

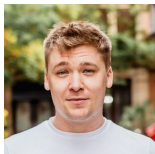
# An Improved Algorithm for Adversarial Linear Contextual Bandits via Reduction

**Tim van Erven**



UNIVERSITY  
OF AMSTERDAM

Joint work with:



Jack Mayo



Julia Olkhovskaya



Chen-Yu Wei

# Outline

## Setting

1. Adversarial Bandits
2. Adversarial Linear Contextual Bandits

## Main Results

3. General Setting
4. First-order Bounds, Initial Setting

## Approach: Reduction to Non-contextual Linear Bandits

5. Reduction: The Basic Idea
6. Approximating  $\Omega$
7. Side-Result: Efficient Robust Linear Bandit Algorithm
8. Controlling the Difference between  $\Psi$  and  $\hat{\Psi}$
9. Restricting  $\pi_t$  to be a Linear Policy

# Adversarial Bandits

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

1. Learner chooses (randomized) arm  $a_t \in \{1, \dots, K\}$
2. Loss value  $\ell_t(a_t)$  is revealed

**Regret w.r.t. arm  $a$ :**

$$R_T(a) = \mathbb{E} \left[ \sum_{t=1}^T \ell_t(a_t) \right] - \sum_{t=1}^T \ell_t(a)$$

# Adversarial Bandits

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

1. Learner chooses (randomized) arm  $a_t \in \{1, \dots, K\}$
2. Loss value  $\ell_t(a_t)$  is revealed

**Regret w.r.t. arm  $a$ :** 
$$R_T(a) = \mathbb{E} \left[ \sum_{t=1}^T \ell_t(a_t) \right] - \sum_{t=1}^T \ell_t(a)$$

- ▶ Oblivious adversary:  $\ell_t(a) \in [0, 1]$  fixed a priori for all  $t, a$
- ▶ Expectation w.r.t. learner's randomness

# Adversarial Contextual Bandits

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

1. **Context**  $X_t \in \mathbb{R}^p$  is revealed
2. Learner chooses (randomized) arm  $a_t \in \{1, \dots, K\}$
3. Loss value  $\ell_t(a_t)$  is revealed

**Regret w.r.t. policy  $\pi$ :**

$$R_T(\pi) = \mathbb{E} \left[ \sum_{t=1}^T \ell_t(a_t) - \sum_{t=1}^T \ell_t(\pi) \right]$$
$$\ell_t(\pi) = \mathbb{E}_{a \sim \pi(X_t)} [\ell_t(a)] \quad \text{for } \pi(X_t) \in \Delta_K$$

# Adversarial Contextual Bandits

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

1. **Context**  $X_t \in \mathbb{R}^p$  is revealed
2. Learner choses (randomized) arm  $a_t \in \{1, \dots, K\}$
3. Loss value  $\ell_t(a_t)$  is revealed

**Regret w.r.t. policy  $\pi$ :**

$$R_T(\pi) = \mathbb{E} \left[ \sum_{t=1}^T \ell_t(a_t) - \sum_{t=1}^T \ell_t(\pi) \right]$$
$$\ell_t(\pi) = \mathbb{E}_{a \sim \pi(X_t)} [\ell_t(a)] \quad \text{for } \pi(X_t) \in \Delta_K$$

**Adversarial losses, stochastic contexts** [Neu and Olkhovskaya, 2020]:

- ▶ **Linear losses:**  $\ell_t(a) = \langle X_t, \theta_{t,a} \rangle \in [-1, +1]$ , where  $\theta_{t,a}$  fixed a priori
- ▶ **I.i.d. contexts:**  $X_t \sim \mathcal{D}$

# Adversarial Contextual Bandits

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

1. **Context**  $X_t \in \mathbb{R}^p$  is revealed
2. Learner choses (randomized) arm  $a_t \in \{1, \dots, K\}$
3. Loss value  $\ell_t(a_t)$  is revealed

**Regret w.r.t. policy  $\pi$ :**

$$R_T(\pi) = \mathbb{E} \left[ \sum_{t=1}^T \ell_t(a_t) - \sum_{t=1}^T \ell_t(\pi) \right]$$
$$\ell_t(\pi) = \mathbb{E}_{a \sim \pi(X_t)} [\ell_t(a)] \quad \text{for } \pi(X_t) \in \Delta_K$$

**Adversarial losses, stochastic contexts** [Neu and Olkhovskaya, 2020]:

- ▶ **Linear losses:**  $\ell_t(a) = \langle X_t, \theta_{t,a} \rangle \in [-1, +1]$ , where  $\theta_{t,a}$  fixed a priori
- ▶ **I.i.d. contexts:**  $X_t \sim \mathcal{D}$
- ▶ (Opposite setting with fixed loss function and adversarial contexts also considered in literature.)

