Online Learning with Gaussian Payoffs and Side Observations

Yifan Wu\(^1\)  \hspace{1cm} \textbf{András György}\(^2\)  \hspace{1cm} Csaba Szepesvári\(^1\)

\(^1\)Department of Computing Science
University of Alberta

\(^2\)Department of Electrical and Electronic Engineering
Imperial College London

January 14, 2016
Outline

1. Introduction

2. Has This Been Done Before?

3. Results
   - Lower Bounds
   - Algorithms/Upper Bounds

4. Summary
Outline

1 Introduction

2 Has This Been Done Before?

3 Results
   - Lower Bounds
   - Algorithms/Upper Bounds

4 Summary
A Fishy Problem

- Each day, you get to choose a fishing spot.
- Which one to choose?
- Every fish you catch: +1 cookies.
- No fish: −10 cookies.
- Fish distribution is i.i.d.
- With some probability, you will see neighboring sites’ yield for the day.
The Fishing Game

Choosing a fishing spot: $K$ actions.
The Fishing Game
Choosing a fishing spot: \( K \) actions.

\( \theta_1, \ldots, \theta_K \): (unknown) mean rewards for the \( K \) spots.
The Fishing Game
Choosing a fishing spot: $K$ actions.

$\theta_1, \ldots, \theta_K$: (unknown) mean rewards for the $K$ spots.

For rounds $t = 1, \ldots, T$: 

$\theta_1, \ldots, \theta_K$: (unknown) mean rewards for the $K$ spots.
The Fishing Game
Choosing a fishing spot: $K$ actions.

$\theta_1, \ldots, \theta_K$: (unknown) mean rewards for the $K$ spots.

For rounds $t = 1, \ldots, T$:

- Choose a fishing spot $l_t \in [K] := \{1, \ldots, K\}$;
The Fishing Game
Choosing a fishing spot: $K$ actions.

$\theta_1, \ldots, \theta_K$: (unknown) mean rewards for the $K$ spots.

For rounds $t = 1, \ldots, T$:

- Choose a fishing spot $l_t \in [K] := \{1, \ldots, K\}$;
- Incur reward $Y_t \in \mathbb{R}$ with mean $\theta_{l_t}$;
The Fishing Game

Choosing a fishing spot: $K$ actions.

$\theta_1, \ldots, \theta_K$: (unknown) mean rewards for the $K$ spots.

For rounds $t = 1, \ldots, T$:

- Choose a fishing spot $I_t \in [K] := \{1, \ldots, K\}$;
- Incur reward $Y_t \in \mathbb{R}$ with mean $\theta_{I_t}$;
- Observe $X_t \in \mathbb{R}^K$; noisy reward observations for all the sites ($Y_t = X_{t,I_t}$).

Assumptions $E[X_t,k] = \theta_k$, and $V(X_t,k|I_t) = \sigma^2_{I_t,k}$ with $\Sigma = (\sigma^2_{i,k})$ known a priori.

Goal: Minimize expected regret $R_T = \max_{i \in [K]} \theta_i - \sum_{t=1}^T E[Y_t]$.
The Fishing Game
Choosing a fishing spot: \( K \) actions.
\[ \theta_1, \ldots, \theta_K : \text{(unknown) mean rewards for the } K \text{ spots.} \]
For rounds \( t = 1, \ldots, T \):

- Choose a fishing spot \( l_t \in [K] := \{1, \ldots, K\} \);
- Incur reward \( Y_t \in \mathbb{R} \) with mean \( \theta_{l_t} \);
- Observe \( X_t \in \mathbb{R}^K \); noisy reward observations for all the sites \( (Y_t = X_{t,l_t}) \).
The Fishing Game
Choosing a fishing spot: $K$ actions.

$\theta_1, \ldots, \theta_K$: (unknown) mean rewards for the $K$ spots.

For rounds $t = 1, \ldots, T$:

- Choose a fishing spot $I_t \in [K] := \{1, \ldots, K\}$;
- Incur reward $Y_t \in \mathbb{R}$ with mean $\theta_{I_t}$;
- Observe $X_t \in \mathbb{R}^K$; noisy reward observations for all the sites $(Y_t = X_{t,I_t})$.

Assumptions

$\mathbb{E}[X_{t,k}] = \theta_k$, and $\nabla(X_{t,k} | I_t) = \sigma^2_{I_t,k}$ with $\Sigma = (\sigma^2_{i,k})$ known a priori.
The Fishing Game
Choosing a fishing spot: \( K \) actions.

\( \theta_1, \ldots, \theta_K \): (unknown) mean rewards for the \( K \) spots.

For rounds \( t = 1, \ldots, T \):

- Choose a fishing spot \( l_t \in [K] := \{1, \ldots, K\} \);
- Incur reward \( Y_t \in \mathbb{R} \) with mean \( \theta_{l_t} \);
- Observe \( X_t \in \mathbb{R}^K \); noisy reward observations for all the sites \( (Y_t = X_{t,l_t}) \).

Assumptions

\[ \mathbb{E}[X_{t,k}] = \theta_k, \text{ and } \nabla(X_{t,k} | l_t) = \sigma^2_{l_t,k} \text{ with } \Sigma = (\sigma^2_{i,k}) \text{ known a priori.} \]

Goal

Minimize expected regret \( R_T = T \max_{i \in [K]} \theta_i - \sum_{t=1}^{T} \mathbb{E}[Y_t] \).
Outline

1. Introduction

2. Has This Been Done Before?

3. Results
   - Lower Bounds
   - Algorithms/Upper Bounds

4. Summary
For rounds $t = 1, \ldots, T$:

- Choose action $l_t \in [K]$;
- Observe $X_t \sim p(\theta, l_t)$;
- Incur reward $R_t = r(\theta, l_t)$. 

(Stochastic) Partial Monitoring and Bandits
For rounds $t = 1, \ldots, T$:

- Choose action $l_t \in [K]$;
- Observe $X_t \sim p(\theta, l_t)$;
- Incur reward $R_t = r(\theta, l_t)$.

**Information Structure**

- Known: $p : \Theta \times [K] \rightarrow \mathbb{M}_1(\mathcal{X})$;
- Known: $r : \Theta \times [K] \rightarrow \mathbb{R}$;
- Unknown: $\theta \in \Theta$. 

(Some) prior work:
- Bandits (Robbins, 1952): $X_t = R_t$.
- Finite $\Theta$ and $Y_t$, $1 = R_t$: Agrawal et al. (1989).
- $X_t = h(I_t, J_t), R_t = r(I_t, J_t)$, $J_t \in [M]$ i.i.d.: Bartók et al. (2011).
- Learning with feedback graphs: Alon et al. (2015).
(Stochastic) Partial Monitoring and Bandits

For rounds $t = 1, \ldots, T$:

- Choose action $l_t \in [K]$;
- Observe $X_t \sim p(\theta, l_t)$;
- Incur reward $R_t = r(\theta, l_t)$.

**Information Structure**

- Known: $p : \Theta \times [K] \rightarrow \mathbb{M}_1(X)$;
- Known: $r : \Theta \times [K] \rightarrow \mathbb{R}$;
- Unknown: $\theta \in \Theta$. 

(Some) prior work:

- Bandits (Robbins, 1952): $X_t = R_t$.
- Finite $\Theta$ and $Y_t, 1 = R_t$: Agrawal et al. (1989) $X_t = h(I_t, J_t)$, $R_t = r(I_t, J_t)$, $J_t \in [M]$ i.i.d.: Bartók et al. (2011)
- Learning with feedback graphs: Alon et al. (2015)
For rounds $t = 1, \ldots, T$:

- Choose action $l_t \in [K]$;
- Observe $X_t \sim p(\theta, l_t)$;
- Incur reward $R_t = r(\theta, l_t)$.

