# A Motivating Examples

Our settings can be used to model many real-world optimization problems. In this section, we expand on some example real-world problems that fall into our framework:

(i) A company who wants to decide a production plan to maximize profit faces a linear program. But when entering a new market, the company may not initially know the average unit price $c_i$ for their different products in this market. Sampling $c_i$ corresponds to surveying a consumer on his willingness to buy product $i$.

(ii) A delivery company who wants to plan driver routes may not know the average traffic/congestion of road segments. Each segment $e$ (an edge in the graph) has a true average travel time $b_e$, but any given day it is $b_e$ + noise. One can formulate shortest paths as an LP where $\mathbf{b}$ is the vector of edge lengths. Sampling $b_e$ corresponds to sending an observer to the road segment $e$ to observe how long it takes to traverse on a given day.

(iii) A ride sharing company (e.g. Uber) wants to decide a set of prices for rides request but it may not know customers' likelihood of accepting the prices $c_i$. Sampling $c_i$ in this setting corresponds to posting different prices to collect information.

(iv) For the purpose of recommending the best route in real time, a navigation App, e.g., Waze[9], may want to collect traffic information or route information from distributed driver via their App.

# B Change of Distribution Lemma

Some of our lower bound proofs are based on the work Chen et al. [2017]. For self-containedness, we restate some of the lemmas in Chen et al. [2017].

A key element to derive the lower bounds is the Change of Distribution lemma, which was first formulated in Kaufmann et al. [2016] to study best arm identification problem in multi-armed bandit model. The lemma provides a general relation between the expected number of draws and Kullback-Leibler divergences of the arms distributions. The core elements of the model that the lemma can be applied to are almost the same as the classical bandit model. We will state it here and explain the applicability of our setting. In the bandit model, there are $n$ arms, with each of them being characterized by an unknown distribution $\nu_i, i = 1, 2, ..., n$. The bandit model consists of a sequential strategy that selects an arm $a_t$ at time $t$. Upon selection, arm $a_t$ reveals a reward $z_t$ generated from its corresponding distribution. The rewards for each arm form an i.i.d. sequence. The selection strategy/algorithm invokes a stopping time $T$ when the algorithm will terminate and output a solution. To present the lemma, we define filtration $(\mathcal{F}_t)_{t\geq 0}$ with $\mathcal{F}_t = \sigma(a_1, z_1, \ldots, a_t, z_t)$.

[9]www.waze.com

If we consider a LP instance $\mathcal{I}$ with unknown parameters $d$ as a bandit model, an unknown parameter $d_i$ will correspond to an arm $a$ in the Lemma B.2, and thus $\nu_a$ is just the Gaussian distribution with mean $d_i$ and variance 1 (both being unknown). Each step, we select one coefficient to sample with. Then we will be able to safely apply the Change of Distribution lemma to our setting. The lemma can be stated as follows in our setting of the problem.

**Lemma B.1.** *Let $\mathcal{A}$ be a $(\delta, \epsilon_1, \epsilon_2)$-correct algorithm with $\delta \in (0, 0.1)$. Let $\mathcal{I}, \mathcal{I}'$ be two LP instances that are equal on all known parameters, and let $\mathrm{d}, \mathrm{d}'$ be their respective vectors of unknown parameters. Suppose each instance has samples distributed Gaussian with variance 1. Suppose $OPT(\mathcal{I}; \epsilon_1, \epsilon_2)$ and $OPT(\mathcal{I}'; \epsilon_1, \epsilon_2)$ are disjoint. Then letting $\tau_i$ be the number of samples $\mathcal{A}$ draws for parameter $\mathrm{d}_i$ on input $\mathcal{I}$, we have*

$$\mathbb{E} \sum_i \tau_i (d_i - d_i')^2 \geq 0.8 \ln \frac{1}{\delta}.$$

*Proof.* We use a result on bandit algorithms by Kaufmann et al. [2016], which is restated as follows.

**Lemma B.2** (Kaufmann et al. [2016])**.** *Let $\nu$ and $\nu'$ be two bandit models with $n$ arms such that for all arm $a$, the distribution $\nu_a$ and $\nu_a'$ are mutually absolutely continuous. For any almost-surely finite stopping time $T$ with respect to $(\mathcal{F}_t)$,*

$$\sum_{i=1}^{n} \mathbb{E}_\nu[N_a(T)] KL(\nu_a, \nu_a') \geq \sup_{\mathcal{E}\in\mathcal{F}_T} d(\Pr_\nu(\mathcal{E}), \Pr_{\nu'}(\mathcal{E})),$$

*where $d(x, y) = x \ln(x/y) + (1 - x) \ln((1 - x)/(1 - y))$ is the binary relative entropy function, $N_a(T)$ is the number of samples drawn on arm $a$ before time $T$ and $KL(\nu_a, \nu_{a'})$ is the KL-divergence between distribution $\nu_a$ and $\nu_{a'}$.*

