-robust problems. In this paper, we deal with robust covering problems under *cardinality* uncertainty sets: i.e., $\Omega := \binom{[n]}{k} = \{S \subseteq [n] \mid |S| = k\}$. We denote this problem by $\mathsf{Robust}_k(\Pi)$.

Max-min problems. Given a covering problem Π and a set Ω of scenarios, the *max-min* problem involves finding a scenario $\omega \in \Omega$ for which the cost of the min-cost solution to ω is maximized. Note that by setting $\lambda = 1$ in any robust covering problem, *the optimal value of the robust problem equals that of its corresponding max-min problem.* In a **k-max-min problem** we have $\Omega = \binom{[n]}{k}$.

2.1 The Abstract Properties We Want from Our Algorithms

Our algorithms for robust and max-min versions of covering problems are based on the following guarantee.

Definition 1. *An algorithm is* $(\alpha_1, \alpha_2, \beta)$-discriminating *iff given as input any instance of* $\mathsf{Robust}_k(\Pi)$ *and a threshold* T*, the algorithm outputs (i) a set* $\Phi_T \subseteq E$*, and (ii) an algorithm* $\mathsf{Augment}_T : \binom{[n]}{k} \to 2^E$*, such that:*

A. For every scenario $D \in \binom{[n]}{k}$*,*

(i) the elements in $\Phi_T \cup \mathsf{Augment}_T(D)$ *satisfy all requirements in* D*, and*

(ii) the resulting augmentation cost $c\left(\mathsf{Augment}_T(D)\right) \leq \beta \cdot T$*.*

B. Let Φ^* and T^* (respectively) denote the first-stage and second-stage cost of an optimal solution to the $\mathsf{Robust}_k(\Pi)$ *instance. If the threshold $T \geq T^*$ then the first stage cost $c(\Phi_T) \leq \alpha_1 \cdot \Phi^* + \alpha_2 \cdot T^*$.*

The next lemma shows why having a discriminating algorithm is sufficient to solve the robust problem. The issue to address is that having guessed T for the optimal second stage cost, we have no direct way of verifying the correctness of that guess—hence we choose the best among all possible values of T. For $T \approx T^*$ the guarantees in Definition 1 ensure that we pay $\approx \Phi^* + T^*$ in the first stage, and $\approx \lambda T^*$ in the second stage; for guesses $T \ll T^*$, the first-stage cost in guarantee (B) is likely to be large compared to Opt.

Lemma 1. *If there is an $(\alpha_1, \alpha_2, \beta)$-discriminating algorithm for a robust covering problem* $\mathsf{Robust}_k(\Pi)$*, then for every $\epsilon > 0$ there is a $\left((1+\epsilon) \cdot \max\left\{\alpha_1, \beta + \frac{\alpha_2}{\lambda}\right\}\right)$-approximation algorithm for* $\mathsf{Robust}_k(\Pi)$*.*

In the rest of the paper, we focus on providing discriminating algorithms for suitable values of $\alpha_1, \alpha_2, \beta$.

2.2 Additional Property Needed for *k*-Max-min Approximations

As we noted above, a k-max-min problem is a robust problem where the inflation $\lambda = 1$ (which implies that in an optimal solution $\Phi^* = 0$, and T^* is the k-max-min value). Hence a discriminating algorithm immediately gives an approximation to the *value*: for any $D \in \binom{[n]}{k}$, $\Phi_T \cup \mathsf{Augment}_T(D)$ satisfies all demands in D, and for the right guess of $T \approx T^*$, the cost is at most $(\alpha_2 + \beta)T^*$. It remains to output a bad k-set as well, and hence the following definition is useful.

Definition 2. *An algorithm for a robust problem is* strongly discriminating *if it satisfies the properties in Definition 1, and when the inflation parameter is $\lambda = 1$ (and hence $\Phi^* = 0$), the algorithm also outputs a set $Q_T \in \binom{[n]}{k}$ such that if $c(\Phi_T) \geq \alpha_2 T$, the cost of optimally covering the set Q_T is $\geq T$.*

Recall that for a covering problem Π, the cost of optimally covering the set of requirements $Q \in \binom{[n]}{k}$ is $\min\{c(E_Q) \mid E_Q \subseteq E \text{ and } E_Q \in \mathcal{R}_i \ \forall i \in Q\}$.