(Some) prior work:

- Bandits (Robbins, 1952): $X_t = R_t$. 

Information Structure

- Known: $p : \Theta \times [K] \rightarrow M_1(\mathcal{X})$;
- Known: $r : \Theta \times [K] \rightarrow \mathbb{R}$;
- Unknown: $\theta \in \Theta$. 

(Stochastic) Partial Monitoring and Bandits

For rounds $t = 1, \ldots, T$:

- Choose action $l_t \in [K]$;
- Observe $X_t \sim p(\theta, l_t)$;
- Incur reward $R_t = r(\theta, l_t)$.

Information Structure

- Known: $p : \Theta \times [K] \rightarrow \mathbb{M}_1(X)$;
- Known: $r : \Theta \times [K] \rightarrow \mathbb{R}$;
- Unknown: $\theta \in \Theta$.

(Some) prior work:

- Bandits (Robbins, 1952): $X_t = R_t$.
- Finite $\Theta$ and $Y_{t,1} = R_t$:
  Agrawal et al. (1989)
For rounds $t = 1, \ldots, T$:

- Choose action $l_t \in [K]$;
- Observe $X_t \sim p(\theta, l_t)$;
- Incur reward $R_t = r(\theta, l_t)$.

(Stochastic) Partial Monitoring and Bandits

Information Structure

- Known: $p : \Theta \times [K] \rightarrow \mathbb{M}_1(\mathcal{X})$;
- Known: $r : \Theta \times [K] \rightarrow \mathbb{R}$;
- Unknown: $\theta \in \Theta$.

(Some) prior work:

- Bandits (Robbins, 1952):
  $X_t = R_t$.
- Finite $\Theta$ and $Y_{t,1} = R_t$:
  Agrawal et al. (1989)
- $X_t = h(l_t, J_t)$, $R_t = r(l_t, J_t)$, $J_t \in [M]$ i.i.d.: Bartók et al. (2011)
For rounds $t = 1, \ldots, T$:

- Choose action $l_t \in [K]$;
- Observe $X_t \sim p(\theta, l_t)$;
- Incur reward $R_t = r(\theta, l_t)$.

### Information Structure

- Known: $p : \Theta \times [K] \to \mathbb{M}_1(\mathcal{X})$;
- Known: $r : \Theta \times [K] \to \mathbb{R}$;
- Unknown: $\theta \in \Theta$.

(Some) prior work:

- Bandits (Robbins, 1952): $X_t = R_t$.
- Finite $\Theta$ and $Y_{t,1} = R_t$: Agrawal et al. (1989)
- $X_t = h(I_t, J_t)$, $R_t = r(I_t, J_t)$, $J_t \in [M]$ i.i.d.: Bartók et al. (2011)
- Learning with feedback graphs: Alon et al. (2015)
Fishing as Partial Monitoring

Fishing round $t = 1, \ldots, T$:

- Choose a fishing spot $l_t \in [K]$;
- Incur (mean) reward $\theta_{l_t}$;
- Observe $X_t \in \mathbb{R}^K$.

**Basic Assumptions**

- $\mathbb{E}[X_{t,k}] = \theta_k$, $\mathbb{V}(X_{t,k}|l_t) = \sigma_{l_t,k}^2$ with
- $\Sigma = (\sigma_{i,k}^2)$ known a priori.

**Distributional Assumptions**

- $X_{t,j} \sim \mathcal{N}(\theta_j, \sigma_{l_t,j})$, independent.
Fishing as Partial Monitoring

Fishing round \( t = 1, \ldots, T \):
- Choose a fishing spot \( l_t \in [K] \);
- Incur (mean) reward \( \theta_{l_t} \);
- Observe \( X_t \in \mathbb{R}^K \).

Partial monitoring \( t = 1, \ldots, T \):
- Choose \( l_t \in [K] \);
- Incur (mean) reward \( r(\theta, l_t) \);
- Observe \( X_t \sim p(\theta, l_t) \).

Basic Assumptions
\[
\mathbb{E}[X_{t,k}] = \theta_k, \quad \mathbb{V}(X_{t,k}|l_t) = \sigma^2_{l_t,k} \text{ with } \Sigma = (\sigma^2_{i,k}) \text{ known a priori.}
\]

Distributional Assumptions
\( X_{t,j} \sim \mathcal{N}(\theta_j, \sigma_{l_t,j}) \), independent.
Fishing as Partial Monitoring

Fishing round $t = 1, \ldots, T$:
- Choose a fishing spot $l_t \in [K]$;
- Incur (mean) reward $\theta_{l_t}$;
- Observe $X_t \in \mathbb{R}^K$.

Basic Assumptions
\[ \mathbb{E}[X_{t,k}] = \theta_k, \quad \mathbb{V}(X_{t,k}|l_t) = \sigma_{l_t,k}^2 \quad \text{with} \quad \Sigma = (\sigma_{i,k}^2) \text{ known a priori.} \]

Distributional Assumptions
$X_{t,j} \sim \mathcal{N}(\theta_j, \sigma_{l_t,j})$, independent.

Partial monitoring $t = 1, \ldots, T$:
- Choose $l_t \in [K]$;
- Incur (mean) reward $r(\theta, l_t)$
- Observe $X_t \sim p(\theta, l_t)$.

Choose:
- $r(\theta, i) = \theta_i$;
- $p(\theta, i) = \mathcal{N}(\theta, \text{diag}(\ldots, \sigma_{i,j}, \ldots))$;
- $\Theta = [0, D]^K$. 
Some Interesting Special Cases

- Full information problems: $\sigma_{ij} = \sigma$ for all $i, j \in [K]$. 

Bandits:

- $\sigma_{ii} = \sigma$ for all $i \in [K]$, $\sigma_{ij} = \infty$ for all $i \neq j$.

Graph feedback (Alon et al., 2015):
- Each $i \in [K]$ has $S_i \subset [K]$:
  $$\sigma_{i, j} = \begin{cases} \sigma, & \text{if } j \in S_i \\ +\infty, & \text{otherwise} \end{cases}$$

Self-observability: $i \in S_i$ for any $i \in [K]$ (Mannor & Shamir, 2011; Caron et al., 2012; Alon et al., 2013; Buccapatnam et al., 2014; Koc´ak et al., 2014).