# Adversarial Contextual Bandits More Abstractly I

Incorporate contexts  $X_t \in \mathbb{R}^p$  into actions  $a$  such that

$$\ell_t(a) = \langle X_t, \theta_{t,a} \rangle = \langle a, \theta_t \rangle:$$

$$\theta_t = \begin{pmatrix} \theta_{t,1} \\ \theta_{t,2} \\ \vdots \\ \theta_{t,K} \end{pmatrix} \in \mathbb{R}^{p \times K} \qquad a = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ X_t \\ 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^{p \times K}$$

Then every round we receive a **random action set** (with  $K$  actions):

$$\mathcal{A}_t = \left\{ \begin{pmatrix} X_t \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} X_t \\ 0 \\ \vdots \end{pmatrix}, \dots, \begin{pmatrix} 0 \\ \vdots \\ X_t \end{pmatrix} \right\} \subset \mathbb{R}^{p \times K}$$

# Adversarial Contextual Bandits More Abstractly II

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

1. Draw **action set**  $\mathcal{A}_t \subset \mathbb{R}^d$  i.i.d. from  $\mathcal{D}$
2. Learner chooses (randomized) action  $a_t \in \mathcal{A}_t$
3. Loss value  $\ell_t(a_t) = \langle a_t, \theta_t \rangle \in [-1, +1]$  is revealed

**Regret w.r.t. policy  $\pi$ :**

$$R_T(\pi) = \mathbb{E} \left[ \sum_{t=1}^T \ell_t(a_t) - \sum_{t=1}^T \ell_t(\pi) \right]$$
$$\ell_t(\pi) = \langle \pi(\mathcal{A}_t), \theta_t \rangle \quad \text{for } \pi(\mathcal{A}_t) \in \text{conv}(\mathcal{A}_t)$$

# Adversarial Contextual Bandits More Abstractly II

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

1. Draw **action set**  $\mathcal{A}_t \subset \mathbb{R}^d$  i.i.d. from  $\mathcal{D}$
2. Learner chooses (randomized) action  $a_t \in \mathcal{A}_t$
3. Loss value  $\ell_t(a_t) = \langle a_t, \theta_t \rangle \in [-1, +1]$  is revealed

**Regret w.r.t. policy  $\pi$ :**

$$R_T(\pi) = \mathbb{E} \left[ \sum_{t=1}^T \ell_t(a_t) - \sum_{t=1}^T \ell_t(\pi) \right]$$
$$\ell_t(\pi) = \langle \pi(\mathcal{A}_t), \theta_t \rangle \quad \text{for } \pi(\mathcal{A}_t) \in \text{conv}(\mathcal{A}_t)$$

**Optimal policy is linear:**

$$\min_{\pi} \mathbb{E} \left[ \sum_{t=1}^T \langle \pi(\mathcal{A}_t), \theta_t \rangle \right] = \min_{\pi} \mathbb{E} \left[ \langle \pi(\mathcal{A}), \sum_{t=1}^T \theta_t \rangle \right]$$

$$\pi^*(\mathcal{A}) = \pi_{\phi}(\mathcal{A}) := \arg \min_{a \in \mathcal{A}} \langle a, \phi \rangle \quad \text{for } \phi = \sum_{t=1}^T \theta_t$$

# Outline

## Setting

1. Adversarial Bandits
2. Adversarial Linear Contextual Bandits

## Main Results

3. General Setting
4. First-order Bounds, Initial Setting

## Approach: Reduction to Non-contextual Linear Bandits

5. Reduction: The Basic Idea
6. Approximating  $\Omega$
7. Side-Result: Efficient Robust Linear Bandit Algorithm
8. Controlling the Difference between  $\Psi$  and  $\hat{\Psi}$
9. Restricting  $\pi_t$  to be a Linear Policy

# Main Results I: General Setting

- ▶  $d$ : dimension of the actions,  $\mathcal{A}_t \subset \mathbb{R}^d$
- ▶  $K$ : maximum number of actions,  $|\mathcal{A}_t| \leq K$
- ▶  $C$ : maximum number of linear constraints that describe  $\text{conv}(\mathcal{A}_t)$
- ▶ Simulator: free access to independent samples  $\mathcal{A} \sim \mathcal{D}$

