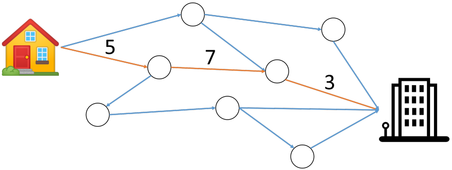
Beating Stochastic and Adversarial Semi-bandits Optimally and Simultaneously

Julian Zimmert (University of Copenhagen)
Haipeng Luo (University of Southern California)
Chen-Yu Wei (University of Southern California)

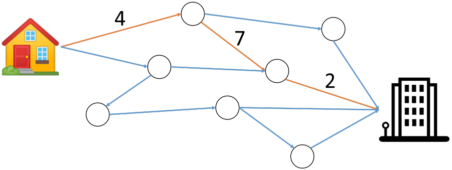
Semi-bandits Example

Day 1



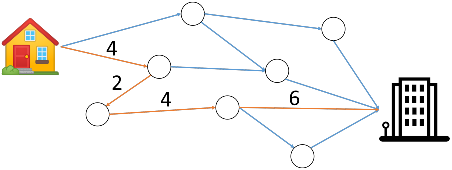
15 mins

Day 2



13 mins

Day 3



16 mins

⋮

Goal: minimize the average commuting time

Types of Environments

i.i.d.
(more benign)

adversarial

Algorithms for i.i.d.: perform bad in the adversarial case.
Algorithms for adversarial: when the environment is i.i.d., they do not take advantage of it.

Types of Environments

i.i.d.
(more benign)

adversarial

Algorithms for i.i.d.: perform bad in the adversarial case.
Algorithms for adversarial: when the environment is i.i.d., they do not take advantage of it.

$\Rightarrow$ To achieve optimal performance, they need to know which environments they are in and pick the corresponding algorithms.

Motivation

i.i.d.
(more benign)

unknown
mixed

adversarial

What if

1. We have no prior knowledge about the environment.
2. The environment is usually i.i.d., but we want to be robust to adversarial attack.
3. The environment is usually arbitrary but we want to exploit the benignness when we got lucky.

Our Results

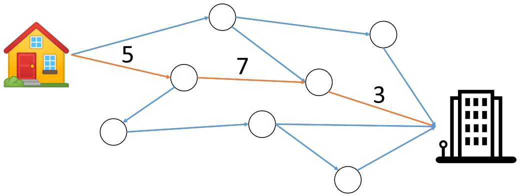
- We propose the first semi-bandit algorithm that has optimal performance guarantees in both i.i.d. and adversarial environments, without knowing which environment it is in.

Formalizing Semi-bandits

Given: action set $\mathcal{X} = \{X^{(1)}, X^{(2)}, \ldots\} \subseteq \{0,1\}^d$.

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

- The learner chooses $X_t \in \mathcal{X}$.
- The environment reveals ℓ_{ti} for which $X_{ti} = 1$.
- The learner suffers loss $\langle X_t, \ell_t \rangle$.



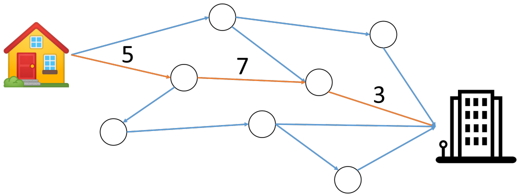
$d = \#\text{edges}$

Formalizing Semi-bandits

Given: action set $\mathcal{X} = \{X^{(1)}, X^{(2)}, \ldots\} \subseteq \{0,1\}^d$. (set of all paths)

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

- The learner chooses $X_t \in \mathcal{X}$.
- The environment reveals ℓ_{ti} for which $X_{ti} = 1$.
- The learner suffers loss $\langle X_t, \ell_t \rangle$.