Strength: Our single model encompasses all these settings and allows continuous interpolation between them.
Some Interesting Special Cases

- Full information problems: \( \sigma_{ij} = \sigma \) for all \( i, j \in [K] \).
- Bandits: \( \sigma_{ii} = \sigma \) for all \( i \in [K] \), \( \sigma_{ij} = \infty \) for all \( i \neq j \).
Some Interesting Special Cases

- Full information problems: $\sigma_{ij} = \sigma$ for all $i, j \in [K]$.
- Bandits: $\sigma_{ii} = \sigma$ for all $i \in [K]$, $\sigma_{ij} = \infty$ for all $i \neq j$.
- Graph feedback (Alon et al., 2015):
Some Interesting Special Cases

- Full information problems: $\sigma_{ij} = \sigma$ for all $i, j \in [K]$.
- Bandits: $\sigma_{ii} = \sigma$ for all $i \in [K]$, $\sigma_{ij} = \infty$ for all $i \neq j$.
- Graph feedback (Alon et al., 2015):
  - Each $i \in [K]$ has $S_i \subset [K]$:
    \[
    \sigma_{i,j} = \begin{cases} 
    \sigma, & \text{if } j \in S_i; \\
    +\infty, & \text{otherwise}.
    \end{cases}
    \]
Some Interesting Special Cases

- Full information problems: \( \sigma_{ij} = \sigma \) for all \( i, j \in [K] \).
- Bandits: \( \sigma_{ii} = \sigma \) for all \( i \in [K] \), \( \sigma_{ij} = \infty \) for all \( i \neq j \).
- Graph feedback (Alon et al., 2015):
  - Each \( i \in [K] \) has \( S_i \subset [K] \):
    \[
    \sigma_{i,j} = \begin{cases} 
    \sigma, & \text{if } j \in S_i; \\
    +\infty, & \text{otherwise}. 
    \end{cases}
    \]
  - Self-observability: \( i \in S_i \) for any \( i \in [K] \) (Mannor & Shamir, 2011; Caron et al., 2012; Alon et al., 2013; Buccapatnam et al., 2014; Kocák et al., 2014).
Some Interesting Special Cases

- Full information problems: $\sigma_{ij} = \sigma$ for all $i, j \in [K]$.
- Bandits: $\sigma_{ii} = \sigma$ for all $i \in [K]$, $\sigma_{ij} = \infty$ for all $i \neq j$.
- Graph feedback (Alon et al., 2015):
  - Each $i \in [K]$ has $S_i \subset [K]$:
    \[
    \sigma_{i,j} = \begin{cases} 
    \sigma, & \text{if } j \in S_i; \\
    +\infty, & \text{otherwise}.
    \end{cases}
    \]
  - Self-observability: $i \in S_i$ for any $i \in [K]$ (Mannor & Shamir, 2011; Caron et al., 2012; Alon et al., 2013; Buccapatnam et al., 2014; Kocák et al., 2014).
Some Interesting Special Cases

- **Full information problems:** $\sigma_{ij} = \sigma$ for all $i, j \in [K]$.
- **Bandits:** $\sigma_{ii} = \sigma$ for all $i \in [K]$, $\sigma_{ij} = \infty$ for all $i \neq j$.
- **Graph feedback** (Alon et al., 2015):
  - Each $i \in [K]$ has $S_i \subset [K]$:
    \[
    \sigma_{i,j} = \begin{cases} 
    \sigma, & \text{if } j \in S_i; \\
    +\infty, & \text{otherwise}.
    \end{cases}
    \]
  - Self-observability: $i \in S_i$ for any $i \in [K]$ (Mannor & Shamir, 2011; Caron et al., 2012; Alon et al., 2013; Buccapatnam et al., 2014; Kocák et al., 2014).

**Strength:** Our single model encompasses all these settings and allows continuous interpolation between them.
How to Compare Algorithms?

### Performance Metric

Expected regret \( R_T = T \max_{i \in [K]} \theta_i - \sum_{t=1}^{T} \mathbb{E} [Y_t] \).
How to Compare Algorithms?

Performance Metric

Expected regret

\[ R_T = T \max_{i \in [K]} \theta_i - \sum_{t=1}^{T} \mathbb{E} [Y_t]. \]

Minimax Regret:

\[ R^*_T = \inf_A \sup_{\theta} R_T(A, \theta) \]
How to Compare Algorithms?

Performance Metric

Expected regret $R_T = T \max_{i \in [K]} \theta_i - \sum_{t=1}^{T} \mathbb{E}[Y_t]$.

Minimax Regret:

$$R_T^* = \inf_A \sup_{\theta} R_T(A, \theta)$$

Typically, $R_T^* = O(T^\alpha)$ with $0 < \alpha < 1$ (polynomial minimax regret), where the constant is a function of $(p, r), \Theta$, but not the individual $\theta$. 
How to Compare Algorithms?

**Performance Metric**

Expected regret $R_T = T \max_{i \in [K]} \theta_i - \sum_{t=1}^{T} \mathbb{E}[Y_t]$.

**Minimax Regret:**

$$R_T^* = \inf_A \sup_{\theta} R_T(A, \theta)$$

Typically, $R_T^* = O(T^\alpha)$ with $0 < \alpha < 1$ (polynomial minimax regret), where the constant is a function of $(p, r)$, $\Theta$, but not the individual $\theta$.

**Regret Asymptotics:**

$A_s = \text{set of algorithms with subpolynomial regret growth, i.e., for any } A \in A_s, \alpha > 0,$

$$R_T(A, \theta) = O(T^\alpha).$$
How to Compare Algorithms?

**Performance Metric**

Expected regret $R_T = T \max_{i \in [K]} \theta_i - \sum_{t=1}^{T} \mathbb{E}[Y_t]$. 

**Minimax Regret:**

$$R_T^* = \inf_{A} \sup_{\theta} R_T(A, \theta)$$

Typically, $R_T^* = O(T^\alpha)$ with $0 < \alpha < 1$ (polynomial minimax regret), where the constant is a function of $(p, r), \Theta$, but not the individual $\theta$.

**Regret Asymptotics:**

$A_s = \text{set of algorithms with subpolynomial regret growth, i.e., for any } A \in A_s, \alpha > 0,$

$$R_T(A, \theta) = O(T^\alpha).$$

**Problem-dependent sharp asymptotic regret lower bound:** For any $\theta \in \Theta$,

$$\inf_{A \in A_s} \liminf_{T \to \infty} \frac{R_T(A, \theta)}{\log(T)} = c(\theta).$$
Outline

1. Introduction

2. Has This Been Done Before?

3. Results
   - Lower Bounds
     - Algorithms/Upper Bounds

4. Summary
A Unified Lower Bound

Under our setting with general variance matrix $\Sigma$, we have a unified, finite-time, problem-dependent lower bound that recovers all of the existing results.
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter.

Regret:

\[
R_T^A(\theta) = \left\langle \bar{c}_{q_{\theta}}, \Delta(\theta) \right\rangle .
\]

- $\Delta_i(\theta) = \max_j \mu_j(\theta) - \mu_i(\theta)$ – the loss due to playing $i$ instead of an optimal action;
  - $\mu_i(\theta)$: the mean reward for action $i \in [K]$ under $\theta$.
- $\bar{c}_q = \int c \, dq(c) \in C_T^{\mathbb{R}^+}$: mean number of plays under $q \in M_1(C_T^N)$.
  - $C_T^S = \{c \in S^K : c_i \geq 0, \sum_{i \in [K]} c_i = T\}$
    - set of $S$-valued, $T$-round allocations.
- $q_{\theta} \in M_1(C_T^N)$: Distribution of $N_T \in C_T^N$, the number of pulls of the $K$ actions under $A$ and $\theta$. Depends on $A$ (dependence hidden).
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter. 

Regret:

$$R^A_T(\theta) = \langle \bar{c}_{q_\theta}, \Delta(\theta) \rangle.$$
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter.

Regret:

$$R^A_T(\theta) = \langle \bar{c}_q, \Delta(\theta) \rangle.$$ 

We want to lower bound this by a quantity that depends on $\theta, \Theta, T$, but not $A$. Only $0$ works if $A$ is allowed to be arbitrary (why?). Which algorithms to allow?

Ideas:

- Use the regret itself!
- Allow algorithms with some predetermined worst-case regret over $\Theta$!

[2000x2000]: $\sup_{\theta' \in \Theta} R^A_T(\theta') \leq B$ for $B > 0$ fixed.
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter.