Algorithm	Regret <sup>1</sup>	Runtime	Simulator
Dai et al. [2023]	$\min\{d\sqrt{T}, \sqrt{dT \log K}\}$	$\text{poly}(d, K, T)$	yes
Liu et al. [2023]	$d\sqrt{T}$	$K \cdot T^{\Omega(d)}$	no
Liu et al. [2023]	$d^2\sqrt{T}$	$\text{poly}(d, K, T)$	no
Ours	$d^{1.5}\sqrt{T \log K}$	$\text{poly}(d, C, T)$	no
Ours	$d\sqrt{T}$	$\text{poly}(d, C, T)$	yes

---

<sup>1</sup>Up to poly-logarithmic factors in  $d$  and  $T$

# Main Results I: General Setting

- ▶  $d$ : dimension of the actions,  $\mathcal{A}_t \subset \mathbb{R}^d$
- ▶  $K$ : maximum number of actions,  $|\mathcal{A}_t| \leq K$
- ▶  $C$ : maximum number of linear constraints that describe  $\text{conv}(\mathcal{A}_t)$
- ▶ Simulator: free access to independent samples  $\mathcal{A} \sim \mathcal{D}$

Algorithm	Regret <sup>1</sup>	Runtime	Simulator
Dai et al. [2023]	$\min\{d\sqrt{T}, \sqrt{dT \log K}\}$	$\text{poly}(d, K, T)$	yes
Liu et al. [2023]	$d\sqrt{T}$	$K \cdot T^{\Omega(d)}$	no
Liu et al. [2023]	$d^2\sqrt{T}$	$\text{poly}(d, K, T)$	no
Ours	$d^{1.5}\sqrt{T \log K}$	$\text{poly}(d, C, T)$	no
Ours	$d\sqrt{T}$	$\text{poly}(d, C, T)$	yes

- ▶ Always  $C \leq K + 1$ , but in many combinatorial problems  $C = \text{poly}(d)$  and  $K = 2^{\Omega(d)}$ 
  - ▶ Example: in shortest path with  $d$  edges, set of all paths can be described by a linear program with  $O(d)$  constraints, but number of paths can be of order  $2^{\Omega(d)}$ .

---

<sup>1</sup>Up to poly-logarithmic factors in  $d$  and  $T$

# Main Results I: General Setting

- ▶  $d$ : dimension of the actions,  $\mathcal{A}_t \subset \mathbb{R}^d$
- ▶  $K$ : maximum number of actions,  $|\mathcal{A}_t| \leq K$
- ▶  $C$ : maximum number of linear constraints that describe  $\text{conv}(\mathcal{A}_t)$
- ▶ Simulator: free access to independent samples  $\mathcal{A} \sim \mathcal{D}$

Algorithm	Regret	Runtime	Simulator
Dai et al. [2023]	$\min\{d\sqrt{T}, \sqrt{dT \log K}\}$	$\text{poly}(d, K, T)$	yes
Liu et al. [2023]	$d\sqrt{T}$	$K \cdot T^{\Omega(d)}$	no
Liu et al. [2023]	$d^2\sqrt{T}$	$\text{poly}(d, K, T)$	no
Ours	$d^{1.5}\sqrt{T \log K}$	$\text{poly}(d, C, T)$	no
Ours	$d\sqrt{L^*}$	$\text{poly}(d, C, T)$	yes

$$L^* = \min_{\pi} \mathbb{E} \left[ \sum_{t=1}^T \langle \pi(\mathcal{A}_t), \theta_t \rangle \right] \leq T$$

---

Assuming  $\ell_t(a) \in [0, 1]$

# Main Results II: First-order Bounds, Initial Setting

- ▶  $p$ : dimension of contexts,  $X_t \in \mathbb{R}^p$
- ▶  $K$ : number of actions,  $|\mathcal{A}_t| = K$
- ▶ Simulator: free access to independent samples  $\mathcal{A} \sim \mathcal{D}$

$$L^* = \min_{\pi} \mathbb{E} \left[ \sum_{t=1}^T \langle \pi(\mathcal{A}_t), \theta_t \rangle \right]$$