Let $\mathcal{I}$ and $\mathcal{I}'$ be the two bandit models in Lemma B.2. Applying above lemma we have

$$\sum_{i=1}^{n} \mathbb{E}_{\mathcal{A},\mathcal{I}}[\tau_i] KL(\mathcal{N}(d_i, 1), \mathcal{N}(d_i', 1))$$
$$\geq d(\Pr_{\mathcal{A},\mathcal{I}}(\mathcal{E}), \Pr_{\mathcal{A},\mathcal{I}'}(\mathcal{E})), \text{ for all } \mathcal{E} \in \mathcal{F}_T,$$

where $\mathcal{N}(\mu, \sigma)$ is the Gaussian distribution with mean $\mu$ and variance $\sigma$, $\Pr_{\mathcal{A},\mathcal{I}}[\mathcal{E}]$ is the probability of event $\mathcal{E}$ when algorithm $\mathcal{A}$ is given input $\mathcal{I}$, and $\mathbb{E}_{\mathcal{A},\mathcal{I}}[X]$ is the expected value of random variable $X$ when algorithm $\mathcal{A}$ is given input $\mathcal{I}$. According to the result in Duchi, the KL-divergence for two Gaussian distribution with mean $\mu_1, \mu_2$ and variance $\sigma_1, \sigma_2$ is equal to

$$\log \frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_2^2}.$$

Thus we have $KL(\mathcal{N}(d_i, 1), \mathcal{N}(d_i', 1)) = \frac{1}{2}(d_i - d_i')^2$. We further define event $\mathcal{E}$ to be the event that algorithm $\mathcal{A}$ finally outputs a solution in set $OPT(\mathcal{I}; \varepsilon_1, \varepsilon_2)$, then since $\mathcal{A}$ is $(\delta, \varepsilon_1, \varepsilon_2)$-correct and $OPT(\mathcal{I}; \varepsilon_1, \varepsilon_2)$ is disjoint from $OPT(\mathcal{I}'; \varepsilon_1, \varepsilon_2)$, we have $\Pr_{\mathcal{A},\mathcal{I}}(\mathcal{E}) \geq 1 - \delta$ and $\Pr_{\mathcal{A},\mathcal{I}'}(\mathcal{E}) \leq \delta$. Therefore

$$\sum_{i=1}^{n} \mathbb{E}_{\mathcal{A},\mathcal{I}}[\tau_i] \frac{1}{2}(d_i - d_i')^2 \geq d(1-\delta, \delta) \geq 0.4 \ln \delta^{-1}.$$

The last step uses the fact that for all $0 < \delta < 0.1$,

$$d(1-\delta, \delta) = (1-2\delta) \ln \frac{1-\delta}{\delta} \geq 0.8 \ln \frac{1}{\sqrt{\delta}} = 0.4 \ln \delta^{-1}.$$

□

# C The Unknown Constraints Case

## C.1 Proof for Theorem 4.1

**Theorem 4.1** (Lower bound for unknown $\mathbf{b}$)**.** *Suppose we have a $(\delta, \varepsilon_1, \varepsilon_2)$-correct algorithm $\mathcal{A}$ where $\delta \in (0, 0.1), \varepsilon_1 > 0, \varepsilon_2 > 0$. Then for any $n > 0$, there exists infinitely many instances of the AIALO problem with unknown-$\mathbf{b}$ with $n$ variables with objective function $\|\mathbf{c}\|_\infty = 1$ such that $\mathcal{A}$ must draw at least*

$$\Omega\left(n \ln(1/\delta) \cdot \max\{\varepsilon_1, \varepsilon_2\}^{-2}\right)$$

*samples in expectation on each of them.*

Let $\mathcal{A}$ be a $(\delta, \varepsilon_1, \varepsilon_2)$-correct algorithm. For a positive integer $n$, consider the following LP instance $\mathcal{I}$ with $n$ variables and $n$ constraints,

$$\begin{aligned} \max \quad & x_1 \\ \text{s.t.} \quad & x_1 \leq C, \\ & x_1 + x_i \leq C, \quad \forall 2 \leq i \leq n, \\ & \mathbf{x} \geq \mathbf{0}. \end{aligned}$$

Clearly the optimal solution is $x_1^* = C$ and $x_i^* = 0$ for $i > 1$. Every constraint is a binding constraint. Now we prove that for any $k \in [n]$, algorithm $\mathcal{A}$ should take at least $\Omega\left(\ln(1/\delta) \cdot \max\{\varepsilon_1, \varepsilon_2\}^{-2}\right)$ for the $k^{th}$ constraint. We construct a new LP $\mathcal{I}'$ by subtracting the right-hand side of $k^{th}$ constraint by $2(\varepsilon_1 + \varepsilon_2)$. Then $OPT(\mathcal{I}; \varepsilon_1, \varepsilon_2)$ and $OPT(\mathcal{I}'; \varepsilon_1, \varepsilon_2)$ must be disjoint, since for any $\mathbf{x} \in OPT(\mathcal{I}'; \varepsilon_1, \varepsilon_2)$, $\mathbf{x}$ will not violate the $k^{th}$ constraint of $\mathcal{I}'$ by more than $\varepsilon_2$,

$$x_1 \leq C - 2(\varepsilon_1 + \varepsilon_2) + \varepsilon_2 < C - 2\varepsilon_1,$$

which means that $\mathbf{x} \notin OPT(\mathcal{I}; \varepsilon_1, \varepsilon_2)$. According to Lemma B.1,

$$\mathbb{E}[\tau_k] \cdot 4(\varepsilon_1 + \varepsilon_2)^2 \geq 0.8 \ln(1/\delta)$$

And since $2 \max\{\varepsilon_1, \varepsilon_2\} \geq \varepsilon_1 + \varepsilon_2$,

$$\mathbb{E}[\tau_k] = \Omega(\max\{\varepsilon_1, \varepsilon_2\}^{-2} \cdot \ln(1/\delta)).$$

## C.2 Proof for Theorem 4.2

Recall that our algorithm and the sample complexity theorem works as follows:

**Algorithm 4** Modified ellipsoid algorithm

Let $\mathcal{E}^{(0)}$ be the initial ellipsoid containing the feasible region.
Draw one sample for each $b_i$, $i \in [m]$.
Let $k = 0$ and $t = m$.
Let $T_i(t) = 1$ for all $i$.
**while** stopping criterion is not met[10] **do**
  Let $\mathbf{x}^{(k)}$ be the center of $\mathcal{E}^{(k)}$
  Call UCB method to get constraint $i$ or "feasible"
  **if** $\mathbf{x}^{(k)}$ is feasible **then**
    Let $\mathbf{x} \leftarrow \mathbf{x}^{(k)}$ if $\mathbf{x}$ is not initialized or $\mathbf{c}^T\mathbf{x}^{(k)} > \mathbf{c}^T\mathbf{x}$.
    $\mathbf{y} \leftarrow -\mathbf{c}$
  **else**
    $\mathbf{y} \leftarrow \mathbf{A}_i^T$
  **end if**
  Let $\mathcal{E}^{(k+1)}$ be the minimal ellipsoid that contains $\mathcal{E}^{(k)} \cap \{\mathbf{t} : \mathbf{y}^T\mathbf{t} \leq \mathbf{y}^T\mathbf{x}^{(k)}\}$
  Let $k \leftarrow k + 1$
**end while**
Output $\mathbf{x}$ or "failure" if it was never set.

**Theorem 4.2** (Ellipsoid-UCB algorithm)**.** *The Ellipsoid-UCB algorithm is $(\delta, \epsilon_1, \epsilon_2)$-correct and with probability $1 - \delta$, draws at most the following number of samples:*

$$O\left(\sum_{i=1}^{m} \frac{\sigma_i^2}{\Delta_{i,\varepsilon_2/2}^2} \log \frac{m}{\delta} + \sum_{i=1}^{m} \frac{\sigma_i^2}{\Delta_{i,\varepsilon_2/2}^2} \log\log\left(\frac{\sigma_i^2}{\Delta_{i,\varepsilon_2/2}^2}\right)\right).$$

[10] Our stopping criterion is exactly the same as in the standard ellipsoid algorithm, for which there are a variety of possible criteria that work. In particular, one is $\sqrt{\mathbf{c}^T\mathbf{P}^{-1}\mathbf{c}} \leq \min\{\varepsilon_1, \varepsilon_2\}$, where $P$ is the matrix corresponding to ellipsoid $\mathcal{E}^{(k)}$ as discussed above.

**Algorithm 5** UCB-method

Input $\mathbf{x}^{(k)}$
Set $\delta' = \left(\frac{\delta}{20m}\right)^{2/3}$
**loop**

1. Let $j$ be the constraint with the largest index,

$$j = \arg\max_i \mathbf{A}_i \mathbf{x}^{(k)} - \widehat{b}_{j,T_j(t)} + U_i(T_i(t)),$$

where $U_i(s) = 3\sqrt{\frac{2\sigma_i^2 \log(\log(3s/2)/\delta')}{s}}$ and $\widehat{b}_{j,T_j(t)}$ as in Definition 4.1.

2. If $\mathbf{A}_j \mathbf{x}^{(k)} - \widehat{b}_{j,T_j(t)} - U_j(T_j(t)) > 0$ return $j$.
3. If $\mathbf{A}_j \mathbf{x}^{(k)} - \widehat{b}_{j,T_j(t)} + U_j(T_j(t)) < 0$ return "feasible".
4. If $U_j(T_i(t)) < \varepsilon_2/2$ return "feasible".
5. Let $t \leftarrow t+1$
6. Draw a sample of $b_j$.
7. Let $T_j(t) = T_j(t-1)+1$.
8. Let $T_i(t) = T_i(t-1)$ for all $i \neq j$.

**end loop**

*Specifically, the number of samples used for $b_i$ is at most $\frac{\sigma_i^2}{\Delta_{i,\varepsilon_2/2}^2}\left(\log(m/\delta) + \log\log(\sigma_i^2/\Delta_{i,\varepsilon_2/2}^2)\right)$.*

Our analysis is inspired by the techniques used in Jamieson et al. [2014]. The following lemma is the same as Lemma 3 in Jamieson et al. [2014], and is simplified by setting $\varepsilon = 1/2$. We choose 1/2 only for simplicity. It will not change our result asymptotically. The constant in this lemma can be optimized by selecting parameters carefully.

**Lemma C.1.** *Let $X_1, X_2, \ldots$ be i.i.d. sub-Gaussian random variables with scale parameter $\sigma$ and mean $\mu$. We has probability at least $1 - 20 \cdot \delta^{3/2}$ for all $t \geq 1$, $\left|\frac{1}{t}\sum_{s=1}^t X_s - \mu\right| \leq L(t,\delta)$, where $L(t,\delta) = 3\sqrt{\frac{2\sigma^2 \log(\log(3t/2)/\delta)}{t}}$.*

Define event $\mathcal{A}$ to be the event that $\left|\widehat{b}_{i,t} - b_i\right| \leq U_i(t)$ for all $t \geq 0$ and $i \in [m]$. Since our definition of $U_i(t)$ is the same as $L(t, (\delta/20m)^{2/3})$ in Lemma C.1 with scale parameter $\sigma_i$, the probability that event $\mathcal{A}$ holds is at least $1-\delta$ according to union bound.

We prove the correctness and the sample number of the algorithm conditioning on that $\mathcal{A}$ holds.

**Correctness:** We first prove that the output of our algorithm satisfies relaxed feasibility and relaxed optimality when $\mathcal{A}$ holds. If our UCB-method always gives correct answer, the ellipsoid algorithm will be able to find an $\varepsilon_1$-suboptimal solution. So we only need to prove the correctness of the UCB-method.