Regret:

$$R^A_T(\theta) = \langle \bar{c}_{q_\theta}, \Delta(\theta) \rangle.$$ 

We want to lower bound this by a quantity that depends on $\theta, \Theta, T$, but not $A$.

Only 0 works if $A$ is allowed to be arbitrary (why?).
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter.

Regret:

$$ R^A_T(\theta) = \langle \bar{c}_q, \Delta(\theta) \rangle. $$

We want to lower bound this by a quantity that depends on $\theta, \Theta, T$, but not $A$.

Only 0 works if $A$ is allowed to be arbitrary (why?). Which algorithms to allow?
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter.

Regret:

$$ R^A_T(\theta) = \langle \bar{c}_{q_\theta}, \Delta(\theta) \rangle. $$

We want to lower bound this by a quantity that depends on $\theta, \Theta, T$, but not $A$.

Only 0 works if $A$ is allowed to be arbitrary (why?). Which algorithms to allow?

Ideas:

- Use the regret itself! Allow algorithms with some predetermined worst-case regret over $\Theta$!
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter.

Regret:

$$R_T^A(\theta) = \langle \bar{c}_{q_{\theta}}, \Delta(\theta) \rangle.$$ 

We want to lower bound this by a quantity that depends on $\theta, \Theta, T$, but not $A$.

Only 0 works if $A$ is allowed to be arbitrary (why?). Which algorithms to allow?

Ideas:

- Use the regret itself! Allow algorithms with some predetermined worst-case regret over $\Theta!$ : $\sup_{\theta' \in \Theta} R_T^A(\theta') \leq B$ for $B > 0$ fixed.
Idea of the Lower Bound

Let $A$ be an algorithm, $\theta \in \Theta$ an environment parameter.

Regret:

$$R^A_T(\theta) = \langle \bar{c}_{q_\theta}, \Delta(\theta) \rangle.$$  

We want to lower bound this by a quantity that depends on $\theta, \Theta, T$, but not $A$.

Only 0 works if $A$ is allowed to be arbitrary (why?). Which algorithms to allow?

Ideas:

- Use the regret itself! Allow algorithms with some predetermined worst-case regret over $\Theta! : \sup_{\theta' \in \Theta} R^A_T(\theta') \leq B$ for $B > 0$ fixed.
- General strategy (for any lower bounds): create perturbations of $\theta$ s.t. any algorithm performs ”badly” on one of them.
Asymptotic Lower Bound for Graph Feedback

Derived from the work of Graves & Lai (1997):

- Let $\Delta_i = \max_j \theta_j - \theta_i$; $\sigma_{i,j} \in \{\sigma, +\infty\}$. Assumption: optimal action is unique; let $i_1, i_2$ be the index of the best, resp., second best action.
Asymptotic Lower Bound for Graph Feedback

Derived from the work of Graves & Lai (1997):

- Let $\Delta_i = \max_j \theta_j - \theta_i$; $\sigma_{i,j} \in \{\sigma, +\infty\}$. Assumption: optimal action is unique; let $i_1, i_2$ be the index of the best, resp., second best action.

**Theorem (Asymptotic lower bound)**

For any algorithm $A \in \mathcal{A}_s$, and for any $\theta \in \Theta$,

$$\liminf_{T \to \infty} \frac{R_T(A, \theta)}{\log T} \geq \inf_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i,$$

where

$$C_\theta = \left\{ c \in [0, \infty)^K : \sum_{i:j \in S_i} c_i \geq \frac{2\sigma^2}{\Delta_j^2} \quad \text{for all } j \neq i_1, \quad \text{and} \quad \sum_{i:i_1 \in S_i} c_i \geq \frac{2\sigma^2}{\Delta_{i_2}^2} \right\}.$$
Lower Bound for Gaussian Case

Given some $B > 0$, for $i \neq i_1$, let

$$
\epsilon_i = \frac{8\sqrt{eB}}{T} e^{W\left(\frac{\Delta_i T}{16\sqrt{eB}}\right)} + \Delta_i, \quad m_i(\theta, B) = \frac{1}{\epsilon_i^2} \log \frac{T(\epsilon_i - \Delta_i)}{8B}.
$$

For $i = i_1$, replace $\Delta_i$ with $\Delta_{i_2}$. Let

$$
C_{\theta, B} = \left\{ c \in \mathbb{C}^{\mathbb{R}^+} : \sum_{j=1}^{K} \frac{c_j}{\sigma_{jj}^2} \geq m_i(\theta, B) \text{ for all } i \in [K] \right\}.
$$

\[W(.)\] is the Lambert W function satisfying $W(x)e^{W(x)} = x.$
Lower Bound for Gaussian Case

Given some $B > 0$, for $i \neq i_1$, let

$$
\epsilon_i = \frac{8\sqrt{eB}}{T} \text{e}^{W\left(\frac{\Delta_i T}{16\sqrt{eB}}\right)} + \Delta_i , \quad m_i(\theta, B) = \frac{1}{\epsilon_i^2} \log \frac{T(\epsilon_i - \Delta_i)}{8B} .
$$

For $i = i_1$, replace $\Delta_i$ with $\Delta_{i_2}$. Let

$$
C_{\theta, B} = \left\{ c \in C_T^{\mathbb{R}^+} : \sum_{j=1}^{K} \frac{c_j}{\sigma_{ji}^2} \geq m_i(\theta, B) \text{ for all } i \in [K] \right\} .
$$

Theorem (Finite-time problem-dependent lower bound)

For any algorithm s.t. $\sup_{\lambda \in \Theta} R_T(\lambda) \leq B$, any $T$ large enough, any $\theta$ inside $\Theta$,

$$
R_T(\theta) \geq b(\theta, B) = \min_{c \in C_{\theta, B}} \sum_{i \neq i_1} c_i \Delta_i .
$$

$W(\cdot)$ is the Lambert W function satisfying $W(x)e^{W(x)} = x$. 
Theorem (Finite-time problem-dependent lower bound)

For any algorithm such that \( \sup_{\lambda \in \Theta} R_T(\lambda) \leq B \), we have, for any \( \theta \in \Theta \),

\[
R_T(\theta) \geq b(\theta, B) = \min_{c \in C_{\theta, B}} \sum_{i \neq i_1} c_i \Delta_i .
\]  \( (*) \)
Recovering the Asymptotic Lower Bound

Theorem (Finite-time problem-dependent lower bound)

For any algorithm such that \( \sup_{\lambda \in \Theta} R_T(\lambda) \leq B \), we have, for any \( \theta \in \Theta \),

\[
R_T(\theta) \geq b(\theta, B) = \min_{c \in C_{\theta, B}} \sum_{i \neq i_1} c_i \Delta_i .
\]  

(*)

Recall asymptotic lower bound:

\[
\liminf_{T \to \infty} \frac{R_T(\theta)}{\log T} \geq \inf_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i .
\]  

(**)
Recovering the Asymptotic Lower Bound

Theorem (Finite-time problem-dependent lower bound)

For any algorithm such that \( \sup_{\lambda \in \Theta} R_T(\lambda) \leq B \), we have, for any \( \theta \in \Theta \),

\[
R_T(\theta) \geq b(\theta, B) = \min_{c \in C_{\theta,B}} \sum_{i \neq i_1} c_i \Delta_i .
\]

(\(*\))

- Recall asymptotic lower bound:

\[
\liminf_{T \to \infty} \frac{R_T(\theta)}{\log T} \geq \inf_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i .
\]

(\(**\))

- For any \( B = \alpha T^\beta \) with \( \alpha > 0 \) and \( \beta \in (0, 1) \) we have

\[
C_{\theta,B} \to \frac{(1 - \beta) \log T}{2} C_\theta
\]

as \( T \to \infty \). Hence, (\(**\)) is recovered from (\(*\)).
Minimax Lower Bounds (Alon et al., 2015)

Each \( i \in [K] \) is associated with an observation set \( S_i \subset [K] \): for \( j \in S_i \), \( \sigma_{ij} = \sigma \); for \( j \notin S_i \), \( \sigma_{ij} = \infty \).
Minimax Lower Bounds (Alon et al., 2015)

Each $i \in [K]$ is associated with an observation set $S_i \subset [K]$: for $j \in S_i$, $\sigma_{ij} = \sigma$; for $j \notin S_i$, $\sigma_{ij} = \infty$.