Algorithm	Regret <sup>1</sup>	Runtime	Simulator	Note
Neu and Olkhovskaya [2020]	$\sqrt{KpT}$	$\text{poly}(p, K, T)$	yes	
Olkhovskaya et al. [2023]	$K\sqrt{pL^*}$	$\Theta\left(T\left(\frac{T}{K^2p}\right)^{Kp}\right)$	no	
Olkhovskaya et al. [2023]	$K\sqrt{pL^*}$	$\text{poly}(p, K, T)$	yes	★
Ours	$Kp\sqrt{L^*}$	$\text{poly}(p, K, T)$	yes	

Strong assumption ★: contexts  $X_t$  have log-concave distribution

---

<sup>1</sup>Up to poly-logarithmic factors

# Outline

## Setting

1. Adversarial Bandits
2. Adversarial Linear Contextual Bandits

## Main Results

3. General Setting
4. First-order Bounds, Initial Setting

## Approach: Reduction to Non-contextual Linear Bandits

5. Reduction: The Basic Idea
6. Approximating  $\Omega$
7. Side-Result: Efficient Robust Linear Bandit Algorithm
8. Controlling the Difference between  $\Psi$  and  $\hat{\Psi}$
9. Restricting  $\pi_t$  to be a Linear Policy

## Reduction: The Basic Idea

Expected loss for policy  $\pi$  in round  $t$  is:

$$\mathbb{E}_{\mathcal{A}_t} [\langle \pi(\mathcal{A}_t), \theta_t \rangle] = \langle \mathbb{E}_{\mathcal{A}_t} [\pi(\mathcal{A}_t)], \theta_t \rangle = \langle \Psi(\pi), \theta_t \rangle,$$

where  $\Psi(\pi)$  is the mean action for  $\pi$ :

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}} [\pi(\mathcal{A})] \in \mathbb{R}^d.$$

## Reduction: The Basic Idea

Expected loss for policy  $\pi$  in round  $t$  is:

$$\mathbb{E}_{\mathcal{A}_t} [\langle \pi(\mathcal{A}_t), \theta_t \rangle] = \langle \mathbb{E}_{\mathcal{A}_t} [\pi(\mathcal{A}_t)], \theta_t \rangle = \langle \Psi(\pi), \theta_t \rangle,$$

where  $\Psi(\pi)$  is the mean action for  $\pi$ :

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}} [\pi(\mathcal{A})] \in \mathbb{R}^d.$$

Possibilities for expected loss:

$$\langle y, \theta_t \rangle \quad \text{for } y \in \Omega = \{\Psi(\pi) \mid \pi \in \Pi\}$$

# Reduction: The Basic Idea

Expected loss for policy  $\pi$  in round  $t$  is:

$$\mathbb{E}_{\mathcal{A}_t} [\langle \pi(\mathcal{A}_t), \theta_t \rangle] = \langle \mathbb{E}_{\mathcal{A}_t} [\pi(\mathcal{A}_t)], \theta_t \rangle = \langle \Psi(\pi), \theta_t \rangle,$$

where  $\Psi(\pi)$  is the mean action for  $\pi$ :

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}} [\pi(\mathcal{A})] \in \mathbb{R}^d.$$

Possibilities for expected loss:

$$\langle y, \theta_t \rangle \quad \text{for } y \in \Omega = \{\Psi(\pi) \mid \pi \in \Pi\}$$

## Reduction:

1. Let linear bandit algorithm choose  $y_t \in \Omega$  in round  $t$
2. Play  $\pi_t$  such that  $\Psi(\pi_t) = y_t$
3. Provide unbiased loss estimate  $\langle \pi_t(\mathcal{A}_t), \theta_t \rangle$  as feedback to linear bandit algorithm

# Reduction: The Basic Idea

Expected loss for policy  $\pi$  in round  $t$  is:

$$\mathbb{E}_{\mathcal{A}_t} [\langle \pi(\mathcal{A}_t), \theta_t \rangle] = \langle \mathbb{E}_{\mathcal{A}_t} [\pi(\mathcal{A}_t)], \theta_t \rangle = \langle \Psi(\pi), \theta_t \rangle,$$

where  $\Psi(\pi)$  is the mean action for  $\pi$ :

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}} [\pi(\mathcal{A})] \in \mathbb{R}^d.$$

Possibilities for expected loss:

$$\langle y, \theta_t \rangle \quad \text{for } y \in \Omega = \{\Psi(\pi) \mid \pi \in \Pi\}$$

## Reduction:

1. Let linear bandit algorithm choose  $y_t \in \Omega$  in round  $t$
2. Play  $\pi_t$  such that  $\Psi(\pi_t) = y_t$
3. Provide unbiased loss estimate  $\langle \pi_t(\mathcal{A}_t), \theta_t \rangle$  as feedback to linear bandit algorithm

NB Hanna et al. [2023] introduced this reduction for different setting of **stochastic** linear contextual bandits, but their techniques do not carry over.