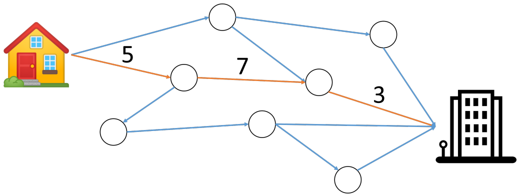
$d = \#\text{edges}$

Formalizing Semi-bandits

Given: action set $\mathcal{X} = \{X^{(1)}, X^{(2)}, \dots\} \subseteq \{0,1\}^d$. (set of all paths)

For $t = 1, \dots, T$,

- The learner chooses $X_t \in \mathcal{X}$ (choose a path).
- The environment reveals ℓ_{ti} for which $X_{ti} = 1$.
- The learner suffers loss $\langle X_t, \ell_t \rangle$.



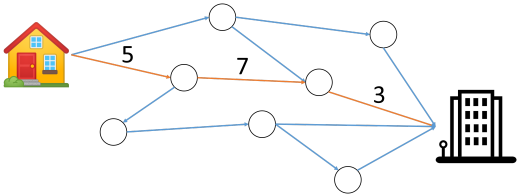
$d = \#\text{edges}$

Formalizing Semi-bandits

Given: action set $\mathcal{X} = \{X^{(1)}, X^{(2)}, \dots\} \subseteq \{0,1\}^d$. (set of all paths)

For $t = 1, \dots, T$,

- The learner chooses $X_t \in \mathcal{X}$ (choose a path).
- The environment reveals ℓ_{ti} for which $X_{ti} = 1$.
  (reveal the cost on each chosen edge)
- The learner suffers loss $\langle X_t, \ell_t \rangle$.



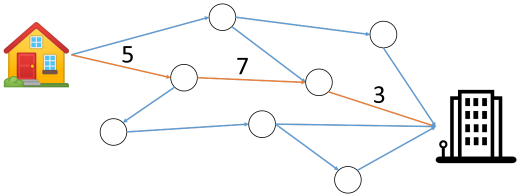
$d = \#\text{edges}$

Formalizing Semi-bandits

Given: action set $\mathcal{X} = \{X^{(1)}, X^{(2)}, \ldots\} \subseteq \{0,1\}^d$. (set of all paths)

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

- The learner chooses $X_t \in \mathcal{X}$ (choose a path).
- The environment reveals ℓ_{ti} for which $X_{ti} = 1$. (reveal the cost on each chosen edge)
- The learner suffers loss $\langle X_t, \ell_t \rangle$. (suffer the path cost)



$d = \#\text{edges}$

Semi-bandits Regret Bounds

Goal: Minimize

$$\text{Regret} = \underbrace{\mathbb{E}\left[\sum_{t=1}^{T} \langle X_t, \ell_t \rangle\right]}_{\text{Learner's total cost}} - \underbrace{\min_{X \in \mathcal{X}} \mathbb{E}\left[\sum_{t=1}^{T} \langle X, \ell_t \rangle\right]}_{\text{Best fixed action's total cost}} .$$

- When ℓ_t are i.i.d.: $\text{Regret} = \Theta(\log T)$
- When ℓ_t are adversarially generated: $\text{Regret} = \Theta\left(\sqrt{T}\right)$

Our algorithm: always has $O(\sqrt{T})$, but gets $O(\log T)$ when the losses happen to be i.i.d.

Related Work in Multi-armed Bandit (MAB)

MAB is special case of SB with $\mathcal{X} = \{\mathbf{e}_1, \dots, \mathbf{e}_d\}$.

Algorithm	Idea

Related Work in Multi-armed Bandit (MAB)

MAB is special case of SB with $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_d\}$.

Algorithm	Idea
SAO [BS12] SAPO [AC16]	i.i.d. algorithm + non-i.i.d. detection

Related Work in Multi-armed Bandit (MAB)

MAB is special case of SB with $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_d\}$.

Algorithm	Idea
SAO [BS12] SAPO [AC16]	i.i.d. algorithm + non-i.i.d. detection
EXP3++ [SS14, SL17]	adversarial algorithm (EXP3) + sophisticated exploration mechanism

Related Work in Multi-armed Bandit (MAB)

MAB is special case of SB with $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_d\}$.