- Assume $\Sigma$ is always observable: for all $i$, there exists $j$ such that $i \in S_j$. 

- $\Sigma$ is strongly observable if all actions are strongly observable.

- An action $i$ is strongly observable if either it is self-observable or is observable under any other action. Otherwise, the action is said to be weakly observable.

- $\Sigma$ is weakly observable if it is observable but not strongly observable.
Minimax Lower Bounds (Alon et al., 2015)

Each $i \in [K]$ is associated with an observation set $S_i \subset [K]$: for $j \in S_i$, $\sigma_{ij} = \sigma$; for $j \notin S_i$, $\sigma_{ij} = \infty$.

- Assume $\Sigma$ is always observable: for all $i$, there exists $j$ such that $i \in S_j$.
- $\Sigma$ is strongly observable if all actions are strongly observable.
Minimax Lower Bounds (Alon et al., 2015)

Each $i \in [K]$ is associated with an observation set $S_i \subset [K]$: for $j \in S_i$, $\sigma_{ij} = \sigma$; for $j \notin S_i$, $\sigma_{ij} = \infty$.

- Assume $\Sigma$ is always observable: for all $i$, there exists $j$ such that $i \in S_j$.
- $\Sigma$ is strongly observable if all actions are strongly observable.
  - An action $i$ is strongly observable if either it is self-observable or is observable under any other action. Otherwise, the action is said to be weakly observable.
Each $i \in [K]$ is associated with an observation set $S_i \subset [K]$: for $j \in S_i$, $\sigma_{ij} = \sigma$; for $j \notin S_i$, $\sigma_{ij} = \infty$.

- Assume $\Sigma$ is always observable: for all $i$, there exists $j$ such that $i \in S_j$.
- $\Sigma$ is strongly observable if all actions are strongly observable.
  - An action $i$ is strongly observable if either it is self-observable or is observable under any other action. Otherwise, the action is said to be weakly observable.
Minimax Lower Bounds (Alon et al., 2015)

Each \( i \in [K] \) is associated with an observation set \( S_i \subset [K] \): for \( j \in S_i \), \( \sigma_{ij} = \sigma \); for \( j \notin S_i \), \( \sigma_{ij} = \infty \).

- Assume \( \Sigma \) is always observable: for all \( i \), there exists \( j \) such that \( i \in S_j \).
- \( \Sigma \) is strongly observable if all actions are strongly observable.
  - An action \( i \) is strongly observable if either it is self-observable or is observable under any other action. Otherwise, the action is said to be weakly observable.
- \( \Sigma \) is weakly observable if it is observable but not strongly observable.
Minimax Lower Bounds for Graph Feedback - Strong Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$. 
Minimax Lower Bounds for Graph Feedback - Strong Observability

- \( \sigma_{i,j} \in \{1, +\infty\} \), \( \Theta = [0, 1] \); \( S_i = \{ j : \sigma_{i,j} = \sigma \} \).
- A set \( A \subset [K] \) is independent in \( \Sigma \) if for any \( i \in A \), \( S_i \cap A \subset \{i\} \).
Minimax Lower Bounds for Graph Feedback - Strong Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$.
- A set $A \subset [K]$ is independent in $\Sigma$ if for any $i \in A$, $S_i \cap A \subset \{i\}$.
  - Choosing $i \in A$ gives no information about any $j \neq i, j \in A$. 

**Independence number of $\Sigma$:**

$\kappa(\Sigma) = \max \{|A| : A \subset [K] \text{ is independent in } \Sigma\}$. 

**Theorem (Mannor & Shamir (2011), Alon et al. (2015))**

Let $\Sigma$ be strongly observable. Then,

$\sup_{\theta \in \Theta} R_T(\theta) \geq c \sqrt{\kappa(\Sigma)} T$. 

18 / 33
Minimax Lower Bounds for Graph Feedback - Strong Observability

- \( \sigma_{i,j} \in \{1, +\infty\} \), \( \Theta = [0, 1] \); \( S_i = \{ j : \sigma_{i,j} = \sigma \} \).

- A set \( A \subset [K] \) is independent in \( \Sigma \) if for any \( i \in A \), \( S_i \cap A \subset \{i\} \).
  - Choosing \( i \in A \) gives no information about any \( j \neq i, j \in A \).

- Independence number of \( \Sigma \):

\[
\kappa(\Sigma) = \max\{|A| : A \subset [K] \text{ is independent in } \Sigma\}.
\]
Minimax Lower Bounds for Graph Feedback - Strong Observability

- $\sigma_{i,j} \in \{1, +\infty\}, \Theta = [0, 1]; S_i = \{j : \sigma_{i,j} = \sigma\}.$
- A set $A \subset [K]$ is independent in $\Sigma$ if for any $i \in A, S_i \cap A \subset \{i\}.$
  - Choosing $i \in A$ gives no information about any $j \neq i, j \in A.$
- Independence number of $\Sigma$:

\[
\kappa(\Sigma) = \max\{|A| : A \subset [K] \text{ is independent in } \Sigma\}.
\]
Minimax Lower Bounds for Graph Feedback - Strong Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$.
- A set $A \subset [K]$ is independent in $\Sigma$ if for any $i \in A$, $S_i \cap A \subset \{i\}$.
  - Choosing $i \in A$ gives no information about any $j \neq i, j \in A$.
- Independence number of $\Sigma$:

$$\kappa(\Sigma) = \max\{|A| : A \subset [K] \text{ is independent in } \Sigma\}.$$

**Theorem (Mannor & Shamir (2011), Alon et al. (2015))**

*Let $\Sigma$ be strongly observable. Then,*

$$\sup_{\theta \in \Theta} R_T(\theta) \geq c \sqrt{\kappa(\Sigma) T}.$$
Minimax Lower Bounds for Graph Feedback - Weak Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$;
Minimax Lower Bounds for Graph Feedback - Weak Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$;
- $A, A' \subset [K]$; $A$ dominates $A'$ if for any $j \in A'$ there exists $i \in A$ such that $j \in S_i$;
Minimax Lower Bounds for Graph Feedback - Weak Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$;
- $A, A' \subset [K]$; $A$ dominates $A'$ if for any $j \in A'$ there exists $i \in A$ such that $j \in S_i$;
  - Any $j \in A'$ can be observed through some $i \in A$. 

$W(\Sigma)$: Set of all weakly observable actions; Weak domination number: $\rho(\Sigma) = \min \{|A| : A \text{ dominates } W(\Sigma)\}$.