- When UCB method returns a violated constraint $j$ in line 2, it is indeed a violated one: since $|\widehat{b}_{j,T_j(t)} - b_j| \leq U_j(T_j(t))$,

$$\begin{aligned}&\mathbf{A}_j\mathbf{x}_k - b_j\\ \geq&\mathbf{A}_j\mathbf{x}_k - \widehat{b}_{j,T_j(t)} - U_j(T_j(t))\\ >&0.\end{aligned}$$

- When it returns "feasible" in line 3, no constraint is violated:

$$\begin{aligned}&\mathbf{A}_i\mathbf{x}_k - b_i\\ \leq&\mathbf{A}_i\mathbf{x}_k - \widehat{b}_{i,T_i(t)} + U_i(T_i(t))\\ \leq&\mathbf{A}_j\mathbf{x}_k - \widehat{b}_{j,T_j(t)} + U_j(T_j(t))\\ <&0, \quad \forall i \in [m].\end{aligned}$$

- When it returns "feasible" in line 4, no constraint is violated more than $\varepsilon_2$:

$$\begin{aligned}&\mathbf{A}_i\mathbf{x}_k - b_i\\ \leq&\mathbf{A}_i\mathbf{x}_k - \widehat{b}_{i,T_i(t)} + U_i(T_i(t))\\ \leq&\mathbf{A}_j\mathbf{x}_k - \widehat{b}_{j,T_j(t)} + U_j(T_j(t))\\ \leq&\mathbf{A}_j\mathbf{x}_k - \widehat{b}_{j,T_j(t)} - U_j(T_j(t)) + 2U_j(T_j(t))\\ \leq&0 + \varepsilon_2, \quad \forall i \in [m].\end{aligned}$$

Therefore the relaxed feasibility should be satisfied and the relaxed optimality is guaranteed by ellipsoid algorithm.

**Number of samples:** We bound the number of samples used on each constraint separately. The number of samples used on constraint $i$ can be stated as the maximum $T_i(t)$ where $t$ is a mini-stage in which a sample of $b_i$ is drawn. We bound $T_i(t)$ by showing that $U_i(T_i(t))$ should be larger than a certain value if $b_i$ is sampled at mini-stage $t$. This immediately give us an upper bound of $T_i(t)$ since $U_i(t)$ is a decreasing function of $t$. Suppose $b_i$ is sampled at mini-stage $t$ in ellipsoid iteration $k$. Let $i^*$ be the constraint with largest violation. Conditioning on $\mathcal{A}$ holds, the fact that constraint $i$ have a larger index than $i^*$ gives

$$\begin{aligned}&V_i(k) + 2U_i(T_i(t))\\ \geq&\mathbf{A}_i\mathbf{x}_k - \widehat{b}_{i,T_i(t)} + U_i(T_i(t))\\ \geq&\mathbf{A}_{i^*}\mathbf{x}_k - \widehat{b}_{i^*,T_{i^*}(t)} + U_{i^*}(T_{i^*}(t))\\ \geq&V_{i^*}(k).\end{aligned} \tag{6}$$

which implies $2U_i(T_i(t)) \geq V^*(k) - V_i(k)$. Now look at line 2 in UCB-method. If a sample of $b_i$ is drawn, we should not quit in this step. So if $V_i(k) > 0$, we must have

$$\begin{aligned} & V_i(k) - 2U_i(T_i(t)) \\ \leq & \mathbf{A}_i \mathbf{x}_k - \widehat{b}_{i,T_i(t)} - U_i(T_i(t)) \\ \leq & 0. \end{aligned} \tag{7}$$

Similarly, because of line 3 in UCB-method, if $V_i(k) \leq 0$, it should be satisfied that

$$\begin{aligned} & V_i(k) + 2U_i(T_i(t)) \\ \geq & \mathbf{A}_i \mathbf{x}_k - \widehat{b}_{i,T_i(t)} + U_i(T_i(t)) \\ \geq & 0. \end{aligned} \tag{8}$$

Putting inequality (6), (7) and (8) and $U_i(T_i(t)) \geq \varepsilon_2/2$ together, we get the conclusion that $2U_i(T_i(t)) \geq \max\{V^*(k) - V_i(k), |V_i(k)|, \varepsilon_2/2\} = gap_{i,\varepsilon_2/2}(k)$ should be satisfied if we draw a sample of $b_i$ at mini-stage $t$ in ellipsoid iteration $k$.

Then we do some calculation,

$$\begin{aligned} & 2U_i(T_i(t)) \geq gap_{i,\varepsilon_2/2}(k) \\ \Rightarrow \quad & 6\sqrt{\frac{2\sigma_i^2 \log(\log(3T_i(t)/2)/\delta'}{T_i(t)}} \geq gap_{i,\varepsilon_2/2}(k) \\ \Rightarrow \quad & \frac{\log(\log(3T_i(t)/2)/\delta')}{T_i(t)} \geq \frac{gap^2_{i,\varepsilon_2/2}(k)}{72\sigma_i^2} \\ \Rightarrow \quad & T_i(t) \leq \frac{108\sigma_i^2}{gap^2_{i,\varepsilon_2/2}(k)} \log\left(\frac{20m}{\delta}\right) \\ & + \frac{72\sigma_i^2}{gap^2_{i,\varepsilon_2/2}(k)} \log\log\left(\frac{108\sigma_i^2}{gap^2_{i,\varepsilon_2/2}(k)\delta'}\right). \end{aligned} \tag{9}$$

In the last step, we use the fact that for $0 < \delta \leq 1$, $c > 0$,

$$\begin{aligned} & \frac{1}{t} \cdot \log\left(\frac{\log(3t/2)}{\delta}\right) \geq c \\ \Rightarrow \quad & t \leq \frac{1}{c} \log\left(\frac{2\log(3/(2c\delta))}{\delta}\right). \end{aligned}$$

Take maximum of (9) over all $k$ and according to the definition of $\Delta_{i,\varepsilon_2/2}$,

$$T_i(t) \leq \frac{108\sigma_i^2}{\Delta^2_{i,\varepsilon_2/2}} \log\left(\frac{20m}{\delta}\right) + \frac{72\sigma_i^2}{\Delta^2_{i,\varepsilon_2/2}} \log\log\left(\frac{108\sigma_i^2}{\Delta^2_{i,\varepsilon_2/2}\delta'}\right)$$

Therefore the overall number of samples is at most

$$O\left(\sum_i \frac{\sigma_i^2}{\Delta^2_{i,\varepsilon_2/2}} \log\frac{m}{\delta} + \sum_i \frac{\sigma_i^2}{\Delta^2_{i,\varepsilon_2/2}} \log\log\left(\frac{\sigma_i^2}{\Delta^2_{i,\varepsilon_2/2}}\right)\right).$$

# D Proofs for the Unknown Objective Function Case

## D.1 Proof for Theorem 5.1

We restate the instance-wise lower bound for unknown objective function LP problems.

**Theorem 5.1** (Instance lower bound)**.** *Let $\mathcal{I}$ be an instance of AIALO in the unknown-$\mathbf{c}$ case. For $0 < \delta < 0.1$, any $\delta$-correct algorithm $\mathcal{A}$ must draw*

$$\Omega(Low(\mathcal{I}) \ln \delta^{-1})$$

*samples in expectation on $\mathcal{I}$.*

Let $\mathcal{I}$ be a LP instance $\max_{\{\mathbf{x}:\mathbf{A}\mathbf{x}\leq\mathbf{b}\}} \mathbf{c}^T\mathbf{x}$, and $\mathcal{A}$ be a $\delta$-correct algorithm, where $0 < \delta < 0.1$. Define $t_i$ to be the expected number of samples that algorithm will draw for $c_i$ when the input is $\mathcal{I}$.

We only need to show that $5\mathbf{t}/\ln(1/\delta)$ is a feasible solution of the convex program (3) that computes $Low(\mathcal{I})$.

Consider a constraint in (3)

$$\sum_{i=1}^n \frac{(s_i^{(k)} - x_i^*)^2}{\tau_i} \leq \left(\mathbf{c}^T(\mathbf{x}^* - \mathbf{s}^{(k)})\right)^2,$$

where $\mathbf{x}^*$ is the optimal solution of $\mathcal{I}$ and $\mathbf{s}^{(k)}$ is a corner point of the feasible region of $\mathcal{I}$. To prove that $5\mathbf{t}/\ln(1/\delta)$ satisfies this constraint, we will construct a new LP instance $\mathcal{I}_{\mathbf{\Delta}}$ by adding $\mathbf{\Delta}$ to the objective function $\mathbf{c}$, such that $\mathbf{s}^{(k)}$ becomes a better solution than $\mathbf{x}^*$. We construct vector $\mathbf{\Delta}$ as follows,

$$\Delta_i = \frac{D(x_i^* - s_i^{(k)})}{t_i}, \quad \text{and} \quad D = \frac{-2\mathbf{c}^T(\mathbf{x}^* - \mathbf{s}^{(k)})}{\sum_{i=1}^n \frac{(s_i^{(k)} - x_i^*)^2}{t_i}}.$$

It is not difficult to verify that $\mathbf{x}^*$ is no longer the optimal solution of $\mathcal{I}_{\mathbf{\Delta}}$:

$$\begin{aligned} & \langle \mathbf{c} + \mathbf{\Delta}, \mathbf{x}^* - \mathbf{s}^{(k)} \rangle \\ = & \langle \mathbf{c}, \mathbf{x}^* - \mathbf{s}^{(k)} \rangle + \langle \mathbf{\Delta}, \mathbf{x}^* - \mathbf{s}^{(k)} \rangle \\ = & \langle \mathbf{c}, \mathbf{x}^* - \mathbf{s}^{(k)} \rangle - \\ & \sum_{i=1}^n \frac{2\mathbf{c}^T(\mathbf{x}^* - \mathbf{s}^{(k)})}{\sum_{i=1}^n \frac{(s_i^{(k)} - x_i^*)^2}{t_i}} \cdot \frac{x_i^* - s_i^{(k)}}{t_i} \cdot (x_i^* - s_i^{(k)}) \\ = & -\langle \mathbf{c}, \mathbf{x}^* - \mathbf{s}^{(k)} \rangle \\ < & 0. \end{aligned}$$

Then by Lemma B.1,

$$\begin{aligned}0.8\ln(1/\delta) &\le \sum_{i=1}^{n} t_i \cdot \Delta_i^2\\ &= \sum_{i=1}^{n} t_i \cdot \left(\frac{D(x_i^* - s_i^{(k)})}{t_i}\right)^2\\ &= \sum_{i=1}^{n} \frac{(x_i^* - s^{(k)})^2}{t_i} \cdot D^2\\ &= \sum_{i=1}^{n} \frac{(x_i^* - s^{(k)})^2}{t_i} \cdot \left(\frac{-2\mathbf{c}^T(\mathbf{x}^* - \mathbf{s}^{(k)})}{\sum_{i=1}^{n} \frac{(s_i^{(k)} - x_i^*)^2}{t_i}}\right)^2\\ &= 4 \cdot \frac{(\mathbf{c}^T(\mathbf{x}^* - \mathbf{s}^{(k)}))^2}{\sum_{i=1}^{n} \frac{(s_i^{(k)} - x_i^*)^2}{t_i}},\end{aligned}$$

which is equivalent to

$$\sum_{i=1}^{n} \frac{(s_i^{(k)} - x_i^*)^2}{5t_i/\ln(1/\delta)} \le (\mathbf{c}^T(\mathbf{x}^* - \mathbf{s}^{(k)}))^2.$$

Therefore $5\mathbf{t}/\ln(1/\delta)$ is a feasible solution of the convex program (3), which completes our proof.