Algorithm	Idea
SAO [BS12] SAPO [AC16]	i.i.d. algorithm + non-i.i.d. detection
EXP3++ [SS14, SL17]	adversarial algorithm (EXP3) + sophisticated exploration mechanism
BROAD [WL18] T-INF [ZS19] (optimal)	adversarial algorithm (FTRL with special regularizer) + improved analysis

Related Work in Multi-armed Bandit (MAB)

MAB is special case of SB with $\mathcal{X} = \{\mathbf{e}_1, \ldots, \mathbf{e}_d\}$.

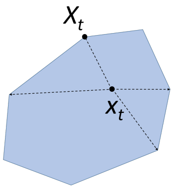
Algorithm	Idea
SAO [BS12] SAPO [AC16]	i.i.d. algorithm + non-i.i.d. detection
EXP3++ [SS14, SL17]	adversarial algorithm (EXP3) + sophisticated exploration mechanism
BROAD [WL18] T-INF [ZS19] (optimal)	adversarial algorithm (FTRL with special regularizer) + improved analysis

Our work is a generalization of [WL18] and [ZS19]'s idea to semi-bandits.

Algorithm

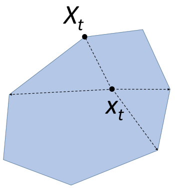
Following the Regularized Leader

Learning rate $\eta_t = 1/\sqrt{t}$, regularizer Ψ



Algorithm

Following the Regularized Leader



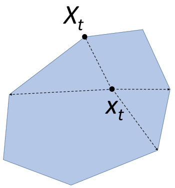
Learning rate $\eta_t = 1/\sqrt{t}$, regularizer Ψ
for $t = 1, 2, 3, \ldots$

- Compute

$$x_t = \operatorname*{argmin}_{x \in \mathrm{Conv}(\mathcal{X})} \left\langle x, \sum_{s=1}^{t-1} \hat{\ell}_s \right\rangle + \eta_t^{-1} \Psi(x).$$

Algorithm

Following the Regularized Leader



Learning rate $\eta_t = 1/\sqrt{t}$, regularizer Ψ
for $t = 1, 2, 3, \ldots$

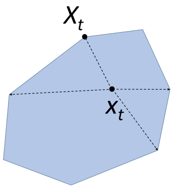
- Compute

$$x_t = \operatorname*{argmin}_{x \in \mathrm{Conv}(\mathcal{X})} \left\langle x, \sum_{s=1}^{t-1} \hat{\ell}_s \right\rangle + \eta_t^{-1} \Psi(x).$$

- Sample X_t such that $\mathbb{E}[X_t] = x_t$, and observe ℓ_{ti} for i with $X_{ti} = 1$.

Algorithm

Following the Regularized Leader



Learning rate $\eta_t = 1/\sqrt{t}$, regularizer Ψ

for $t = 1, 2, 3, \ldots$

- Compute

$$x_t = \operatorname*{argmin}_{x \in \mathrm{Conv}(\mathcal{X})} \left\langle x, \sum_{s=1}^{t-1} \hat{\ell}_s \right\rangle + \eta_t^{-1} \Psi(x).$$

- Sample X_t such that $\mathbb{E}[X_t] = x_t$, and observe ℓ_{ti} for i with $X_{ti} = 1$.
- Construct ℓ_t's unbiased estimator $\hat{\ell}_t$: $\hat{\ell}_{ti} = \frac{\ell_{ti} \mathbf{1}[X_{ti}=1]}{x_{ti}}$.

Regularizer (Key Contribution)

Two-sided hybrid regularizer:

$$\Psi(x) = \underbrace{\sum_{i=1}^{d} -\sqrt{x_i}}_{\text{[AB09]'s Poly-INF}} + \underbrace{\sum_{i=1}^{d} (1 - x_i) \log(1 - x_i)}_{\text{Neg-entropy for complement}} .$$

Regularizer (Key Contribution)

Two-sided hybrid regularizer:

$$\Psi(x) = \underbrace{\sum_{i=1}^{d} -\sqrt{x_i}}_{\text{[AB09]'s Poly-INF}} + \underbrace{\sum_{i=1}^{d} (1 - x_i) \log(1 - x_i)}_{\text{Neg-entropy for complement}} .$$

Intuition:

- when x_i is close to 0, the learner starves for information
  $\Rightarrow$ like a bandit problem
  $\Rightarrow$ using the optimal regularizer for bandit (Poly-INF)
- when x_i is close to 1
  $\Rightarrow$ like a full-info problem
  $\Rightarrow$ using the optimal regularizer for full-info (Neg-entropy)

Results Overview

Env. \ $\mathcal{X}$	General		
i.i.d.	$\frac{md \log T}{\Delta_{\min}}$		
Adversarial	$\sqrt{mdT}$		

$m \triangleq \max_{X \in \mathcal{X}} \|X\|_1$.