Theorem (Mannor & Shamir (2011), Alon et al. (2015))

Let $\Sigma$ be weakly observable. Then,

$$\sup_{\theta \in \Theta} R_T(\theta) \geq c(\log K)^{1/3} - 2/3^{2/3} \rho(\Sigma)^{1/3}.$$
Minimax Lower Bounds for Graph Feedback - Weak Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$;
- $A, A' \subset [K]$; $A$ dominates $A'$ if for any $j \in A'$ there exists $i \in A$ such that $j \in S_i$;
  - Any $j \in A'$ can be observed through some $i \in A$.
- $\mathcal{W}(\Sigma)$: Set of all weakly observable actions;
Minimax Lower Bounds for Graph Feedback - Weak Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$;
- $A, A' \subset [K]$; $A$ dominates $A'$ if for any $j \in A'$ there exists $i \in A$ such that $j \in S_i$;
  - Any $j \in A'$ can be observed through some $i \in A$.
- $\mathcal{W}(\Sigma)$: Set of all weakly observable actions;
- Weak domination number:

\[
\rho(\Sigma) = \min\{|A| : A \text{ dominates } \mathcal{W}(\Sigma)\}.
\]
Minimax Lower Bounds for Graph Feedback - Weak Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$;
- $A, A' \subset [K]$; $A$ dominates $A'$ if for any $j \in A'$ there exists $i \in A$ such that $j \in S_i$;
  - Any $j \in A'$ can be observed through some $i \in A$.
- $\mathcal{W}(\Sigma)$: Set of all weakly observable actions;
- *Weak domination number:*

$$\rho(\Sigma) = \min\{|A| : A \text{ dominates } \mathcal{W}(\Sigma)\}.$$
Minimax Lower Bounds for Graph Feedback - Weak Observability

- $\sigma_{i,j} \in \{1, +\infty\}$, $\Theta = [0, 1]$; $S_i = \{j : \sigma_{i,j} = \sigma\}$;
- $A, A' \subset [K]$; $A$ dominates $A'$ if for any $j \in A'$ there exists $i \in A$ such that $j \in S_i$;
  - Any $j \in A'$ can be observed through some $i \in A$.
- $\mathcal{W}(\Sigma)$: Set of all weakly observable actions;
- Weak domination number:

$$\rho(\Sigma) = \min\{|A| : A \text{ dominates } \mathcal{W}(\Sigma)\}.$$ 

**Theorem (Mannor & Shamir (2011), Alon et al. (2015))**

Let $\Sigma$ be weakly observable. Then,

$$\sup_{\theta \in \Theta} R_T(\theta) \geq c (\log K)^{-2/3} \rho(\Sigma)^{1/3} T^{2/3}.$$
### Recovering Minimax Lower Bounds

#### Theorem (Finite-time problem-dependent lower bound)

For any algorithm such that $\sup_{\lambda \in \Theta} R_T(\lambda) \leq B$, we have, for any $\theta \in \Theta$,

$$R_T(\theta) \geq b(\theta, B) = \min_{c \in C_{\theta, B}} \sum_{i \neq i_1} c_i \Delta_i.$$  \hfill (*)&
Recovering Minimax Lower Bounds

Theorem (Finite-time problem-dependent lower bound)

For any algorithm such that $\sup_{\lambda \in \Theta} R_T(\lambda) \leq B$, we have, for any $\theta \in \Theta$,

$$R_T(\theta) \geq b(\theta, B) = \min_{c \in C_{\theta, B}} \sum_{i \neq i_1} c_i \Delta_i. \quad (\ast)$$

- If $\Sigma$ is strongly observable, by choosing
  $$B = \frac{\sigma \sqrt{\kappa(\Sigma) T}}{8 \sqrt{e}}$$

  we have $\sup_{\theta \in \Theta} b(\theta, B) \geq B$ for $T$ large enough.
Recovering Minimax Lower Bounds

Theorem (Finite-time problem-dependent lower bound)

For any algorithm such that \( \sup_{\lambda \in \Theta} R_T(\lambda) \leq B \), we have, for any \( \theta \in \Theta \),

\[
R_T(\theta) \geq b(\theta, B) = \min_{c \in C_{\theta, B}} \sum_{i \neq i_1} c_i \Delta_i .
\]  

(*)

- If \( \Sigma \) is strongly observable, by choosing

\[
B = \frac{\sigma \sqrt{\kappa(\Sigma) T}}{8 \sqrt{e}}
\]

we have \( \sup_{\theta \in \Theta} b(\theta, B) \geq B \) for \( T \) large enough.

- If \( \Sigma \) is weakly observable, by choosing

\[
B = \frac{\left( \rho(\Sigma) D \right)^{1/3} \left( \sigma T \right)^{2/3}}{73 \left( \log K \right)^{2/3}}
\]

we have \( \sup_{\theta \in \Theta} b(\theta, B) \geq B \).
Outline

1 Introduction

2 Has This Been Done Before?

3 Results
   - Lower Bounds
   - Algorithms/Upper Bounds

4 Summary
Upcoming Attractions

- Just for feedback graphs;
- Near asymptotically optimal algorithm (new);
- *Single* near-minimax optimal algorithm – with logarithmic asymptotic regret (new).
Asymptotically (Almost) Optimal Algorithm

Recall

\[ C_\theta = \left\{ c \in [0, \infty)^K : \sum_{i:j \in S_i} c_i \geq \frac{2\sigma^2}{\Delta_j^2} \text{ for all } j \neq i_1, \text{ and } \sum_{i:i_1 \in S_i} c_i \geq \frac{2\sigma^2}{\Delta_{i_2}^2} \right\} . \]

Let \( c(\theta) = \arg\min_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i \).
Asymptotically (Almost) Optimal Algorithm

Recall

\[ C_\theta = \left\{ c \in [0, \infty)^K : \sum_{i:j \in S_i} c_i \geq \frac{2\sigma^2}{\Delta^2_j} \text{ for all } j \neq i_1, \text{ and } \sum_{i:i_1 \in S_i} c_i \geq \frac{2\sigma^2}{\Delta^2_{i_2}} \right\} . \]

Let \( c(\theta) = \arg\min_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i. \)

Goal: Find an algorithm that achieves \( O\left( (\sum_{i \neq i_1} c_i(\theta)\Delta_i) \log T \right) \) regret.

(Simple) idea borrowed from Magureanu et al. (2014):
Use forced exploration to ensure that \( c(\theta) \) is well-approximated by \( c(\hat{\theta}_t) \) uniformly in time, while paying a constant price in total.

Exploration schedule \( \beta(\cdot) : \mathbb{N} \rightarrow \mathbb{R} \) is chosen to be sublinear.

\( \Rightarrow \) Magureanu et al. (2014)'s linear schedule \( \beta(n) = \beta n \) requires that they choose a parameter of their algorithm based on the unknown \( \Delta_{\min} \).

The sublinear schedule avoids this.
Asymptotically (Almost) Optimal Algorithm

Recall

\[ C_\theta = \left\{ c \in [0, \infty)^K : \sum_{i:j \in S_i} c_i \geq \frac{2\sigma^2}{\Delta^2_j} \text{ for all } j \neq i_1, \text{ and } \sum_{i:i_1 \in S_i} c_i \geq \frac{2\sigma^2}{\Delta^2_{i_2}} \right\}. \]

Let \( c(\theta) = \arg\min_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i \).

Goal: Find an algorithm that achieves \( O((\sum_{i \neq i_1} c_i(\theta) \Delta_i) \log T) \) regret.