## D.2 Proof for Theorem 5.2

We prove the worst case lower bound for unknown $\mathbf{c}$ case.

**Theorem 5.2** (Worst-case lower bound for unknown $\mathbf{c}$)**.** *Let $n$ be a positive integer and $\delta \in (0, 0.1)$. For any $\delta$-correct algorithm $\mathcal{A}$, there exists an infinite sequence of LP instances with $n$ variables, $\mathcal{I}_1, \mathcal{I}_2, \ldots$, such that $\mathcal{A}$ takes*

$$\Omega\left(Low(\mathcal{I}_k)(\ln|S_k^{(1)}| + \ln\delta^{-1})\right)$$

*samples in expectation on $\mathcal{I}_k$, where $S_k^{(1)}$ is the set of all extreme points of the feasible region of $\mathcal{I}_k$, and $Low(\mathcal{I}_k)$ goes to infinity.*

The following lemma will be used in the construction of desired LP instances.

**Lemma D.1.** *Let $n$ be a positive integer. There exists a constant $c$, a positive integer $l = \Omega(n)$ and $z = 2^{cn}$ sets $W_1, \ldots, W_z \subseteq [n]$ such that*

- *For all $i \in [z]$, we have $|W_i| = l = \Omega(n)$.*
- *For all $i \ne j$, $|W_i \cap W_j| \le l/2$.*

*Proof.* Define $l = n/10$. Let each $W_i$ be a uniformly random subset of $[n]$ with size $l$. Then it is satisfied that

$$\Pr[|W_i \cap W_j| > l/2] \le 2^{-\Omega(n)}$$

for all $1 \le i, j \le n, i \ne j$. So we can choose sufficiently small $c$ such that

$$\Pr[\exists i \ne j, |W_i \cap W_j| > l/2] \le z^2 2^{-\Omega(n)} < 1,$$

which implies the existence of a desired sequence of subsets. □

Now for any $\delta$-correct algorithm $\mathcal{A}$, we prove the existence of LP instances $\mathcal{I}_1, \mathcal{I}_2, \ldots$, which all have $n$ variables.

For simplicity, all the linear program instances we construct in this proof share the same feasible region, which we define as follows. Let $W_1, \ldots, W_z \subseteq [n]$ be the sequence of subsets in Lemma D.1. For a subset $W \subseteq [n]$, we define a point $\mathbf{p}^W$

$$p_i^W = \begin{cases} 1, & \text{if } i \in W;\\ 0, & \text{otherwise.}\end{cases}$$

The feasible region we are going to use throughout this proof is the convex hull of $\mathbf{p}^{W_1}, \ldots, \mathbf{p}^{W_z}$.

To find a desired LP instance $\mathcal{I}_k$, we first choose an arbitrary constant $\Delta_k$. We construct $z$ different LP instances $\mathcal{I}_{\Delta_k, W_1}, \ldots, \mathcal{I}_{\Delta_k, W_z}$ and show that at least one of them satisfies the condition in the theorem. Define the objective function $\mathbf{c}^{W_j}$ of $\mathcal{I}_{\Delta_k, W_j}$ to be

$$c_i^{W_j} = \begin{cases} \Delta_k, & \text{if } i \in W_j;\\ -\Delta_k, & \text{otherwise.}\end{cases}$$

Then clearly the optimal solution of $\mathcal{I}_{\Delta_k, W_j}$ is point $\mathbf{p}^{W_j}$. We define $\Pr[\mathcal{A}(\mathcal{I}_{\Delta, W_i}) = \mathbf{p}^{W_j}]$ to be the probability that algorihtm $\mathcal{A}$ outputs $\mathbf{p}^{W_j}$ when the input is $\mathcal{I}_{\Delta, W_i}$. Then we have

$$\Pr[\mathcal{A}(\mathcal{I}_{\Delta, W_i}) = \mathbf{p}^{W_i}] \ge 1 - \delta,$$

and

$$\sum_{j: j \ne i} \Pr[\mathcal{A}(\mathcal{I}_{\Delta, W_i}) = \mathbf{p}^{W_j}] \le \delta.$$

Thus there must exists $W_k$ such that

$$\Pr[\mathcal{A}(\mathcal{I}_{\Delta, W_i}) = \mathbf{p}^{W_k}] \le 2\delta/z.$$

Let $T$ be the number of samples used by algorithm $\mathcal{A}$ when the input is $\mathcal{I}_{\Delta_k, W_k}$. Since $\mathcal{A}$ is a $\delta$-correct algorithm, $\Pr[\mathcal{A}(\mathcal{I}_{\Delta, W_k}) = \mathbf{p}^{W_k}] \ge 1 - \delta > 0.9$. So if we define event $\mathcal{E}$ to be the event that $\mathcal{A}$ outputs $\mathbf{p}^{W_k}$ and apply Lemma B.2,

$$\begin{aligned}&\mathbb{E}[T] \cdot (2\Delta^2)\\ \ge& d\left(\Pr[\mathcal{A}(\mathcal{I}_{\Delta, W_k})\mathbf{p}^{W_k}], \Pr[\mathcal{A}(\mathcal{I}_{\Delta, W_i}) = \mathbf{p}^{W_k}]\right)\\ \ge& \Omega(\ln(z/\delta))\\ =& \Omega(\ln z + \ln(1/\delta)).\end{aligned}$$

Here we use the following property of $d(1-\delta,\delta)$ function: for $0<\delta<0.1$, $d(1-\delta,\delta)\geq 0.4\ln(1/\delta)$. So we get a lower bound for $\mathbb{E}[T]$,

$$\mathbb{E}[T] \geq \Omega\left(\Delta^{-2}(\ln z + \ln(1/\delta))\right).$$

Meanwhile if we look at the Instance Lower Bound, $Low(\mathcal{I}_{\Delta,W_k})$,

$$\begin{aligned}
\min_{\tau} \quad & \sum_{i=1}^{n} \tau_i \\
s.t. \quad & \sum_{i=1}^{n} \frac{(p_i^{W_j}-p_i^{W_k})^2}{\tau_i} \leq \langle \mathbf{c}^{W_k}, (\mathbf{p}^{W_k}-\mathbf{p}^{W_j})\rangle^2, \forall j \\
& \tau_i \geq 0,
\end{aligned}$$

It is easy to verify that $\tau_i = \frac{8}{l\Delta^2}$ for all $i$ is a feasible solution. So we have $Low(\mathcal{I}_{\Delta,W_k}) = \Theta(\frac{8n}{l\Delta^2}) = \Theta(\Delta^{-2})$. Therefore the number of samples that $\mathcal{A}$ will use on $\mathcal{I}_{\Delta,W_k}$ is $\Omega\left(Low(\mathcal{I}_{\Delta,W_k})(\ln z + \ln(\delta^{-1}))\right)$ in expectation.

By simply setting $\Delta_k = \frac{1}{k}$, we will get an infinite sequence of LP instances as stated in the theorem.

### D.3 Proof for Theorem 5.3

In this section, we prove the sample complexity of our successive elimination algorithm for unknown $c$ case.

**Theorem 5.3** (Sample complexity of Algorithm 3)**.** *For the AIALO with unknown-$\mathbf{c}$ problem, Algorithm 3 is $\delta$-correct and, on instance $\mathcal{I}$, with probability $1-\delta$ draws at most the following number of samples:*

$$O\left(Low(\mathcal{I})\ln\Delta^{-1}(\ln|S^{(1)}| + \ln\delta^{-1} + \ln\ln\Delta^{-1})\right),$$

*where $S^{(1)}$ is the set of all extreme points of the feasible region and $\Delta$ is the gap in objective value between the optimal extreme point and the second-best,*

$$\Delta = \max_{\mathbf{x}\in S^{(1)}} \mathbf{c}^T\mathbf{x} - \max_{\mathbf{x}\in S^{(1)}\setminus\mathbf{x}^*} \mathbf{c}^T\mathbf{x}.$$

The following lemma will be used in our proof.

**Lemma D.2.** *Given a set of Gaussian arms with unit variance and mean $c_1,\ldots,c_n$. Suppose we take $\tau_i$ samples for arm $i$. Let $X_i$ be the empirical mean. Then for an arbitrary vector $\mathbf{p}$,*

$$\Pr\left[|\mathbf{p}^T\mathbf{X} - \mathbf{p}^T\mathbf{c}| \geq \varepsilon\right] \leq 2\exp\left(-\frac{\varepsilon^2}{2\sum p_i^2/\tau_i}\right)$$

*Proof.* By definition, $\mathbf{p}^T\mathbf{X} - \mathbf{p}^T\mathbf{c}$ follows Gaussian distribution with mean 0 and variance $\sum_i p_i^2/\tau_i$. □

We define a good event $\mathcal{E}$ to be the event that $|(\mathbf{x}-\mathbf{y})^T(\widehat{\mathbf{c}}^{(r)}-\mathbf{c})| \leq \varepsilon^{(r)}/\lambda$ for all $r$ and $\mathbf{x},\mathbf{y}\in S^{(r)}$. According to Lemma D.2,

$$\Pr[\mathcal{E}] \geq 1 - \sum_r \sum_{\mathbf{x}\in S^{(r)}} \sum_{\mathbf{y}\in S^{(r)}} 2\exp -\frac{(\varepsilon/\lambda)^2}{2\sum(x_i-y_i)^2/\tau_i}.$$

Since $\tau$ satisfies the constraints in (4),

$$\begin{aligned}
&\sum_r \sum_{\mathbf{x}\in S^{(r)}} \sum_{\mathbf{y}\in S^{(r)}} 2\exp -\frac{(\varepsilon/\lambda)^2}{2\sum(x_i-y_i)^2/\tau_i} \\
\leq &\sum_r \sum_{\mathbf{x}\in S^{(r)}} \sum_{\mathbf{y}\in S^{(r)}} 2\exp\left(-\ln(2/\delta^{(r)})\right) \\
= &\sum_r \sum_{\mathbf{x}\in S^{(r)}} \sum_{\mathbf{y}\in S^{(r)}} \delta^{(r)} \\
\leq &\delta
\end{aligned}$$

Therefore $\Pr[\mathcal{E}] \geq 1-\delta$.

We first prove the correctness of the algorithm conditioning on $\mathcal{E}$.