# Approximating $\Omega$

## Reduction:

1. Let linear bandit algorithm choose  $y_t \in \Omega$  in round  $t$
2. Play  $\pi_t$  such that  $\Psi(\pi_t) = y_t$
3. Provide unbiased loss estimate  $\langle \pi_t(\mathcal{A}_t), \theta_t \rangle$  as feedback to linear bandit algorithm

$$\Omega = \{\Psi(\pi) \mid \pi \in \Pi\},$$

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}}[\pi(\mathcal{A})]$$

Issue:  $\Psi$  and  $\Omega$  depend on **unknown distribution  $\mathcal{D}$  of  $\mathcal{A}_t$**

# Approximating $\Omega$

## Reduction:

1. Let linear bandit algorithm choose  $y_t \in \Omega$  in round  $t$
2. Play  $\pi_t$  such that  $\Psi(\pi_t) = y_t$
3. Provide unbiased loss estimate  $\langle \pi_t(\mathcal{A}_t), \theta_t \rangle$  as feedback to linear bandit algorithm

$$\Omega = \{\Psi(\pi) \mid \pi \in \Pi\}, \quad \Psi(\pi) = \mathbb{E}_{\mathcal{A}}[\pi(\mathcal{A})]$$

Issue:  $\Psi$  and  $\Omega$  depend on **unknown distribution  $\mathcal{D}$  of  $\mathcal{A}_t$**

**Using simulator:** Given separate sample  $\tilde{\mathcal{A}}_1, \dots, \tilde{\mathcal{A}}_N$  from  $\mathcal{D}$ :

$$\hat{\Omega} = \{\hat{\Psi}(\pi) \mid \pi \in \Pi\}, \quad \hat{\Psi}(\pi) = \frac{1}{N} \sum_{i=1}^N \pi(\tilde{\mathcal{A}}_i)$$

# Approximating $\Omega$

## Reduction:

1. Let linear bandit algorithm choose  $y_t \in \hat{\Omega}$  in round  $t$
2. Play  $\pi_t$  such that  $\hat{\Psi}(\pi_t) = y_t$
3. Provide **biased** loss estimate  $\langle \pi_t(\mathcal{A}_t), \theta_t \rangle$  as feedback to linear bandit algorithm

$$\Omega = \{\Psi(\pi) \mid \pi \in \Pi\}, \quad \Psi(\pi) = \mathbb{E}_{\mathcal{A}}[\pi(\mathcal{A})]$$

**Using simulator:** Given separate sample  $\tilde{\mathcal{A}}_1, \dots, \tilde{\mathcal{A}}_N$  from  $\mathcal{D}$ :

$$\hat{\Omega} = \{\hat{\Psi}(\pi) \mid \pi \in \Pi\}, \quad \hat{\Psi}(\pi) = \frac{1}{N} \sum_{i=1}^N \pi(\tilde{\mathcal{A}}_i)$$

- Need computationally efficient adversarial linear bandit algorithm that is robust to biased stochastic feedback

# Side-Result: Efficient Robust Linear Bandit Algorithm

## Definition ( $\alpha$ -misspecification-robust linear bandit algorithm)

Given random feedback  $f_t(y_t) \in [-1, +1]$  with bias at most some **known**  $\epsilon \geq 0$ :

$$|\mathbb{E}_t[f_t(y_t)] - \langle y_t, \theta_t \rangle| \leq \epsilon,$$

the algorithm achieves regret at most

$$\mathbb{E} \left[ \sum_{t=1}^T \langle y_t, \theta_t \rangle \right] \leq \min_{y \in \hat{\Omega}} \sum_{t=1}^T \langle y, \theta_t \rangle + \tilde{O}(d\sqrt{T} + \alpha\sqrt{d}\epsilon T).$$

- ▶ [Liu et al., 2024]: optimal  $\alpha = 1$ , but runtime scales with number of actions  $K$
- ▶ New alg:  $\alpha = \sqrt{d}$ , and  $\text{poly}(d, C, T)$  runtime
  - ▶ Version of continuous exponential weights similar to Ito et al. [2020]
  - ