Approximating the minmax (regret) selecting items problem*

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

Adam Kurpisz, Paweł Zieliński
Faculty of Fundamental Problems of Technology, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland, {adam.kurpisz,pawel.zielinski}@pwr.wroc.pl

Abstract

In this paper the problem of selecting p items out of n available to minimize the total cost is discussed. This problem is a special case of many important combinatorial optimization problems such as 0-1 knapsack, minimum assignment, single machine scheduling, minimum matroid base or resource allocation. It is assumed that the item costs are uncertain and they are specified as a scenario set containing K distinct cost scenarios. In order to choose a solution the min-max and min-max regret criteria are applied. It is shown that both min-max and min-max regret problems are not approximable within any constant factor unless P=NP, which strengthens the results known up to date. In this paper a deterministic approximation algorithm with performance ratio of $O(\ln K)$ for the min-max version of the problem is also proposed.

1 Preliminaries

Let $U = \{u_1, \dots, u_n\}$ be a set of items. In the deterministic case, each item $u \in U$ has a nonnegative cost $c_u \geq 0$ and we wish to choose a subset $X \subseteq U$ of exactly p items, i.e. $|X| = p$, to minimize the total cost $F(X) = \sum_{u \in X} c_u$. We will use Φ to denote the set of all feasible solutions, i.e. $\Phi = \{X \subseteq U : |X| = p\}$. This problem, denoted by Selecting Items, is one of the simplest combinatorial optimization problems and it is straightforward to check that every optimal solution is composed of p items of the smallest costs. This optimal solution can be computed in $O(n)$ time. The Selecting Items problem can be seen as a basic resource allocation problem [6]. Some other applications of this problem are described, for example, in [2, 4, 9]. We list below some important problems, whose special and polynomially solvable case is Selecting Items:

1. 0-1 Knapsack: we wish to minimize $\sum_{i=1}^{n} c_i x_i$ subject to the constraints $\sum_{i=1}^{n} x_i w_i \geq W$ and $x_i \in \{0, 1\}$ for $i = 1, \dots, n$. Clearly, we get an optimal solution of Selecting Items if $w_1 = \dots = w_n = 1$ and $W = p$.

2. The single machine scheduling problem $1|p_i = 1|\sum w_i X_i$, (see, e.g., [10]): we are given a set of jobs $J = \{1, \dots, n\}$ with unit processing times, due date windows $[\underline{d}_i, \overline{d}_i]$ and

*This work was partially supported by Polish Committee for Scientific Research, grant N N206 492938.
†Corresponding author

weights w_i for $i \in J$. A solution is a sequence of the jobs. A job is early/tardy in a given sequence if its completion time C_i is such that $C_i \notin [\underline{d}_i, \overline{d}_i]$. The aim is to find a sequence which minimizes the weighted number of early/tardy jobs. It is easily seen that we obtain an optimal solution of SELECTING ITEMS when $w_i = c_i$, $\underline{d}_i = p+1$, $\overline{d}_i = n$ for $i \in J$.

3. MINIMUM ASSIGNMENT: we are given a bipartite graph $G = (V_1 \cup V_2, E)$, $|V_1| = |V_2|$ with costs c_{ij} for $i \in V_1$ and $j \in V_2$. We seek an assignment (perfect matching) in G of the minimum cost. It can be easily verified that we obtain an optimal solution of SELECTING ITEMS by solving the minimum assignment problem in graph G depicted in Figure 1.

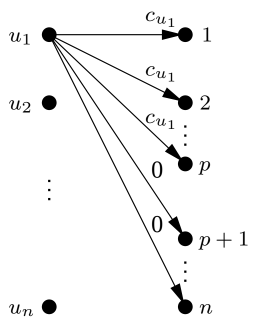

Figure 1: A cost preserving reduction from SELECTING ITEMS to MINIMUM ASSIGNMENT. The arcs from the nodes $u_2, \ldots, u_n$ are not shown.

4. MINIMUM MATROID BASE: we are given a finite set of elements $E = \{e_1, \ldots, e_n\}$ with costs c_e, $e \in E$. Let $\mathcal{F}$ be a family of subsets of E such that if $A \subseteq B$ and $B \in \mathcal{F}$, then $A \in \mathcal{F}$ and for all $A, B \in \mathcal{F}$ if $|A| < |B|$, then there is an element $e \in B \setminus A$ such that $A \cup \{e\} \in \mathcal{F}$. A base is a maximal (under inclusion) element of $\mathcal{F}$ and the set of solutions Φ consists of all the bases. The deterministic matroidal problems are one of the simplest combinatorial optimization problems for which an efficient greedy algorithms works (see, e.g. [12]). It is easy to see that the selecting items problem has a matroidal structure, where $E = U$ and $\mathcal{F} = \{X \subseteq U : |X| \leq p\}$.

In this paper we discuss the case in which the item costs are uncertain. Namely, a *scenario set* Γ is specified, which contains all the possible realizations of the item costs. No probability distribution in Γ is given. Each *scenario* is a vector $S = (c_{u_1}(S), \ldots, c_{u_n}(S)) \in \Gamma$ of the item costs which can appear with positive but perhaps unknown probability. Without loss of generality we can assume that all the item costs are nonnegative integers. In the popular and widely discussed in the literature *robust approach* we minimize a solution cost in the worst case [9]. Let $F(X, S) = \sum_{u \in X} c_u(S)$ be the cost of solution $X \in \Phi$ under scenario S and let $F^*(S)$ be the cost of an optimal solution under S. In order to choose a solution, two robust criteria, called the *min-max* and the *min-max regret*, can be adopted. In the MIN-MAX SELECTING ITEMS problem, we seek a solution which minimizes the largest cost over all scenarios, that is

$$OPT_1 = \min_{X \in \Phi} cost_1(X) = \min_{X \in \Phi} \max_{S \in \Gamma} F(X, S). \tag{1}$$

In the MIN-MAX REGRET SELECTING ITEMS problem, we wish to find a solution which minimizes the *maximal regret*, that is

$$OPT_2 = \min_{X\in\Phi} cost_2(X) = \min_{X\in\Phi}\max_{S\in\Gamma}(F(X,S) - F^*(S)).$$

The motivation of the min-max (regret) approach and a deeper discussion on the two robust criteria can be found in [9].

Previous results The min-max and min-max regret version of SELECTING ITEMS were discussed in several papers. We briefly recall some known results on both robust problems. It turns out that their computational complexity strongly depends on the way in which the scenario set Γ is defined. For the *interval uncertainty representation*, each item cost c_u is only known to belong to a closed interval $[\underline{c}_u, \overline{c}_u]$ and Γ is the Cartesian product of all the uncertainty intervals. In this case both MIN-MAX SELECTING ITEMS and MIN-MAX REGRET SELECTING ITEMS are polynomially solvable. In fact, MIN-MAX SELECTING ITEMS is trivial, since it is enough to compute an optimal solution for the pessimistic scenario $(\overline{c}_u)_{u\in U}$. On the other hand, MIN-MAX REGRET SELECTING ITEMS is more complex but it can also be solved in polynomial, $O(n\min\{p, n-p\})$, time (see [2, 4]). It is worth pointing out that MIN-MAX REGRET SELECTING ITEMS is one of a few nontrivial min-max regret combinatorial optimization problem with interval costs which is known to be polynomially solvable (see [1] for a survey).

Another method of defining the scenario set Γ, which is later discussed in this paper, is to list all possible scenarios - the *discrete scenario uncertainty representation*. Hence $\Gamma = \{S_1, \dots, S_K\}$ contains $K \geq 1$ distinct and explicitly given cost vectors. When K is constant, it has been shown in [2] that MIN-MAX SELECTING ITEMS is weakly NP-hard for two scenarios (i.e. when $K = 2$). Using a slightly modified proof one can also show that MIN-MAX REGRET SELECTING ITEMS is weakly NP-hard for $K = 2$. For constant K, the MIN-MAX (REGRET) SELECTING ITEMS problem admits a fully polynomial time approximation scheme (FPTAS), which easily follows from the results obtained in [1, 7, 9]. This FPTAS, however, is exponential in K, so its applicability is limited. On the other hand, if K is unbounded (it is a part of the input), then MIN-MAX (REGRET) SELECTING ITEMS is strongly NP-hard and not approximable within $2-\epsilon$ for any $\epsilon > 0$ unless P=NP [8]. In this case, both robust problems have a simple deterministic K-approximation algorithm, which outputs an optimal solution for the average costs $c_u = \frac{1}{K}\sum_{S\in\Gamma} c_u(S)$, $u \in U$, (see, e.g., [1]). Furthermore, MIN-MAX SELECTING ITEMS admits a randomized $O(\ln K)$-approximation algorithm [8].

It is worth noting that MIN-MAX SELECTING ITEMS under the discrete scenario uncertainty representation can be seen as an optimization problem in $[0,\infty)^K$ denoted here by SELECTING VECTORS. Namely, we are given a set U of n K-dimensional vectors $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n$ from $[0,\infty)^K$. The goal is to choose a subset $X \subseteq U$ of exactly p vectors that minimizes $||\sum_{u\in X}\boldsymbol{x}_u||_\infty$, where $||\cdot||_\infty$ denotes the standard l_∞ norm.

Our results Up to now, it has been an open problem whether MIN-MAX (REGRET) SELECTING ITEMS can be approximated within a constant factor when K is unbounded. Providing an answer is important, because SELECTING ITEMS is one of the simplest combinatorial optimization problems and it is a special case of some other basic problems listed at the beginning of this section. In this paper we show that the answer is negative. Namely, for any $\gamma > 1$ there is no γ-approximation algorithm for MIN-MAX (REGRET) SELECTING ITEMS unless P=NP. Note that this result remains valid for the minmax (regret) versions of all the more general problems listed at the beginning of this section. Moreover, we will construct a deterministic polynomial time approximation algorithm with performance ratio of $O(\ln K)$ for Min-max Selecting Items, which strengthens the result obtained in [8], where only a randomized $O(\ln K)$-approximation algorithm was designed. We also show that in some cases the approximation ratio of the algorithm is better than $O(\ln K)$.

2 Hardness of approximation of Min-max (Regret) Selecting Items

In this section we prove the main result of this paper. We start by recalling some graph-theoretic definitions. Let $G = (V, E)$ be an undirected graph. We use $deg(v)$ to denote the degree of node $v \in V$ and $deg(G) = \max_{v \in V} deg(v)$ to denote the maximal node degree in G. Recall that an independent set W in G is a subset of the nodes of G such that no two nodes in W are incident in G. The number $\alpha(G)$ is called the *independence number* of G and denotes the size of the largest independent set in G. A *clique* in G is a subset of the nodes of G which forms a complete subgraph of G. The number $\omega(G)$ is called the *clique number* of G and denotes the size of the largest clique in G. In [3] (Lemma 4.3) a relationship between $\alpha(G)$ and $\omega(G)$ was provided. For completeness, we prove now a similar but a little bit more precise result.

Lemma 1. *Let $G = (V, E)$ and $l \geq 1$ If $deg(G) < l$, then $\alpha(G) \geq \left\lceil \frac{|V|}{l+1} \right\rceil$.*

Proof. Let us choose any node $v_1 \in V$. Let us remove the node v_1 and all its incident nodes from G. We have thus removed at most $\lceil l \rceil$ nodes from G. We then choose one of the remaining nodes, say v_2 and repeat the previous construction. We proceed in this way until all the nodes of G are removed. As a result we obtain a sequence of nodes $v_1, v_2, \ldots, v_k$ which form an independent set in G. Hence $\alpha(G) \geq k = \left\lceil \frac{|V|}{\lceil l \rceil} \right\rceil \geq \left\lceil \frac{|V|}{l+1} \right\rceil$. □

Lemma 2. *Let $G = (V, E)$ and $l \in \{1, \ldots, |V|\}$. If $\omega(G) \leq l$, then $\alpha(G) \geq \lfloor |V|^{1/l} \rfloor$.*

Proof. We prove this lemma by induction on l. The case $l = 1$ is obvious. Suppose that the lemma is true for all integers $1, \ldots, l-1$ and assume that $\omega(G) = l$. We consider two cases. (i) $deg(G) < |V|^{(l-1)/l}$. Then, according to Lemma 1, we get

$$\alpha(G) \geq \left\lceil \frac{|V|}{|V|^{(l-1)/l} + 1} \right\rceil = \left\lceil \frac{|V|}{|V|^{(l-1)/l}} - \frac{|V|}{|V|^{2(l-1)/l} + |V|^{(l-1)/l}} \right\rceil \geq \lfloor |V|^{1/l} \rfloor,$$

where the last inequality follows from the fact that $l \geq 2$ and $|V|/(|V|^{2(l-1)/l} + |V|^{(l-1)/l}) < 1$. (ii) $deg(G) \geq |V|^{(l-1)/l}$. Let us choose a node $v \in V$ in G with the maximum degree. Let us form graph $G' = (V', E')$ induced by the nodes incident to v. It must hold $\omega(G') \leq l - 1$. Indeed, if $\omega(G') = l$, then G' contains a clique of size l, which together with node v would create a clique of size $l+1$ in G which is a contradiction. Now, using the induction hypothesis and the fact that $|V'| \geq |V|^{(l-1)/l}$, we get $\alpha(G) \geq \alpha(G') \geq \lfloor |V'|^{1/(l-1)} \rfloor \geq \lfloor |V|^{1/l} \rfloor$. □

In order to prove the main result, we use a type of decision problem, called a *gap problem* (see, e.g. [5]). In a gap version of an optimization (maximization) problem we are given two functions $s(n)$ and $b(n)$ of the input size n. The set of instances is the union of all yes-instances whose cost is at least $b(n)$ and all no-instances whose cost is less than $s(n)$. Consider the following gap version of the Max Independent Set problem, denoted as $\mathbf{gapIndSet}_{b,s}$. We are given an undirected graph $G = (V, E)$. The graph G is a yes-instance if $\alpha(G) \geq b(|V|)$ and it is a no-instance if $\alpha(G) < s(|V|)$. An algorithm for solving $\mathbf{gapIndSet}_{b,s}$ should distinguish between the yes and no instances, provided that G is either yes or no instance. The following theorem states that any such polynomial time algorithm would imply P=NP:

Theorem 1 ([16]). *For every $\epsilon \in (0, 1/2)$, $\mathbf{gapIndSet}_{b_\epsilon,s_\epsilon}$ is NP-hard, where $b_\epsilon(|V|) = |V|^{1-\epsilon}$ and $s_\epsilon(|V|) = |V|^\epsilon$.*

We now prove the following result:

Theorem 2. *For any constant $\gamma > 1$ and for every graph $G = (V, E)$ we can construct in polynomial time an instance (U, p, Γ) of* Min-max Selecting Items *such that*

(i) if $\alpha(G) \geq \lceil |V|^{1-\epsilon} \rceil$, then there is a feasible solution $X \in \Phi$ such that $cost_1(X) \leq 1$,

(ii) if $\alpha(G) < \lfloor |V|^\epsilon \rfloor$, then every feasible solution $X \in \Phi$ is such that $cost_1(X) > \gamma$,

where $\epsilon > 0$ is such that $\frac{1}{1+\gamma} \geq \epsilon$.

Proof. Let us fix the constant $C = \lfloor \gamma \rfloor$ and let us fix ϵ such that $\frac{1}{1+\gamma} \geq \epsilon$. The construction is as follows. We associate with each node $v \in V$ an item u, hence $U = V$ and $n = |V|$. Then we form scenario set Γ as follows. We enumerate all the cliques in G of size $C + 1$. For each such a clique, namely $\{v_{i_1}, \dots, v_{i_{C+1}}\}$, we create scenario $S \in \Gamma$ under which the costs of all $v_{i_1}, \dots, v_{i_{C+1}}$ are 1 and the costs of all the remaining items are 0. Obviously $|\Gamma| \leq \binom{|V|}{C+1} = O(|V|^{C+1})$. Finally, we set $p = \lceil |V|^{\epsilon\gamma} \rceil$. The presented construction is polynomial in the size of G, provided that γ is a constant.

We now prove the implication (i). Assume that $\alpha(G) \geq \lceil |V|^{1-\epsilon} \rceil$. Since $\frac{1}{1+\gamma} \geq \epsilon$, $\alpha(G) \geq \lceil |V|^{\epsilon\gamma} \rceil$. So there exists an independent set $V' \subseteq V$ in G of size $\lceil |V|^{\epsilon\gamma} \rceil = p$. Consider the solution $X = V'$. If, on the contrary $cost_1(X) > 1$, then there is scenario $S \in \Gamma$ such that $F(X, S) > 1$. This implies that there are two nodes $v_i, v_j \in V'$ which belong to some clique of size $C + 1 > 1$, which contradicts the assumption that V' is independent.

We now prove the implication (ii) by proving its contraposition. Assume that there is a feasible solution X such that $cost_1(X) = \max_{S \in \Gamma} F(X, S) \leq \gamma$. Since $cost_1(X)$ is integer, $cost_1(X) \leq C$. Let G_X be the subgraph of G induced by the nodes corresponding to the items in X. It holds $\omega(G_X) \leq C$, since otherwise $V' = X$ contains a clique of size $C + 1$ and $F(X, S) > C$ for some scenario $S \in \Gamma$. After applying Lemma 2 to G_X we get $\alpha(G_X) \geq \lfloor p^{1/C} \rfloor$ and $\alpha(G) \geq \lfloor p^{1/C} \rfloor$, which is due to the fact that G_X is a subgraph of G. Since $C \leq \gamma$ and $p = \lceil |V|^{\epsilon\gamma} \rceil$, we get $\alpha(G) \geq \lfloor \lceil |V|^{\epsilon\gamma} \rceil^{1/\gamma} \rfloor \geq \lfloor |V|^\epsilon \rfloor$. □

We now prove the following theorem, which is the main result in this paper:

Theorem 3. *For any constant $\gamma > 1$, there is no polynomial time algorithm that approximates* Min-max Selecting Items *within γ unless P=NP.*

Proof. This result follows from Theorems 1 and 2. Indeed, having a γ-approximation algorithm for Min-max Selecting Items , we could solve in polynomial time the NP-hard problem $\mathbf{gapIndSet}_{|V|^{1-\epsilon},|V|^\epsilon}$ for some $\epsilon \in (0, 1/2)$, which would imply P=NP. □