**Lemma D.3.** *When the good event $\mathcal{E}$ holds, the optimal LP solution $\mathbf{x}^* = \max_{\mathbf{A}\mathbf{x}\leq\mathbf{b}} \mathbf{c}^T\mathbf{x}$ will not be deleted.*

*Proof.* Suppose to the contrary $\mathbf{x}^*$ is deleted in iteration $r$, i.e., $\mathbf{x}^*\in S^{(r)}$ but $\mathbf{x}^*\notin S^{(r+1)}$. Then according to (5), when the objective function is $\widehat{\mathbf{c}}^{(r)}$, $\mathbf{x}^*$ is at least $\varepsilon^{(r)}/2 - 2\varepsilon^{(r)}/\lambda$ worse than $\mathbf{x}^{(r)}$,

$$\langle \mathbf{x}^{(r)} - \mathbf{x}^*, \widehat{\mathbf{c}}^{(r)}\rangle > \varepsilon^{(r)}/2 + 2\varepsilon^{(r)}/\lambda.$$

By the definition of the optimal solution $\mathbf{x}^*$,

$$\langle \mathbf{c}, \mathbf{x}^* - \mathbf{x}^{(r)}\rangle > 0.$$

Combining the two inequalities will give

$$\langle \mathbf{c} - \widehat{\mathbf{c}}^{(r)}, \mathbf{x}^* - \mathbf{x}^{(r)}\rangle > \varepsilon^{(r)}/2 + 2\varepsilon^{(r)}/\lambda > \varepsilon^{(r)}/\lambda,$$

contradictory to that event $\mathcal{E}$ holds. □

We then bound the number of samples conditioning on $\mathcal{E}$. We first prove the following lemma.

**Lemma D.4.** *When event $\mathcal{E}$ holds, all points $\mathbf{s}$ in set $S^{(r+1)}$ satisfies*

$$\langle \mathbf{c}, \mathbf{x}^* - \mathbf{s}\rangle < \varepsilon^{(r)}.$$

*after the $r^{th}$ iteration.*

*Proof.* Suppose when entering the $r^{th}$ iteration, there exists $\mathbf{s}\in S^{(r)}$ such that $\langle \mathbf{c}, \mathbf{x}^* - \mathbf{s}\rangle > \varepsilon^{(r)}$. Then since $\mathcal{E}$ holds and $\lambda = 10$,

$$\begin{aligned}
\langle \mathbf{c}, \mathbf{x}^* - \mathbf{s}\rangle &> \langle \widehat{\mathbf{c}}^{(r)}, \mathbf{x}^* - \mathbf{s}\rangle - \varepsilon^{(r)}/\lambda \\
&> (1-1/\lambda)\varepsilon^{(r)} \\
&> \varepsilon^{(r)}/2 + 2\varepsilon^{(r)}/\lambda.
\end{aligned}$$

By Lemma D.3, we have $\mathbf{x}^* \in S^{(r)}$. Therefore $\mathbf{s}$ will be deleted in this iteration. $\square$

Now consider a fixed iteration $r$. Let $\tau^*$ be the optimal solution of the convex program (3) that computes $low(\mathcal{I})$. Define $\alpha = 32\lambda^2 \ln(2/\delta^{(r)})$. We show that $\mathbf{t} = \alpha\tau^*$ is a feasible solution in the convex program (4) that computes $LowAll(S^{(r)}, \varepsilon^{(r)}, \delta^{(r)})$. For any $\mathbf{x}, \mathbf{y} \in S^{(r)}$,

$$\begin{aligned}\sum \frac{(x_i - y_i)^2}{t_i} &= \frac{1}{\alpha}\sum \frac{(x_i - y_i)^2}{\tau_i^*} \\ &= \frac{1}{\alpha}\sum \frac{(x_i - x_i^* + x_i^* - y_i)^2}{\tau_i^*} \\ &\le \frac{1}{\alpha}\sum \frac{2(x_i - x_i^*)^2 + 2(x_i^* - y_i)^2}{\tau_i^*}\end{aligned}$$

due to the fact that $(a+b)^2 \le 2a^2 + 2b^2$ for all $a, b \in \mathbb{R}$.

Since $\tau^*$ satisfies the constraints in $Low(\mathcal{I})$ function (3),

$$\begin{aligned}&\frac{1}{\alpha}\sum \frac{2(x_i - x_i^*)^2 + 2(x_i^* - y_i)^2}{\tau_i^*} \\ \le &\frac{2}{\alpha}\left((\mathbf{c}^T(\mathbf{x}^* - \mathbf{x}))^2 + (\mathbf{c}^T(\mathbf{x}^* - \mathbf{y}))^2\right)\end{aligned}$$

And because of Lemma D.4,

$$\begin{aligned}&\frac{2}{\alpha}\left((\mathbf{c}^T(\mathbf{x}^* - \mathbf{x}))^2 + (\mathbf{c}^T(\mathbf{x}^* - \mathbf{y}))^2\right) \\ \le &\frac{4}{\alpha}(\varepsilon^{(r-1)})^2 = \frac{(\varepsilon^{(r)})^2}{2\lambda^2 \ln(2/\delta^{(r)})}.\end{aligned}$$

So we have proved that $\mathbf{t} = \alpha\tau^*$ is a feasible solution of the convex program that computes $LowAll(S^{(r)}, \varepsilon^{(r)}, \delta^{(r)})$. Thus the number of samples used in iteration $r$, $\sum_{i=1}^n t_i^{(r)}$, is no more than

$$\begin{aligned}&\sum_{i=1}^n t_i^{(r)} \le \sum_{i=1}^n t_i = \alpha \sum_i \tau_i^* \\ =&O(Low(\mathcal{I})(\ln |S^{(r)}| + \ln \delta^{-1} + \ln r)\end{aligned}$$

Conditioning on $\mathcal{E}$, the algorithm will terminate before $\lfloor \log(\Delta^{-1}) \rfloor + 1$ iterations according to Lemma D.4. Therefore the total number of samples is

$$O\left(Low(\mathcal{I}) \ln \Delta^{-1} (\ln |S^{(1)}| + \ln \delta^{-1} + \ln \ln \Delta^{-1})\right).$$