(Simple) idea borrowed from Magureanu et al. (2014):

- Use forced exploration to ensure that \( c(\theta) \) is well-approximated by \( c(\hat{\theta}_t) \) uniformly in time, while paying a constant price in total.
Asymptotically (Almost) Optimal Algorithm

Recall

\[ C_\theta = \left\{ c \in [0, \infty)^K : \sum_{i:j \in S_i} c_i \geq \frac{2\sigma^2}{\Delta^2_j} \text{ for all } j \neq i_1, \text{ and } \sum_{i:i_1 \in S_i} c_i \geq \frac{2\sigma^2}{\Delta^2_{i_2}} \right\}. \]

Let \( c(\theta) = \arg\min_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i. \)

Goal: Find an algorithm that achieves \( O((\sum_{i \neq i_1} c_i(\theta)\Delta_i) \log T) \) regret.

(Simple) idea borrowed from Magureanu et al. (2014):

- Use forced exploration to ensure that \( c(\theta) \) is well-approximated by \( c(\hat{\theta}_t) \) uniformly in time, while paying a constant price in total.
- Exploration schedule \( \beta(\cdot) : \mathbb{N} \to \mathbb{R} \) is chosen to be sublinear.
Asymptotically (Almost) Optimal Algorithm

Recall

\[ C_\theta = \left\{ c \in [0, \infty)^K : \sum_{i:j \in S_i} c_i \geq \frac{2\sigma^2}{\Delta_j^2} \text{ for all } j \neq i_1, \text{ and } \sum_{i:i_1 \in S_i} c_i \geq \frac{2\sigma^2}{\Delta_{i_1}^2} \right\} \]

Let \( c(\theta) = \arg\min_{c \in C_\theta} \sum_{i \neq i_1} c_i \Delta_i \).

Goal: Find an algorithm that achieves \( O((\sum_{i \neq i_1} c_i(\theta) \Delta_i) \log T) \) regret.

(Simple) idea borrowed from Magureanu et al. (2014):

- Use forced exploration to ensure that \( c(\theta) \) is well-approximated by \( c(\hat{\theta}_t) \) uniformly in time, while paying a constant price in total.
- Exploration schedule \( \beta(\cdot) : \mathbb{N} \to \mathbb{R} \) is chosen to be sublinear.
  - Magureanu et al. (2014)’s linear schedule \( \beta(n) = \beta n \) requires that they choose a parameter of their algorithm based on the unknown \( \Delta_{\min} \). The sublinear schedule avoids this.
Asymptotically (Almost) Optimal Algorithm -

$t := t + 1$

**Y:**

- Exploitation:
  - Play $I_t := i_1(\hat{\theta}_t)$.
  - Set $n_e(t + 1) := n_e(t)$.

- $\frac{\text{plays}(t)}{4\alpha \log t} \in C_{\hat{\theta}_t}'$?

  **Y:**
  - Play $I_t$ s.t. $\arg\min_i \text{obs}_i(t) \in S_{I_t}$.

  **N:**
  - Play $I_t = i$ s.t. $\text{plays}_i(t) < c_i(\hat{\theta}_t)4\alpha \log t$

- Set $n_e(t + 1) = n_e(t) + 1$.

- Update $\hat{\theta}_t$ to $\hat{\theta}_{t+1}$.

**N:**

- $\min_i \text{obs}_i(t) < \beta(n_e(t))/K$?

  **Y:**
  - Set $n_e(t + 1) = n_e(t) + 1$.

  **N:**
  - Update $\hat{\theta}_t$ to $\hat{\theta}_{t+1}$.
Asymptotically Almost Optimal Algorithm - Upper Bound

Upper bound

For any $\alpha > 2$, $\beta(n) = an^b$ with $a \in (0, \frac{1}{2}]$, $b \in (0, 1)$ and for any $\theta \in \Theta$ such that $c(\theta)$ is unique,

$$\limsup_{T \to \infty} \frac{R_T(\theta)}{\log T} \leq 4\alpha \sum_{i \neq i_1} c_i(\theta) \Delta_i.$$
Near Minimax Optimal Algorithm

**Successive elimination:** maintain a set of possibly optimal actions ("good" actions) until only one action remains.
Near Minimax Optimal Algorithm

Successive elimination: maintain a set of possibly optimal actions ("good" actions) until only one action remains.

In each round $r$,

- Explore all “good actions” by playing only “good actions”. (exploitation)
- Due to weak observability, sometimes some actions can only be explored by “bad actions” (exploration-exploitation trade off).
- Use a sublinear function $\gamma$ to control the exploration using “bad actions”.
**Near Minimax Optimal Algorithm**

**Successive elimination:** maintain a set of possibly optimal actions ("good" actions) until only one action remains.

In each round $r$,

- Explore all "good actions" by playing only "good actions". (exploitation)
- Due to weak observability, sometimes some actions can only be explored by "bad actions" (exploration-exploitation trade off).
- Use a sublinear function $\gamma$ to control the exploration using "bad actions".

The idea is similar to the CBP algorithm in Bartók et al. (2014). Here we use a better exploration method to exploit the feedback structure, which leads to the optimal dependence on factors such as $\rho(\Sigma)$ and $\kappa(\Sigma)$. 
For $E, G \subset [K]$, let $c(E, G) = \arg \max_{c \in \text{Simplex}_{|E|}} \min_{i \in G} \sum_{j : i \in S_j} c_j$: optimal way of using actions in $E$ to uniformly explore actions in $G$.

$c(E, G) = \min_{i \in G} \sum_{j : i \in S_j} c_j(E, G)$: least coverage.

For any $A \subset [K]$ and $|A| \geq 2$,

- let $A^S = \{i \in A : \exists j \in A, i \in S_j\}$ denote the set of actions that can be observed while using of actions of $A$ only;
- and $A^W = A \setminus A^S$ (Note: actions in $A^W$ must be weakly observable).

- Exploration schedule for $A^W$: $\gamma(r) = \left(\sigma \alpha_r t_r / D\right)^{2/3}$
- $\alpha_r = \min_{1 \leq s \leq r, A^W_s \neq \emptyset} c([K] \setminus A^W_s)$

At round $r$, define confidence width $g_{r,i}(\delta) = \sigma \sqrt{\frac{2 \log(8K^2r^3/\delta)}{\text{obs}_i(r)}}$ where $\text{obs}_i(r)$ is the number of observations gained for action $i$ so far.
Near Minimax Optimal Algorithm

$r := r + 1$

$A_r^w \neq \emptyset$ & $\text{obs}_{A_r^w}(r) < \text{obs}_{A_r^s}(r)$ & $\text{obs}_{A_r^w}(r) < \gamma(r)$ ?

$Y$

$c_r = c([K], A_r^w)$

Play $i_r = \lceil c_r \cdot \|c_r\|_0 \rceil$

Set $t_{r+1} \leftarrow t_r + \|i_r\|_1$

Update $\hat{\theta}_r$ to $\hat{\theta}_{r+1}$

$A_{r+1} \leftarrow \{i \in A_r : \text{UCB}_{r+1,i} \geq \max_{j \in A_r} \text{LCB}_{r+1,j}\}$

$N$

Keep playing the remaining action

$\text{Set } t_{r+1} \leftarrow t_r + \|i_r\|_1$

$\text{Update } \hat{\theta}_r$ to $\hat{\theta}_{r+1}$

$A_{r+1} \leftarrow \{i \in A_r : \text{UCB}_{r+1,i} \geq \max_{j \in A_r} \text{LCB}_{r+1,j}\}$

$|A_{r+1}| > 1$ ?