$\Delta_{\min} = \mathbb{E}[\text{second-best action's loss}] - \mathbb{E}[\text{best action's loss}]$
(minimal optimality gap)

Results Overview

Env. \ $\mathcal{X}$	General	$\{X \in \{0,1\}^d : \|X\|_1 = m\}$	$\{0,1\}^d$
i.i.d.	$\frac{md \log T}{\Delta_{\min}}$	$\sum_{i>m} \frac{\log T}{\Delta_i}$	$\sum_i \frac{\log T}{\Delta_i}$
Adversarial	$\sqrt{mdT}$	$\begin{cases} \sqrt{mdT}, & m \leq \frac{d}{2} \\ (d-m)\sqrt{T \log d} & m > \frac{d}{2} \end{cases}$	$d\sqrt{T}$

$m \triangleq \max_{X \in \mathcal{X}} \|X\|_1$.

$\Delta_{\min} = \mathbb{E}[\text{second-best action's loss}] - \mathbb{E}[\text{best action's loss}]$
(minimal optimality gap)

Results Overview

Env. \ $\mathcal{X}$	General	$\{X \in \{0,1\}^d : \|X\|_1 = m\}$	$\{0,1\}^d$
i.i.d.	$\frac{md \log T}{\Delta_{\min}}$	$\sum_{i>m} \frac{\log T}{\Delta_i}$	$\sum_i \frac{\log T}{\Delta_i}$
Adversarial	$\sqrt{mdT}$	$\begin{cases} \sqrt{mdT}, & m \le \frac{d}{2} \\ (d-m)\sqrt{T \log d} & m > \frac{d}{2} \end{cases}$	$d\sqrt{T}$

$m \triangleq \max_{X \in \mathcal{X}} \|X\|_1$.

$\Delta_{\min} = \mathbb{E}[\text{second-best action's loss}] - \mathbb{E}[\text{best action's loss}]$ (minimal optimality gap)

Analysis Steps

1. Analyze FTRL for the new regularizer and get $O(\sqrt{T})$ for the adversarial setting.
2. Further use self-bounding technique to get $O(\log T)$ for the i.i.d. setting.

Analyzing FTRL for the New Regularizer

Key lemma.

$$\text{Reg} \le \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \sum_{i} \min\left\{\sqrt{x_{ti}}, \quad (1 - x_{ti})\left(1 + \log \frac{1}{1 - x_{ti}}\right)\right\}.$$

Remarks.

1. The analysis is mostly standard, but needs more care (don't drop some terms as did in usual analysis).
2. The **two-sided**-ness of the regularizer is the key to get "$\min\{\cdot,\cdot\}$".
3. From this bound, we get $O(\sqrt{T})$ bound easily.

Self-bounding to Get $O(\log T)$ Bound

$$\mathrm{Reg} \le \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \underbrace{\sum_i \min\left\{\sqrt{x_{ti}}, \quad (1 - x_{ti})\left(1 + \log \frac{1}{1 - x_{ti}}\right)\right\}}_{\textbf{Goal}\text{: upper bound this by } C\sqrt{\Pr[X_t \neq X^*]}}$$

Intuitively true: $\Pr[X_t \neq X^*] \to 0$

$\Rightarrow x_t \to X^*$

$\Rightarrow$ the above expression $\to 0$.

Self-bounding to Get $O(\log T)$ Bound

$$\text{Reg} \le \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \underbrace{\sum_i \min\left\{\sqrt{x_{ti}}, \quad (1-x_{ti})\left(1+\log\frac{1}{1-x_{ti}}\right)\right\}}_{\textbf{Goal}\text{: upper bound this by } C\sqrt{\Pr[X_t \neq X^*]}}$$

Assume it is proved...

Self-bounding to Get $O(\log T)$ Bound

$$\text{Reg} \le \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \underbrace{\sum_i \min\left\{\sqrt{x_{ti}}, \quad (1-x_{ti})\left(1+\log\frac{1}{1-x_{ti}}\right)\right\}}_{\textbf{Goal}\text{: upper bound this by } C\sqrt{\Pr[X_t \neq X^*]}}$$

Assume it is proved...

$$\sum_t \Delta_{\min} \Pr[X_t \neq X^*] \le \text{Reg}$$

Self-bounding to Get $O(\log T)$ Bound

$$\text{Reg} \le \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \underbrace{\sum_i \min\left\{\sqrt{x_{ti}}, \quad (1-x_{ti})\left(1+\log\frac{1}{1-x_{ti}}\right)\right\}}_{\textbf{Goal}:\ \text{upper bound this by } C\sqrt{\Pr[X_t \neq X^*]}}$$

Assume it is proved...

$$\sum_t \Delta_{\min} \Pr[X_t \neq X^*] \le \text{Reg} \le \sum_t \frac{C\sqrt{\Pr[X_t \neq X^*]}}{\sqrt{t}}$$

$$\le \sum_t \frac{C^2}{2t\Delta_{\min}} + \sum_t \frac{\Delta_{\min}\Pr[X_t \neq X^*]}{2}$$

(AM-GM)

Self-bounding to Get $O(\log T)$ Bound

$$\mathrm{Reg} \le \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \underbrace{\sum_i \min\left\{\sqrt{x_{ti}}, \quad (1-x_{ti})\left(1+\log\frac{1}{1-x_{ti}}\right)\right\}}_{\textbf{Goal}\text{: upper bound this by } C\sqrt{\Pr[X_t \ne X^*]}}$$

Assume it is proved...

$$\sum_t \Delta_{\min} \Pr[X_t \ne X^*] \le \mathrm{Reg} \le \sum_t \frac{C\sqrt{\Pr[X_t \ne X^*]}}{\sqrt{t}}$$

$$\le \sum_t \frac{C^2}{2t\Delta_{\min}} + \sum_t \frac{\Delta_{\min}\Pr[X_t \ne X^*]}{2}$$

(AM-GM)

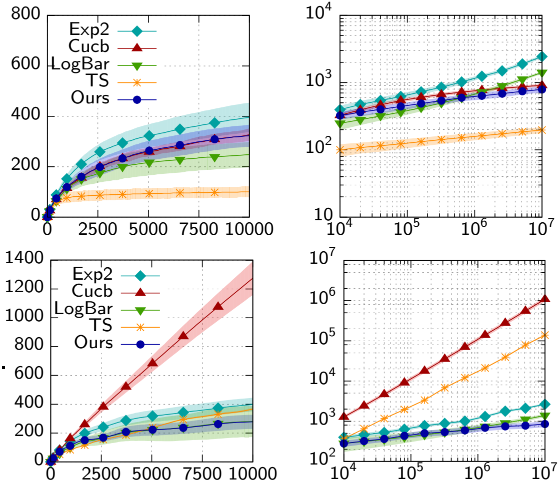
Thus, $\sum_t \Delta_{\min} \Pr[X_t \ne X^*] \le \sum_{t=1}^{T} \frac{C^2}{t\Delta_{\min}} = \frac{C^2 \log T}{\Delta_{\min}}$

$\Longrightarrow \mathrm{Reg} \le \frac{C^2 \log T}{\Delta_{\min}}$.

Experiments (regret vs. time)

i.i.d.

Non-i.i.d.



Summary

- This paper considers semi-bandits, and proposes the first **single** algorithm that has optimal regret guarantees both in **adversarial** and **i.i.d.** environments.
- The algorithm is a simple instantiation of the **Follow the Regularized Leader** framework. The keys to get $O(log T)$ bound in the i.i.d. setting are to
  1. use the two-sided hybrid regularizer
  2. analyze it using the self-bounding technique
- Experiments show our algorithm indeed has best-of-both-world performance, while previous algorithms do not.

Poster #126