Lemma 2. *If there is an $(\alpha_1, \alpha_2, \beta)$-*strongly*-discriminating algorithm for a robust covering problem* $\mathsf{Robust}_k(\Pi)$*, then for every $\epsilon > 0$ there is an algorithm for k-max-min(Π) that outputs a set Q such that for some T, the optimal cost of covering this set Q is at least T, but every k-set can be covered with cost at most $(1+\epsilon) \cdot (\alpha_2 + \beta)\, T$.*

3 Set Cover

Consider the k-robust set cover problem where there is a set system $(U, \mathcal{F})$ with a universe of $|U| = n$ elements, and m sets in $\mathcal{F}$ with each set $R \in \mathcal{F}$ costing c_R, an inflation parameter λ, and an integer k such that each of the sets $\binom{U}{k}$ is a possible scenario for the second-stage. Given Lemma 1, it suffices to

Algorithm 1. Algorithm for k-Robust Set Cover

1: **input:** k-robust set-cover instance and threshold T.
2: **let** $\beta \leftarrow 36 \ln m$, and $S \leftarrow \{v \in U \mid$ min cost set covering v has cost at least $\beta \cdot \frac{T}{k}\}$.
3: **output** first stage solution Φ_T as the Greedy-Set-Cover(S).
4: **define** $\mathsf{Augment}_T(\{i\})$ as the min-cost set covering i, for $i \in U \setminus S$; and $\mathsf{Augment}_T(\{i\}) = \emptyset$ for $i \in S$.
5: **output** second stage solution $\mathsf{Augment}_T$, where $\mathsf{Augment}_T(D) := \bigcup_{i \in D} \mathsf{Augment}_T(\{i\})$ for all $D \subseteq U$.

show a discriminating algorithm as defined in Definition 1 for this problem. The algorithm given below is easy: pick all elements which can only be covered by expensive sets, and cover them in the first stage.

Claim 1 (Property A for Set Cover). *For all $T \geq 0$ and scenario $D \in \binom{U}{k}$, the sets $\Phi_T \bigcup \mathsf{Augment}_T(D)$ cover elements in D, and $c(\mathsf{Augment}_T(D)) \leq \beta T$.*

Proof: The elements in $D \cap S$ are covered by Φ_T; and by definition of $\mathsf{Augment}_T$, each element $i \in D \setminus S$ is covered by set $\mathsf{Augment}_T(\{i\})$. Thus we have the first part of the claim. For the second part, note that by definition of S, the cost of $\mathsf{Augment}_T(\{i\})$ is at most $\beta T/k$ for all $i \in U$. ■

Theorem 1 (Property B for Set Cover). *Let Φ^* denote the optimal first stage solution (and its cost), and T^* the optimal second stage cost. Let $\beta = 36 \ln m$. If $T \geq T^*$ then $c(\Phi_T) \leq H_n \cdot (\Phi^* + 12 \cdot T^*)$.*

Proof: We claim that there is a *fractional* solution $\bar{x}$ for the set covering instance S with small cost $O(\Phi^* + T^*)$, whence rounding this to an integer solution implies the theorem, since the greedy algorithm has performance ratio H_n. For a contradiction, assume not: let every fractional set cover be expensive, and hence there must be a dual solution of large value. We then *round this dual solution* to get a dual solution to a sub-instance with only k elements that costs $> \Phi^* + T^*$, which is impossible (since using the optimal solution we can solve every instance on k elements with that cost).

To this end, let $S' \subseteq S$ denote the elements that are *not* covered by the optimal first stage Φ^*, and let $\mathcal{F}' \subseteq \mathcal{F}$ denote the sets that contain at least one element from S'. By the choice of S, all sets in $\mathcal{F}'$ cost at least $\beta \cdot \frac{T}{k} \geq \beta \cdot \frac{T^*}{k}$. Define the "coarse" cost for a set $R \in \mathcal{F}'$ to be $\widehat{c}_R = \lceil \frac{c_R}{6T^*/k} \rceil$. For each set $R \in \mathcal{F}'$, since $c_R \geq \frac{\beta T^*}{k} \geq \frac{6T^*}{k}$, it follows that $\widehat{c}_R \cdot \frac{6T^*}{k} \in [c_R, 2 \cdot c_R)$, and also that $\widehat{c}_R \geq \beta/6$.

Now consider the following primal-dual pair of LPs for the set cover instance with elements S' and sets $\mathcal{F}'$ having the coarse costs $\widehat{c}$.

$$
\begin{array}{llll}
\min \sum_{R \in \mathcal{F}'} \widehat{c}_R \cdot x_R & & \max \sum_{e \in S'} y_e & \\
\sum_{R \ni e} x_R \geq 1, & \forall e \in S', & \sum_{e \in R} y_e \leq \widehat{c}_R, & \forall R \in \mathcal{F}', \\
x_R \geq 0, & \forall R \in \mathcal{F}'. & y_e \geq 0, & \forall e \in S'.
\end{array}
$$

Let $\{x_R\}_{R \in \mathcal{F}'}$ be an optimal primal and $\{y_e\}_{e \in S'}$ an optimal dual solution. The following claim bounds the (coarse) cost of these fractional solutions.

Claim 2. *If $\beta = 36 \ln m$, then the LP cost is $\sum_{R \in \mathcal{F}'} \widehat{c}_R \cdot x_R = \sum_{e \in S'} y_e \leq 2 \cdot k$.*

Before we prove Claim 2, let us assume it and complete the proof of Theorem 1. Given the primal LP solution $\{x_R\}_{R \in \mathcal{F}'}$ to cover elements in S', define an LP solution to cover elements in S as follows: define $z_R = 1$ if $R \in \Phi^*$, $z_R = x_R$ if $R \in \mathcal{F}' \setminus \Phi^*$; and $z_R = 0$ otherwise. Since the solution $\bar{z}$ contains Φ^* integrally, it covers elements $S \setminus S'$ (i.e. the portion of S covered by Φ^*); since $z_R \geq x_R$, $\bar{z}$ fractionally covers S'. Finally, the cost of this solution is $\sum_R c_R z_R \leq \Phi^* + \sum_R c_R x_R \leq \Phi^* + \frac{6T^*}{k} \cdot \sum_R \widehat{c}_R x_R$. But Claim 2 bounds this by $\Phi^* + 12 \cdot T^*$. Since we have a LP solution of value $\Phi^* + 12T^*$, and the greedy algorithm is an H_n-approximation relative to the LP value for set cover, this completes the proof. ■

Claim 1 and Theorem 1 show our algorithm for set cover to be an $(H_n, 12H_n, 36 \ln m)$-discriminating algorithm. Applying Lemma 1 converts this discriminating algorithm to an algorithm for k-robust set cover, and gives the following improvement to the result of [6].

Theorem 2. *There is an $O(\log m + \log n)$-approximation for k-robust set cover.*

It remains to give the proof for Claim 2 above; indeed, that is where the technical heart of the result lies.

Proof of Claim 2: Recall that we want to bound the optimal fractional set cover cost for the instance $(S', \mathcal{F}')$ with the coarse (integer) costs; x_R and y_e are the optimal primal and dual solutions. For a contradiction, assume that the LP cost $\sum_{R \in \mathcal{F}'} \widehat{c}_R x_R = \sum_{e \in S'} y_e$ lies in the unit interval $((\gamma - 1)k, \gamma k]$ for some integer $\gamma \geq 3$.

Define integer-valued random variables $\{Y_e\}_{e \in S'}$ by setting, for each $e \in S'$ independently, $Y_e = \lfloor y_e \rfloor + I_e$, where I_e is a Bernoulli$(y_e - \lfloor y_e \rfloor)$ random variable. We claim that whp the random variables $Y_e/3$ form a feasible dual— i.e., they satisfy all the constraints $\{\sum_{e \in R}(Y_e/3) \leq \widehat{c}_R\}_{R \in \mathcal{F}'}$ with high probability. Indeed, consider a dual constraint corresponding to $R \in \mathcal{F}'$: since we have $\sum_{e \in R} \lfloor y_e \rfloor \leq \widehat{c}_R$, we get that $Pr[\sum_{e \in R} Y_e > 3 \cdot \widehat{c}_R] \leq Pr[\sum_{e \in R} I_e > 2 \cdot \widehat{c}_R]$. But now we use a Chernoff bound [16] to bound the probability that the sum of independent 0-1 r.v.s, $\sum_{e \in R} I_e$, exceeds twice its mean (here $\sum_{e \in R} E[I_e] \leq \sum_{e \in R} y_e \leq \widehat{c}_R$) by $\varepsilon^{-\widehat{c}_R/3} \leq e^{-\beta/18} \leq m^{-2}$, since each $\widehat{c}_R \geq \beta/6$ and $\beta = 36 \cdot \ln m$. Finally, a trivial union bound implies that $Y_e/3$ satisfies all the m contraints with probability at least $1 - 1/m$. Moreover, the expected dual objective is $\sum_{e \in S'} y_e \geq (\gamma - 1)k \geq 1$ (since $\gamma \geq 3$ and $k \geq 1$), and by another Chernoff Bound, $\Pr[\sum_{e \in S'} Y_e > \frac{\gamma - 1}{2} \cdot k] \geq a$ (where $a > 0$ is some constant). Putting it all together, with probability at least $a - \frac{1}{m}$, we have a *feasible* dual solution $Y'_e := Y_e/3$ with objective value at least $\frac{\gamma - 1}{6} \cdot k$.

Why is this dual Y'_e any better than the original dual y_e? It is "near-integral"— specifically, each Y'_e is either zero or at least $\frac{1}{3}$. So order the elements of S' in decreasing order of their Y'-value, and let Q be the set of the *first k elements* in this order. The total dual value of elements in Q is at least $\min\{\frac{\gamma - 1}{6}k, \frac{k}{3}\} \geq \frac{k}{3}$, since $\gamma \geq 3$, and each non-zero Y' value is $\geq 1/3$. This valid dual for elements in Q shows a lower bound of $\frac{k}{3}$ on minimum (fractional) $\widehat{c}$-cost to cover the k elements in Q. Using $c_R > \frac{3T^*}{k} \cdot \widehat{c}_R$ for each $R \in \mathcal{F}'$, the minimum c-cost to fractionally cover Q is $> \frac{3T^*}{k} \cdot \frac{k}{3} = T^*$. Hence, if Q is the realized scenario, the optimal second stage cost will be $> T^*$ (as no element in Q is covered by Φ^*)—this contradicts the fact that OPT can cover $Q \in \binom{U}{k}$ with cost at most T^*. Thus we must have $\gamma \leq 2$, which completes the proof of Claim 2. ■

The k-Max-Min Set Cover Problem. The proof of Claim 2 suggests how to get a $(H_n, 12H_n, 36\ln m)$ strongly discriminating algorithm. When $\lambda = 1$ (and so $\Phi^* = 0$), the proof shows that if $c(\Phi_T) > 12H_n \cdot T$, there is a randomized algorithm that outputs k-set Q with optimal covering cost $> T$ (witnessed by the dual solution having cost $> T$). Now using Lemma 2, we get the claimed $O(\log m + \log n)$ algorithm for the k-max-min set cover problem. This nearly matches the hardness of $\Omega(\frac{\log m}{\log\log m} + \log n)$ given by [6].

Remarks: We note that our k-robust algorithm also extends to the more general setting of (uncapacitated) *Covering Integer Programs*; a CIP (see eg. [22]) is given by $A \in [0,1]^{n\times m}$, $b \in [1,\infty)^n$ and $c \in \mathbb{R}_+^m$, and the goal is to minimize $\{c^T \cdot x \mid Ax \geq b, x \in \mathbb{Z}_+^m\}$. The results above (as well as the [6] result) also hold in the presence of set-dependent inflation factors—details will appear in the full version. Results for the other covering problems do not extend to the case of non-uniform inflation: this is usually inherent, and not just a flaw in our analysis. Eg., [15] give an $\Omega(\log^{1/2-\epsilon} n)$ hardness for k-robust Steiner forest under just two distinct inflation-factors, whereas we give an $O(1)$-approximation under uniform inflations.

4 Minimum Cut

We now consider the k-robust minimum cut problem, where we are given an undirected graph $G = (V, E)$ with edge capacities $c : E \to \mathbb{R}_+$, a root $r \in V$, terminals $U \subseteq V$, inflation factor λ. Again, any subset in $\binom{U}{k}$ is a possible second-stage scenario, and again we seek to give a discriminating algorithm. This algorithm, like for set cover, is non-adaptive: we just pick all the "expensive" terminals and cut them in the first stage.

Algorithm 2. Algorithm for k-Robust Min-Cut

1: **input:** k-robust minimum-cut instance and threshold T.
2: **let** $\beta \leftarrow \Theta(1)$, and
$S \leftarrow \{v \in U \mid$ min cut separating v from root r costs at least $\beta \cdot \frac{T}{k}\}$.
3: **output** first stage solution Φ_T as the minimum cut separating S from r.
4: **define** $\mathsf{Augment}_T(\{i\})$ as the min-r-i cut in $G \setminus \Phi_T$, for $i \in U \setminus S$; and $\mathsf{Augment}_T(\{i\}) = \emptyset$ for $i \in S$.
5: **output** second stage solution $\mathsf{Augment}_T$, where
$\mathsf{Augment}_T(D) := \bigcup_{i \in D} \mathsf{Augment}_T(\{i\})$ for all $D \subseteq U$.

Claim 3 (Property A for Min-Cut). *For all $T \geq 0$ and $D \in \binom{U}{k}$, the edges $\Phi_T \bigcup \mathsf{Augment}_T(D)$ separate terminals D from r; and $c(\mathsf{Augment}_T(D)) \leq \beta T$.*

Theorem 3 (Property B for Min-Cut). *Let Φ^* denote the optimal first stage solution (and its cost), and T^* the optimal second stage cost. If $\beta \geq \frac{10e}{e-1}$ and $T \geq T^*$ then $c(\Phi_T) \leq 3 \cdot \Phi^* + \frac{\beta}{2} \cdot T^*$.*

Here's the intuition for this theorem: As in the set cover proof, we claim that if the optimal cost of separating S from the root r is high, then there must be a dual solution (which prescribes flows from vertices in S to r) of large value. We again "round" this dual solution by aggregating these flows to get a set of k terminals that have a large combined flow (of value $> \Phi^* + T^*$) to the root—but this is impossible, since the optimal solution promises us a cut of at most $\Phi^* + T^*$ for any set of k terminals.

However, more work is required. For set-cover, each element was either covered by the first-stage, or it was not; for cut problems, things are not so cut-and-dried, since both stages may help in severing a terminal from the root! So we divide S into two parts differently: the first part contains those nodes whose min-cut in G is large (since they belonged to S) but it fell by a constant factor in the graph $G \setminus \Phi^*$. These we call "low" nodes, and we use a Gomory-Hu tree based analysis to show that all low nodes can be completely separated from r by paying only $O(\Phi^*)$ more (this we show in Claim 4). The remaining "high" nodes continue to have a large min-cut in $G \setminus \Phi^*$, and for these we use the dual rounding idea sketched above to show a min-cut of $O(T^*)$ (this is proved in Claim 5). Together these claims imply Theorem 3.

To begin the proof of Theorem 3, let $H := G \setminus \Phi^*$, and let $S_h \subseteq S$ denote the "high" vertices whose min-cut from the root in H is at least $M := \frac{\beta}{2} \cdot \frac{T^*}{k}$. The following claim is essentially from Golovin et al. [10].

Claim 4 (Cutting Low Nodes). *If $T \geq T^*$, the minimum cut in H separating $S \setminus S_h$ from r costs at most $2 \cdot \Phi^*$.*

Claim 5 (Cutting High Nodes). *If $T \geq T^*$, the minimum r-S_h cut in H costs at most $\frac{\beta}{2} \cdot T^*$, when $\beta \geq \frac{10 \cdot e}{e-1}$.*

Proof: Consider a r-S_h max-flow in the graph $H = G \setminus \Phi^*$, and suppose it sends $\alpha_i \cdot M$ flow to vertex $i \in S_h$. By making copies of terminals, we can assume each $\alpha_i \in (0,1]$; the k-robust min-cut problem remains unchanged under making copies. Hence if we show that $\sum_{i \in S_h} \alpha_i \leq k$, the total flow (which equals the min r-S_h cut) would be at most $k \cdot M = \frac{\beta}{2} \cdot T^*$, which would prove the claim. For a contradiction, we suppose that $\sum_{i \in S_h} \alpha_i > k$. We will now claim that there exists a subset $W \subseteq S_h$ with $|W| \leq k$ such that the min r-W cut is more than T^*, contradicting the fact that every k-set in H can be separated from r by a cut of value at most T^*. To find this set W, the following redistribution lemma (see [13] for proof) is useful.

Lemma 3 (Redistribution Lemma). *Let $N = (V, E)$ be a capacitated undirected graph. Let $X \subseteq V$ be a set of terminals such min-cut$_N(i,j) \geq 1$ for all nodes $i, j \in X$. For each $i \in X$, we are given a value $\epsilon_i \in (0,1]$. Then for any* *integer $\ell \leq \sum_{i \in X} \epsilon_i$, there exists a subset $W \subseteq X$ with $|W| \leq \ell$ vertices, and a feasible flow f in N from X to W so that (i) the total f-flow into W is at least $\frac{1-e^{-1}}{4} \cdot \ell$ and (ii) the f-flow out of $i \in X$ is at most $\epsilon_i/4$.*

We apply this lemma to $H = G \setminus \Phi^*$ with terminal set S_h, but with capacities scaled down by M. Since for any cut separating $x, y \in S_h$, the root r lies on one side on this cut (say on y's side), min-cut$_H(x, y) \geq M$—hence the scaled-down capacities satisfy the conditions of the lemma. Now set $\ell = k$, and $\epsilon_i := \alpha_i$ for each terminal $i \in S_h$; by the assumption $\sum_{i \in S_h} \epsilon_i = \sum_{i \in S_h} \alpha_i \geq k = \ell$. Hence Lemma 3 finds a subset $W \subseteq S_h$ with k vertices, and a flow f in (unscaled) graph H such that f sends a total of at least $\frac{1-1/e}{4} \cdot kM$ units into W, and at most $\frac{\alpha_i}{4} \cdot M$ units out of each $i \in S_h$. Also, there is a feasible flow g in the network H that simultaneously sends $\alpha_i \cdot M$ flow from the root to each $i \in S_h$, namely the max-flow from r to S_h. Hence the flow $\frac{g+4f}{5}$ is *feasible* in H, and sends $\frac{4}{5} \cdot \frac{1-1/e}{4} \cdot kM = \frac{1-1/e}{5} \cdot kM$ units from r into W. Finally, if $\beta > \frac{10 \cdot e}{e-1}$, we obtain that the min-cut in H separating W from r is greater than T^*: since $|W| \leq k$, this is a contradiction to the assumption that any set with at most k vertices can separated from the root in H at cost at most T^*. ■

From Claim 3 and Theorem 3, we obtain a $(3, \frac{\beta}{2}, \beta)$-discriminating algorithm for k-robust minimum cut, when $\beta \geq \frac{10e}{e-1}$. We set $\beta = \frac{10e}{e-1}$ and use Lemma 1 to infer that the approximation ratio of this algorithm is $\max\{3, \frac{\beta}{2\lambda} + \beta\} = \frac{\beta}{2\lambda} + \beta$. Since picking edges only in the second-stage is a trivial λ-approximation, the better of the two gives an approximation of $\min\{\frac{\beta}{2\lambda} + \beta,\ \lambda\} < 17$. Thus we have,

Theorem 4 (Min-cut Theorem). *There is a 17-approximation algorithm for k-robust minimum cut.*

5 Final Remarks

In this paper, we have presented a unified approach to directly solving k-robust covering problems and k-max-min problems. As mentioned in the introduction, one can show that solving the k-max-min problem also leads to a k-robust algorithm—we give a general reduction in [13]. While this general reduction leads to poorer approximation guarantees for the cardinality case, it easily extends to more general cases. Indeed, if the uncertainty sets for robust problems are not just defined by cardinality constraints, we can ask: which families of downwards-closed sets can we devise robust algorithms for? In [13] we show how to incorporate intersections of matroid-type and knapsack-type uncertainty sets.

Our work suggests several general directions for research. While the results for the k-robust case are fairly tight, can we improve on our results for general uncertainty sets to match those for the cardinality case? Can we devise algorithms to handle one matroid constraint that are as simple as our algorithms for the cardinality case? An intriguing specific problem is to find a constant factor approximation for the robust Steiner forest problem with explicit scenarios.

Acknowledgments. We thank Chandra Chekuri, Ravishankar Krishnaswamy, Danny Segev, and Maxim Sviridenko for invaluable discussions.

References

1. Agrawal, S., Ding, Y., Saberi, A., Ye, Y.: Correlation Robust Stochastic Optimization. In: SODA (2010)
2. Alon, N., Awerbuch, B., Azar, Y., Buchbinder, N., Naor, S.: The Online Set Cover Problem. In: STOC, pp. 100–105 (2003)
3. Calinescu, G., Chekuri, C., Pál, M., Vondrák, J.: Maximizing a monotone submodular function under a matroid constraint. In: Fischetti, M., Williamson, D.P. (eds.) IPCO 2007. LNCS, vol. 4513, pp. 182–196. Springer, Heidelberg (2007)
4. Dhamdhere, K., Goyal, V., Ravi, R., Singh, M.: How to pay, come what may: Approximation algorithms for demand-robust covering problems. In: FOCS, pp. 367–378 (2005)
5. Fakcharoenphol, J., Rao, S., Talwar, K.: A tight bound on approximating arbitrary metrics by tree metrics. J. Comput. System Sci. 69(3), 485–497 (2004)
6. Feige, U., Jain, K., Mahdian, M., Mirrokni, V.S.: Robust combinatorial optimization with exponential scenarios. In: Fischetti, M., Williamson, D.P. (eds.) IPCO 2007. LNCS, vol. 4513, pp. 439–453. Springer, Heidelberg (2007)
7. Fisher, M.L., Nemhauser, G.L., Wolsey, L.A.: An analysis of approximations for maximizing submodular set functions II. Mathematical Programming Study 8, 73–87 (1978)
8. Gandhi, R., Khuller, S., Srinivasan, A.: Approximation algorithms for partial covering problems. J. Algorithms 53(1), 55–84 (2004)
9. Garg, N.: Saving an epsilon: a 2-approximation for the k-mst problem in graphs. In: STOC 2005: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing, pp. 396–402 (2005)
10. Golovin, D., Goyal, V., Ravi, R.: Pay today for a rainy day: improved approximation algorithms for demand-robust min-cut and shortest path problems. In: Durand, B., Thomas, W. (eds.) STACS 2006. LNCS, vol. 3884, pp. 206–217. Springer, Heidelberg (2006)
11. Golovin, D., Nagarajan, V., Singh, M.: Approximating the k-multicut problem. In: SODA 2006: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm, pp. 621–630 (2006)
12. Gupta, A., Hajiaghayi, M.T., Nagarajan, V., Ravi, R.: Dial a ride from k-forest. In: Proceedings of the 15th Annual European Symposium on Algorithms, pp. 241–252 (2007)
13. Gupta, A., Nagarajan, V., Ravi, R.: Thresholded covering algorithms for robust and max-min optimization (2009), (full version) `http://arxiv.org/abs/0912.1045`
14. Immorlica, N., Karger, D., Minkoff, M., Mirrokni, V.S.: On the costs and benefits of procrastination: approximation algorithms for stochastic combinatorial optimization problems. In: SODA, pp. 691–700 (2004)
15. Khandekar, R., Kortsarz, G., Mirrokni, V.S., Salavatipour, M.R.: Two-stage robust network design with exponential scenarios. In: Halperin, D., Mehlhorn, K. (eds.) Esa 2008. LNCS, vol. 5193, pp. 589–600. Springer, Heidelberg (2008)
16. Motwani, R., Raghavan, P.: Randomized Algorithms. Cambridge University Press, Cambridge (1995)

17. Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.: An analysis of approximations for maximizing submodular set functions I. Mathematical Programming 14, 265–294 (1978)
18. Räcke, H.: Optimal hierarchical decompositions for congestion minimization in networks. In: STOC, pp. 255–264 (2008)
19. Ravi, R., Sinha, A.: Hedging uncertainty: approximation algorithms for stochastic optimization problems. In: Bienstock, D., Nemhauser, G.L. (eds.) IPCO 2004. LNCS, vol. 3064, pp. 101–115. Springer, Heidelberg (2004)
20. Shmoys, D., Swamy, C.: Stochastic Optimization is (almost) as Easy as Deterministic Optimization. In: FOCS, pp. 228–237 (2004)
21. Slavík, P.: Improved performance of the greedy algorithm for partial cover. Inf. Process. Lett. 64(5), 251–254 (1997)
22. Srinivasan, A.: Improved approximation guarantees for packing and covering integer programs. SIAM J. Comput. 29(2), 648–670 (1999)
23. Sviridenko, M.: A note on maximizing a submodular set function subject to knapsack constraint. Operations Research Letters 32, 41–43 (2004)
24. Swamy, C.: Algorithms for Probabilistically-Constrained Models of Risk-Averse Stochastic Optimization with Black-Box Distributions (2008), http://arxiv.org/abs/0805.0389
25. Vondrák, J.: Optimal approximation for the submodular welfare problem in the value oracle model. In: STOC, pp. 67–74 (2008)