Theorem 4. *For any constant $\gamma > 1$, there is no polynomial time algorithm that approximates* Min-max Regret Selecting Items *within γ unless P=NP.*

Proof. We show a simple cost preserving reduction from Min-max Selecting Items to Min-max Regret Selecting Items . Given an instance (U, p, Γ) of Min-max Selecting Items we add to U additional p dummy items, which have costs equal to 0 under all scenarios in Γ. We also add one additional scenario S' under which all the dummy items have a large cost, say $\sum_{u\in U}\sum_{S\in\Gamma} c_u(S)$, and the costs of all the original items are 0. It is clear that $F^*(S) = 0$ for all scenarios in Γ', where $\Gamma' = \Gamma \cup \{S'\}$ is the extended scenario set. Since each dummy item has a large cost under S', no such an item will be contained in an optimal solution. Now its clear that $F(X, S) = F(X, S) - F^*(S)$ for all $X \in \Phi$ and $S \in \Gamma'$. Hence $cost_1(X) = cost_2(X)$ for all $X \in \Phi$. □

The following corollary follows immediately from Theorems 3 and 4.

Corollary 1. *If $P \neq NP$, then for any constant $\gamma > 1$ there is no polynomial time algorithm that approximates within γ the following problems:* Min-max (Regret) 0-1 Knapsack, Min-max (Regret) $1|p_i = 1|\sum w_i X_i$, Min-max (Regret) Minimum Assignment, Min-max (Regret) Minimum Matroid Base *and* Selecting Vectors.

3 Approximation algorithm for Min-max Selecting Items

In this section, we construct a deterministic approximation algorithm for the Min-max Selecting Items problem, whose performance ratio is $O(\ln K)$. Let us fix a parameter $C > 0$ and let $U(C) \subseteq U$ be the set of all the items $u \in U$ for which $c_u(S) \leq C$ for all scenarios $S \in \Gamma$. Consider the following linear program:

$$
\begin{aligned}
LP(C):\quad & \sum_{u\in U(C)} x_u = p, & & \\
& \sum_{u\in U(C)} c_u(S)x_u \leq C & \text{for} \quad & S \in \Gamma, \\
& 0 \leq x_u \leq 1 & \text{for} \quad & u \in U,
\end{aligned}
\tag{2}
$$

We set $x_u = 0$ for all $u \notin U(C)$. From now on C^* denotes the smallest value of the parameter C for which $LP(C)$ is feasible. Clearly, C^* is a lower bound on OPT_1. Let $\alpha > 1$ be a given scaling parameter, whose precise value will be specified later and let $c_{\max} = \max_{u\in U, S\in\Gamma} c_u(S)$. Consider a randomized algorithm for the Min-max Selecting Items problem shown in the form of Algorithm 1. This algorithm performs, in Step 5, a randomized rounding of a solution $(x^*_u)_{u\in U(C^*)} \in [0, 1]^n$ to the linear program $LP(C^*)$. We derandomize this step and execute Step 6 (if necessary) to obtain a feasible solution to Min-max Selecting Items, whose cost is no more than $O(\ln K)$ times the optimal value OPT_1. To analyze Step 5 of the algorithm, we will use the following well-known versions of Chernoff bounds [11, 13]:

Theorem 5. *Let $a_1, a_2, \ldots, a_r$ be reals in $(0, 1]$. Let $Z_1, \ldots, Z_r$ be independent Poisson trials with $\mathbf{E}[Z_j] = p_j$. Let $\Psi = \sum_{j=1}^r a_j Z_j$, $\mu = \mathbf{E}[\Psi] > 0$ and $\delta > 0$. Then $\Pr[\Psi > (1 + \delta)\mu] < F(\mu, \delta)$, where $F(\mu, \delta) = (e^\delta/(1 + \delta)^{(1+\delta)})^\mu$.*

Algorithm 1: Randomized algorithm for MIN-MAX SELECTING ITEMS

Step 1. *Solving LP relaxation:* Use a binary search to find the minimal value $C^* \in [0, pc_{\max}]$ such that there exists a feasible solution $(x^*_u)_{u\in U(C^*)}$ to $LP(C^*)$. Let $r = |U(C^*)| \geq p$.
Step 2. Let us number the items in $U(C^*)$ so that $x^*_1 \geq x^*_2 \geq \cdots \geq x^*_r$.
Step 3. *Scaling:* Let $x'_i = \min\{\alpha x^*_i, 1\}$ for $i \in [r]$ and let $i^* = |\{i \in [r] : x'_i \geq 1\}|$
Step 4. If $p - i^* < 2\ln(K+1)$ then $X := \{1, \ldots, p\}$ and return X.
Step 5. *Rounding:* If $p - i^* \geq 2\ln(K+1)$ then $X := \{1, \ldots, i^*\}$ and for each $j = i^*+1, \ldots, r$ add item j to X with probability x'_j.
Step 6. If $|X| > p$ then remove some arbitrary items from X so that $|X| = p$.
Step 7. Return X.

Let us fix $\epsilon > 0$ and let $\Delta(\mu, \epsilon)$ be the value of δ for which $F(\mu, \delta) = \epsilon$, that is $F(\mu, \Delta(\mu, \epsilon)) = \epsilon$. The following bounds on $\Delta(\mu, \epsilon)$ can be found in [13]:

$$\Delta(\mu, \epsilon) \leq \begin{cases} (\mathrm{e}-1)\sqrt{(\ln 1/\epsilon)/\mu} & \text{for } \mu > \ln 1/\epsilon, \\ \frac{\mathrm{e}\ln 1/\epsilon}{\mu \ln((\mathrm{e}\ln 1/\epsilon)/\mu)} & \text{for } \mu \leq \ln 1/\epsilon. \end{cases} \tag{3}$$

Theorem 6. *Let $Z_1, \ldots, Z_r$ be independent Poisson trials with $\mathbf{E}[Z_j] = p_j$. Let $\Psi = \sum_{j=1}^r Z_j$, $\mu = \mathbf{E}[\Psi] > 0$ and $0 < \delta \leq 1$. Then $\Pr[\Psi < (1-\delta)\mu] < H(\mu, \delta)$, where $H(\mu, \delta) = (\mathrm{e}^{-\delta}/(1-\delta)^{(1-\delta)})^\mu$.*

Set $c^*_{\max} = \max_{u\in U(C^*), S\in\Gamma} c_u(S)$. Notice that $c^*_{\max} \leq C^* \leq OPT_1$. Let us define for each scenario $S \in \Gamma$ the corresponding scaled scenario $\widehat{S}$ under which $c_u(\widehat{S}) = c_u(S)/c^*_{\max}$ for all $u \in U(C^*)$. Accordingly, $c_u(\widehat{S}) \leq \widehat{C}^* \leq \widehat{OPT}_1$ for each $u \in U(C^*)$, where $\widehat{C}^* = C^*/c^*_{\max}$ and $\widehat{OPT}_1 = OPT_1/c^*_{\max}$ and also $\widehat{cost}_1(X) = cost_1(X)/c^*_{\max}$. From now on, $c_u(\widehat{S}) \in [0,1]$ for each $u \in U(C^*)$. Let $Z_{i^*+1}, \ldots, Z_r \in \{0,1\}$ be independent Poisson trials with $\Pr[Z_i = 1] = x'_i \in [0,1)$, $i = i^*+1, \ldots, r$, such that $Z_i = 1$ if the item i is added to X in Step 5. Let $\beta = 1 + \Delta(\alpha\widehat{C}^*, 1/(K+1))$.

Lemma 3. *Let $\alpha > (3+\sqrt{5})/2$. If $p - i^* \geq 2\ln(K+1)$, then*

1. *there exists a feasible solution X such that $\widehat{cost}_1(X) \leq \alpha\beta\widehat{C}^*$,*
2. *and the solution X can be constructed in polynomial time.*

Proof. We will first prove part 1 of the lemma (the nonconstructive part) by showing that the event that the solution X constructed in Step 5 is such and $|X| \geq p$ and satisfies the inequality $\widehat{cost}_1(X) \leq \alpha\beta\widehat{C}^*$ holds with a non-zero probability, i.e. we will prove that a good approximation solution exists. If $|X| > p$, then additional items can be removed so that $|X| = p$. Consider any scaled scenario $\widehat{S}_k$ for $k \in [K]$. Define random variables $Y_k = \sum_{i=i^*+1}^r c_i(\widehat{S}_k)Z_i$ for $k \in [K]$ and $Y_{K+1} = \sum_{i=i^*+1}^r Z_i$. Hence Y_k denotes the random cost of the items chosen in Step 5 under scenario k and Y_{K+1} is the random number of items chosen in Step 5. The bad event, $\boldsymbol{\xi}_k$, that the cost of X under $\widehat{S}_k$ exceeds $\alpha\beta\widehat{C}^*$ is given by

$$\boldsymbol{\xi}_k \equiv \text{“}Y_k > \mu_k(1+\delta_k)\text{”},\ \ \mu_k = \mathbf{E}[Y_k] = \sum_{i=i^*+1}^r c_i(\widehat{S})x'_i,\ \ \delta_k = \frac{\alpha\beta\widehat{C}^* - \sum_{i=1}^{i^*} c_i(\widehat{S}_k)x'_i}{\mu_k} - 1, \tag{4}$$

where $\mu_k > 0$, if $\mu_k = 0$ then $Y_k = 0$ and so $\Pr[\boldsymbol{\xi}_k] = 0$. The bad event $\boldsymbol{\xi}_{K+1}$ that the cardinality of X is less than p is given by

$$\boldsymbol{\xi}_{K+1} \equiv \text{``}Y_{K+1} < \mu_{K+1}(1-\delta_{K+1})\text{''},\ \mu_{K+1} = \mathbf{E}[Y_{K+1}] = \sum_{i=i^*+1}^{r} x_i',\ \delta_{K+1} = 1 - \frac{p - \sum_{i=1}^{i^*} x_i'}{\mu_{K+1}}. \tag{5}$$

Clearly, $\mu_{K+1} > 0$. It is easy to check that $\delta_k > 0$ for each $k \in [K+1]$. Furthermore, if $\delta_{K+1} \geq 1$ then $\Pr[\boldsymbol{\xi}_{K+1}] = 0$, because $\sum_{i=i^*+1}^{r} Z_i \geq 0$. So, we have $\delta_{K+1} \in (0,1)$. From Theorem 5, we get $\Pr[\boldsymbol{\xi}_k] < F(\mu_k, \delta_k)$ for each $k \in [K]$. Observe that $\mu_k + \sum_{i=1}^{i^*} c_i(\widehat{S}_k)x_i' \leq \alpha\widehat{C}^*$ with $\sum_{i=1}^{i^*} c_i(\widehat{S}_k)x_i' \geq 0$, $\mu_k > 0$. An analysis similar to that in the proof of Lemma 3.2 in [14] shows that $F(\mu_k, \delta_k)$ is maximized, when $\mu_k = \alpha\widehat{C}^*$ and $\sum_{i=1}^{i^*} c_i(\widehat{S}_k)x_i' = 0$. Hence, $F(\mu_k, \delta_k) \leq F(\alpha\widehat{C}^*, \beta - 1)$ for each $k \in [K]$. By using the definition of β, we obtain:

$$\Pr[\boldsymbol{\xi}_k] < F(\mu_k, \delta_k) \leq F(\alpha\widehat{C}^*, \Delta(\alpha\widehat{C}^*, 1/(K+1))) = 1/(K+1). \tag{6}$$

From Theorem 6 and the fact that $\sum_{i=1}^{i^*} x_i' = i^* < p$ we obtain:

$$\Pr[\boldsymbol{\xi}_{K+1}] < H(\mu_{K+1}, \delta_{K+1}) = \left(\frac{\mathrm{e}^{-\delta_{K+1}}}{(1-\delta_{K+1})^{(1-\delta_{K+1})}}\right)^{\mu_{K+1}} = \frac{\mu_{K+1}^{p-i^*}\mathrm{e}^{p-i^*-\mu_{K+1}}}{(p-i^*)^{p-i^*}}. \tag{7}$$

It holds $\mu_{K+1} + \alpha i^* \geq \alpha p$. Let us fix all the parameters but μ_{K+1} in the last term of (7). This term is a decreasing function of μ_{K+1} for $\mu_{K+1} > p - i^*$ and it is maximized when $\mu_{K+1} + \alpha i^* = \alpha p$ (the reasoning is similar in spirit to that in the proof of Lemma 5.1 in [15]). Substituting $\mu_{K+1} = \alpha(p - i^*) > p - i^*$ into (7) yields:

$$\Pr[\boldsymbol{\xi}_{K+1}] < H(\mu_{K+1}, \delta_{K+1}) \leq H(\alpha(p-i^*), 1-1/\alpha) = (\alpha\mathrm{e}^{-(\alpha-1)})^{(p-i^*)} \leq (\alpha\mathrm{e}^{-(\alpha-1)})^{2\ln(K+1)}, \tag{8}$$

where the last inequality follows from the assumption that $p - i^* \geq 2\ln(K+1)$ and the inequality $\alpha\mathrm{e}^{-(\alpha-1)} < 1$. The right hand side of (8) can be upper bounded by $(\mathrm{e}^{\ln(K+1)})^{-(\alpha-1)^2/\alpha}$ (see [15, Lemma 5.1]). Thus, for any $\alpha > (3+\sqrt{5})/2$

$$\Pr[\boldsymbol{\xi}_{K+1}] < H(\mu_{K+1}, \delta_{K+1}) \leq H(\alpha(p-i^*), 1-1/\alpha) \leq (K+1)^{-(\alpha-1)^2/\alpha} < 1/(K+1). \tag{9}$$

By the union bound and (6) and (9), we get

$$\Pr[\bigcup_{k=1}^{K+1} \boldsymbol{\xi}_k] \leq \sum_{k=1}^{K+1} \Pr[\boldsymbol{\xi}_k] < \sum_{k=1}^{K} F(\mu_k, \delta_k) + H(\mu_{K+1}, \delta_{K+1}) < (K+1)(1/(K+1)) = 1, \tag{10}$$

so X is such that $|X| \geq p$ and satisfies $\widehat{cost}_1(X) \leq \alpha\beta\widehat{C}^*$ with a non-zero probability.

The proof of part 2 (the constructive part), i.e. the construction of X in polynomial time, is deferred to Appendix A. □

The following theorem shows the approximation bounds of the derandomized Algorithm 1:

Theorem 7. *Let $\alpha \geq 2.62$ ($(3+\sqrt{5})/2 \approx 2.62$) and let X be a solution returned by the derandomized Algorithm 1. Then the following inequalities hold:*

(i) if $p - i^* < 2\ln(K+1)$, then $cost_1(X) \leq (\alpha + 2\ln(K+1))\,OPT_1$

(ii) if $p - i^* \geq 2\ln(K+1)$ *then*

$$cost_1(X) \leq \begin{cases} (\alpha + \mathrm{e}\ln(K+1))OPT_1 & \text{for } \alpha\widehat{C}^* \leq \ln(K+1), \\ \mathrm{e}\alpha\, OPT_1 & \text{for } \alpha\widehat{C}^* > \ln(K+1). \end{cases}$$

Proof. Consider first the case (i). In this case Algorithm 1 deterministically returns, in Step 4, a feasible solution X composed of the items numbered from 1 to p. Under each scenario $\hat{S}$ it holds $\sum_{i=1}^{i^*} c_i(\hat{S}) \leq \alpha \sum_{i=1}^{i^*} c_i(\hat{S})x_i^* \leq \alpha\widehat{C}^*$ and $\sum_{i=i^*+1}^{p} c_i(\hat{S}) \leq (p-i^*)\widehat{C}^* \leq (2\ln(K+1))\widehat{C}^*$. Hence $\sum_{i=1}^{p} c_i(\hat{S}) \leq (\alpha + 2\ln(K+1))\widehat{C}^* \leq (\alpha + 2\ln(K+1))\widehat{OPT_1}$. Since $cost_1(X) = \widehat{cost}_1(X)c^*_{\max}$ and $OPT_1 = \widehat{OPT}_1 c^*_{\max}$, we have (i).

Let us turn to the case (ii), in which the derandomized Step 5 and Step 6 (if necessary) are executed. From Lemma 3, it follows that, in this case, the returned solution X is such that $|X| = p$ and $\widehat{cost}_1(X) \leq \alpha\beta\widehat{C}^*$. Recall that $\beta = 1 + \Delta(\alpha\widehat{C}^*, 1/(K+1))$. If $\alpha\widehat{C}^* \leq \ln(K+1)$, then according to (3), $\Delta(\alpha\widehat{C}^*, 1/(K+1)) \leq \frac{\mathrm{e}\ln(K+1)}{\alpha\widehat{C}^*\ln((\mathrm{e}\ln(K+1))/\alpha\widehat{C}^*)} \leq (\mathrm{e}\ln(K+1))/(\alpha\widehat{C}^*)$ and thus $\widehat{cost}_1(X) \leq \alpha\widehat{C}^* + \mathrm{e}\ln(K+1) \leq \alpha\widehat{OPT}_1 + \mathrm{e}\ln(K+1)$. Hence and $cost_1(X) = \widehat{cost}_1(X)c^*_{\max}$, $OPT_1 = \widehat{OPT}_1 c^*_{\max}$ and $c^*_{\max} \leq OPT_1$, we get the first case in (ii). If $\alpha\widehat{C}^* > \ln(K+1)$, then according to (3), $\Delta(\alpha\widehat{C}^*, 1/(K+1)) \leq (\mathrm{e}-1)\sqrt{\ln(K+1)/(\alpha\widehat{C}^*)} \leq \mathrm{e}-1$, consequently $\widehat{cost}_1(X) \leq \mathrm{e}\alpha\widehat{C}^* \leq \mathrm{e}\alpha\widehat{OPT}_1$. Since $cost_1(X) = \widehat{cost}_1(X)c^*_{\max}$ and $OPT_1 = \widehat{OPT}_1 c^*_{\max}$, we obtain the second case in (ii). □

The following corollary summarizes the results presented in this section:

Corollary 2. Min-max Selecting Items *has a polynomial* $O(\ln K)$*-approximation algorithm, when* K *is unbounded.*

4 Concluding remarks

In this paper, we have investigated the Min-max Selecting Items and Min-max Regret Selecting Items problems with a discrete unbounded scenario set. We have proved that both problems are hard to approximate within any constant factor. These hardness results apply also to the min-max (regret) versions of other important combinatorial optimization problems such as: 0-1 knapsack, minimum assignment, single machine scheduling, minimum matroid base or resource allocation. However, there are still some open questions concerning Min-max (Regret) Selecting Items. The best known approximation ratios of $O(\ln K)$ for the min-max version and K for the min-max regret version are still possible to be improved, which is an interesting subject of further research.

References

[1] H. Aissi, C. Bazgan, and D. Vanderpooten. Min–max and min–max regret versions of combinatorial optimization problems: A survey. *European Journal of Operational Research*, 197:427–438, 2009.

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

[3] C. Chekuri and S. Khanna. On Multi-dimensional Packing Problems. *SIAM Journal on Computing*, 33:837–851, 2004.

[4] E. Conde. An improved algorithm for selecting p items with uncertain returns according to the minmax regret criterion. *Mathematical Programming*, 100:345–353, 2004.

[5] O. Goldreich. On Promise Problems: A Survey. In O. Goldreich, A. L. Rosenberg, and A. L. Selman, editors, *Essays in Memory of Shimon Even*, volume 3895 of *Lecture Notes in Computer Science*, pages 254–290. Springer-Verlag, 2006.

[6] T. Ibaraki and N. Katoh. *Resource Allocation Problems.* The MIT Press, 1988.

[7] H. Iida. A note on the max-min 0-1 knapsack problem. *Journal of Combinatorial Optimization*, 3:89–94, 2004.

[8] A. Kasperski and P. Zieliński. A randomized algorithm for the min-max selecting items problem with uncertain weights. *Annals of Operations Research*, 172:221–230, 2009.

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

[10] A. Lann and G. Mosheiov. Single machine scheduling to minimize the number of early and tardy jobs. *Computers & Operations Research*, 23:769–781, 1996.

[11] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[12] C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity.* Dover Publications Inc., 1998.

[13] P. Raghavan. Probabilistic Construction of Deterministic Algorithms: Approximating Packing Integer Programs. *Journal of Computer and System Sciences*, 37:130–143, 1988.

[14] A. Srinivasan. Improved Approximation Guarantees for Packing and Covering Integer Programs. *SIAM Journal on Computing*, 29:648–670, 1999.

[15] A. Srinivasan. An Extension of the Lovász Local Lemma, and its Applications to Integer Programming. *SIAM Journal on Computing*, 36:609–634, 2006.

[16] D. Zuckerman. Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number. *Theory of Computing*, 3:103–128, 2007.

A The proof of Lemma 3 (part 2)

We need to show that solution X can be constructed in polynomial time. In order to do this, we will apply the standard method of conditional probabilities and pessimistic estimators (see, e.g. [13]). Let us define a vector $(z_{i^*+1},\ldots,z_r)$, where $z_i=1$ if the item i is picked in Step 5 of Algorithm 1 and $z_i=0$ otherwise, $i=i^*+1,\ldots,r$. We say that a *failure* occurs if, after performing Step 5, the cost of X under some scenario $\widehat{S}_k$, $k\in[K]$, exceeds $\alpha\beta\widehat{C}^*$, or $|X|<p$, i.e. "$\mathbf{fail}\equiv\bigcup_{k=1}^{K}\boldsymbol{\xi}_k\cup\boldsymbol{\xi}_{K+1}$" Let $P_i(\mathbf{fail}|z_{i^*+1},\ldots,z_{i-1})$ be the probability of failure for items $i,\ldots,r$ conditioned by the event that variables $z_{i^*+1},\ldots,z_{i-1}$ have been already

fixed. It is clear that $P_{i^*+1}(\mathbf{fail}) \geq \min\{P_{i^*+2}(\mathbf{fail}|z_{i^*+1} = 0), P_{i^*+2}(\mathbf{fail}|z_{i^*+1} = 1)\}$. We can thus assign 0 or 1 to z_{i^*+1} without increasing the probability of failure. Determining an assignment $z_{i^*+1}, \ldots, z_r$ can be seen as walking down a binary decision tree whose levels correspond to the subsequent variables and the sons of each node, except for the leaves, correspond to the possible assignments to a variable. The nodes of the tree are labeled by the conditional probabilities, in particular the root is labeled by $P_{i^*+1}(\mathbf{fail})$ and the leaves are labeled as $P_{r+1}(\mathbf{fail}|z_{i^*+1}, \ldots, z_r)$. Hence, each 0-1 assignment to $z_{i^*+1}, \ldots, z_r$ corresponds to a path from the root to a leaf. For each node, except for the leaves, there is a son whose label is not greater than a label of the node and by choosing at each step such a son we get an assignment which satisfies

$$P_{r+1}(\mathbf{fail}|z_{i^*+1}, \ldots, z_r) \leq P_r(\mathbf{fail}|z_{i^*+1}, \ldots, z_{r-1}) \leq \cdots \leq P_{i^*+2}(\mathbf{fail}|z_{i^*+1}) \leq P_{i^*+1}(\mathbf{fail}).$$

However, from part 1 of the lemma we know that $P_{i^*+1}(\mathbf{fail}) < 1$ (see (10)) and, since $z_{i^*+1}, \ldots, z_r$ is a complete assignment, it holds $P_{r+1}(\mathbf{fail}|z_{i^*+1}, \ldots, z_r) = 0$. Unfortunately, computing the conditional probabilities is not an easy task. However, we can estimate them from above by using the so-called *pessimistic estimators* $U_i(\mathbf{fail}|z_{i^*+1}, \ldots, z_{i-1})$, which can be computed in polynomial time. Using Markov's inequality we get for any $t_k > 0$, $k \in [K]$, $t_{K+1} > 0$:

$$\begin{aligned} \Pr[\boldsymbol{\xi}_k] &= \Pr[\mathrm{e}^{t_k Y_k} > \mathrm{e}^{t_k \mu_k (1+\delta_k)}] < \mathrm{e}^{-t_k \mu_k (1+\delta_k)} \mathbf{E}\left[\mathrm{e}^{t_k Y_k}\right] \\ &= \mathrm{e}^{-t_k \mu_k (1+\delta_k)} \prod_{i=i^*+1}^{r} \mathrm{E}[\mathrm{e}^{t_k c_i(\widehat{S}_k) Z_i}] = \mathrm{e}^{-t_k \mu_k (1+\delta_k)} \prod_{i=i^*+1}^{r} (x'_i \mathrm{e}^{t_k c_i(\widehat{S}_k)} + (1 - x'_i)), \quad (11) \\ \Pr[\boldsymbol{\xi}_{K+1}] &= \Pr[\mathrm{e}^{-t_{K+1} Y_{K+1}} > \mathrm{e}^{-t_{K+1} \mu_{K+1} (1-\delta_{K+1})}] < \mathrm{e}^{t_{K+1} \mu_{K+1} (1-\delta_{K+1})} \mathbf{E}\left[\mathrm{e}^{-t_{K+1} Y_{K+1}}\right] \\ &= \mathrm{e}^{t_{K+1} \mu_{K+1} (1-\delta_{K+1})} \prod_{i=i^*+1}^{r} (x'_i \mathrm{e}^{-t_{K+1}} + (1 - x'_i)). \quad (12) \end{aligned}$$

The above inequalities are strict in (11) and (12), since $\delta_k, \mu_k > 0$, $k \in [K]$, $\delta_{K+1}, \mu_{K+1} > 0$. The values of δ_k, μ_k, $k \in [K]$, and δ_{K+1}, μ_{K+1} are given, respectively, in (4) and (5). Using (11), (12) and the union bound we get

$$\begin{aligned} P_{i^*+1}(\mathbf{fail}) < \sum_{k \in [K]} \mathrm{e}^{-t_k \mu_k (1+\delta_k)} \prod_{i=i^*+1}^{r} (x'_i \mathrm{e}^{t_k c_i(\widehat{S}_k)} + (1 - x'_i)) + \\ + \mathrm{e}^{t_{K+1} \mu_{K+1} (1-\delta_{K+1})} \prod_{i=i^*+1}^{r} (x'_i \mathrm{e}^{-t_{K+1}} + (1 - x'_i)) = U_{i^*+1}(\mathbf{fail}). \quad (13) \end{aligned}$$

A routine computation shows that setting $t_k = \ln(1 + \delta_k)$ and $t_{K+1} = \ln(1/(1 - \delta_{K+1}))$ in (13) yields $U_{i^*+1}(\mathbf{fail}) \leq \sum_{k \in [K]} F(\mu_k, \delta_k) + H(\mu_{K+1}, \delta_{K+1})$ (see, e.g., the proofs of Theorems 4.1 and 4.2 in [11]). From (10), we have $\sum_{k \in [K]} F(\mu_k, \delta_k) + H(\mu_{K+1}, \delta_{K+1}) < 1$. Thus $P_{i^*+1}(\mathbf{fail}) < U_{i^*+1}(\mathbf{fail}) < 1$. We now examine the effect of fixing the variable z_{i^*+1}. It holds

$$U_{i^*+1}(\mathbf{fail}) = x'_{i^*+1} U_{i^*+2}(\mathbf{fail}|z_{i^*+1} = 1) + (1 - x'_{i^*+1}) U_{i^*+2}(\mathbf{fail}|z_{i^*+1} = 0),$$

where

$$U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1) = \sum_{k\in[K]} \mathrm{e}^{-t_k\mu_k(1+\delta_k)}\mathrm{e}^{t_k c_{i^*+1}(\widehat{S}_k)} \prod_{i=i^*+2}^{r} (x'_i \mathrm{e}^{t_k c_i(\widehat{S}_k)} + (1-x'_i))$$
$$+ \mathrm{e}^{t_{K+1}\mu_{K+1}(1-\delta_{K+1})}\mathrm{e}^{-t_{K+1}} \prod_{i=i^*+2}^{r} (x'_i \mathrm{e}^{-t_{K+1}} + (1-x'_i)),$$
$$U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=0) = \sum_{k\in[K]} \mathrm{e}^{-t_k\mu_k(1+\delta_k)} \prod_{i=i^*+2}^{r} (x'_i \mathrm{e}^{t_k c_u(\widehat{S}_k)} + (1-x'_i))$$
$$+ \mathrm{e}^{t_{K+1}\mu_{K+1}(1-\delta_{K+1})} \prod_{i=i^*+2}^{r} (x'_i \mathrm{e}^{-t_{K+1}} + (1-x'_i)).$$

The estimator $U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1)$ is an upper bound on the probability $P_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1)$ and the reasoning is the same as in formula (13), so $P_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1) < U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1)$. Similarly $P_{i^*+2}(\mathbf{fail}|z_{i^*+1}=0) < U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=0)$. Since $U_{i^*+1}(\mathbf{fail})$ is a convex combination of $U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1)$ and $U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=0)$, $U_{i^*+1}(\mathbf{fail}) \geq \min\{U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1), U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=0)\}$. Therefore, we set variable z_{i^*+1} to 1, if $U_{i^*+2}(\mathbf{fail}|z_{i^*+1}=1)$ is the minimum and to 0 otherwise. After this assignment we have $P_{i^*+2}(\mathbf{fail}|z_{i^*+1}) < U_{i^*+2}(\mathbf{fail}|z_{i^*+1}) \leq U_{i^*+1}(\mathbf{fail}) < 1$. Proceeding in this way and applying the same argument to the remaining variables $i^*+2,\ldots,r$, we get a complete 0-1 assignment to $z_{i^*+1},\ldots,z_r$, corresponding to a path from the root to a leaf, such that

$$1 > U_{i^*+1}(\mathbf{fail}) \geq U_{i^*+1}(\mathbf{fail}|z_{i^*+1}) \geq \cdots \geq U_{r+1}(\mathbf{fail}|z_{i^*+1},\ldots,z_r) > P_{r+1}(\mathbf{fail}|z_{i^*+1},\ldots,z_r).$$

From the above it follows that $1 > P_{r+1}(\mathbf{fail}|z_{i^*+1},\ldots,z_r)$, so $P_{r+1}(\mathbf{fail}|z_{i^*+1},\ldots,z_r)$ must be 0 and $z_{i^*+1},\ldots,z_r$ determine a solution X such that $|X| \geq p$ and $\widehat{cost}_1(X) \leq \alpha\beta\widehat{C}^*$.