$Y$

Keep playing the remaining action

$N$
Theorem

With $\delta = \frac{1}{T}$, for any $\theta \in \Theta$:

- If $\Sigma$ is strongly observable,

  \[ R_T(\theta) = O \left( \sigma \log K \sqrt{\kappa(\Sigma) T \log T} \right). \]

- If $\Sigma$ is weakly observable,

  \[ R_T(\theta) = O \left( (\rho(\Sigma) D)^{1/3} (\sigma T)^{2/3} \cdot \sqrt{\log KT} \right). \]

- If we view $\Delta_{\text{min}}$ as constant and only consider dependence on $T$,

  \[ R_T(\theta) = O \left( \log^{3/2} T \right). \]
Outline

1 Introduction

2 Has This Been Done Before?

3 Results
   - Lower Bounds
   - Algorithms/Upper Bounds

4 Summary
Conclusions

- Online learning with Gaussian payoffs and side observations;
- Smooth interpolation between full-information and bandit settings;
- First non-asymptotic, problem-dependent lower bounds in regret minimization;
Conclusions

- Online learning with Gaussian payoffs and side observations;
- Smooth interpolation between full-information and bandit settings;
- First non-asymptotic, problem-dependent lower bounds in regret minimization;
- Algorithms for $\sigma_{i,j} \in \{\sigma, +\infty\}$;

\begin{itemize}
  \item Asymptotically near-optimal algorithm;
  \item First for learning with feedback graphs to do this;
  \item Single near minimax algorithm regardless of observability, with poly-logarithmic asymptotic regret;
\end{itemize}

\* Mannor & Shamir (2011); Alon et al. (2013) and Alon et al. (2015):
  \* No log asymptotic regret, minimax loss.

\* Caron et al. (2012) and Buccapatnam et al. (2014):
  \* Log asymptotics, but no near-minimax finite time regret.
Conclusions

- Online learning with Gaussian payoffs and side observations;
- Smooth interpolation between full-information and bandit settings;
- First non-asymptotic, problem-dependent lower bounds in regret minimization;
- Algorithms for $\sigma_{i,j} \in \{\sigma, +\infty\}$;
  - Asymptotically near-optimal algorithm;
    - First for learning with feedback graphs to do this;
Conclusions

- Online learning with Gaussian payoffs and side observations;
- Smooth interpolation between full-information and bandit settings;
- First non-asymptotic, problem-dependent lower bounds in regret minimization;
- Algorithms for $\sigma_{i,j} \in \{\sigma, +\infty\}$;
  - Asymptotically near-optimal algorithm;
    - First for learning with feedback graphs to do this;
  - Single near minimax algorithm regardless of observability, with poly-logarithmic asymptotic regret;
    - First for learning with feedback graphs to do this:
Conclusions

- Online learning with Gaussian payoffs and side observations;
- Smooth interpolation between full-information and bandit settings;
- First non-asymptotic, problem-dependent lower bounds in regret minimization;
- Algorithms for $\sigma_{i,j} \in \{\sigma, +\infty\}$;
  - Asymptotically near-optimal algorithm;
    - First for learning with feedback graphs to do this;
  - Single near minimax algorithm regardless of observability, with poly-logarithmic asymptotic regret;
    - First for learning with feedback graphs to do this:
    - Mannor & Shamir (2011); Alon et al. (2013) and Alon et al. (2015): No log asymptotic regret, minimax alss.
Conclusions

- Online learning with Gaussian payoffs and side observations;
- Smooth interpolation between full-information and bandit settings;
- First non-asymptotic, problem-dependent lower bounds in regret minimization;
- Algorithms for $\sigma_{i,j} \in \{\sigma, +\infty\}$;
  - Asymptotically near-optimal algorithm;
    - First for learning with feedback graphs to do this;
  - **Single** near minimax algorithm regardless of observability, with poly-logarithmic asymptotic regret;
    - First for learning with feedback graphs to do this:
      - Mannor & Shamir (2011); Alon et al. (2013) and Alon et al. (2015): No log asymptotic regret, minimax all.
      - Caron et al. (2012) and Buccapatnam et al. (2014): Log asymptotics, but no near-minimax finite time regret.
Open Problems

- Remove the assumption that $c(\theta)$ is unique for the optimality of the first algorithm;
- Remove the $\log^{1/2} T$ overhead for the second algorithm;
Open Problems

- Remove the assumption that $c(\theta)$ is unique for the optimality of the first algorithm;
- Remove the $\log^{1/2} T$ overhead for the second algorithm;
- A single algorithm that achieves both asymptotic and minimax optimal bounds up to constant factors;
  - For bandits, achieved (very) recently (Lattimore, 2015)
Open Problems

- Remove the assumption that $c(\theta)$ is unique for the optimality of the first algorithm;
- Remove the $\log^{1/2} T$ overhead for the second algorithm;
- A single algorithm that achieves both asymptotic and minimax optimal bounds up to constant factors;
  - For bandits, achieved (very) recently (Lattimore, 2015)
- Algorithm for general $\Sigma$;
- Algorithm for unknown $\Sigma$;
Open Problems

- Remove the assumption that $c(\theta)$ is unique for the optimality of the first algorithm;
- Remove the $\log^{1/2} T$ overhead for the second algorithm;
- A single algorithm that achieves both asymptotic and minimax optimal bounds up to constant factors;
  - For bandits, achieved (very) recently (Lattimore, 2015)
- Algorithm for general $\Sigma$;
- Algorithm for unknown $\Sigma$;
- General tightness of the new lower bound;
- Algorithms for the (general) stochastic partial monitoring setting.
References

Agrawal, R., Teneketzis, D., and Anantharam, V. Asymptotically efficient adaptive allocation schemes for controlled i.i.d. processes: Finite parameter space. *IEEE Transaction on Automatic Control*, 34:258–267, 1989.

Alon, N., Cesa-Bianchi, N., Gentile, C., and Mansour, Y. From bandits to experts: A tale of domination and independence. In *NIPS*, pp. 1610–1618, 2013.

Alon, N., Cesa-Bianchi, N., Gentile, C., and Mansour, Y. Online learning with feedback graphs: beyond bandits. In *COLT*, pp. 23–35, 2015.

Bartók, G., Pál, D., and Szepesvári, Cs. Minimax regret of finite partial-monitoring games in stochastic environments. In *COLT 2011*, pp. 133–154, July 2011.

Bartók, Gábor, Foster, Dean P., Pál, Dávid, Rakhlin, Alexander, and Szepesvári, Csaba. Partial monitoring – classification, regret bounds, and algorithms. *Mathematics of Operations Research*, 39:967–997, 2014.

Buccapatnam, Swapna, Eryilmaz, Atilla, and Shroff, Ness B. Stochastic bandits with side observations on networks. *SIGMETRICS Perform. Eval. Rev.*, 42(1):289–300, June 2014.

Caron, S., Kveton, B., Lelarge, M., and Bhagat, S. Leveraging side observations in stochastic bandits. In *UAI*, pp. 142–151, 2012.

Graves, Todd L. and Lai, Tze Leung. Asymptotically efficient adaptive choice of control laws in controlled markov chains. *SIAM Journal on Control and Optimization*, 35 (3):715–743, 1997.

Kocák, Tomáš, Neu, Gergely, Valko, Michal, and Munos, Rémi. Efficient learning by implicit exploration in bandit problems with side observations. In *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 613–621, 2014.

Lattimore, T. Optimally confident UCB: Improved regret for finite-armed bandits. Arxiv preprint, 2015.

Magureanu, S., Combes, R., and Proutiere, A. Lipschitz bandits: Regret lower bounds and optimal algorithms. In *COLT*, pp. 975–999, 2014.

Mannor, S. and Shamir, O. From bandits to experts: on the value of side-observations. In *NIPS*, pp. 684–692, 2011.

Robbins, H. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematics Society*, 58:527–535, 1952.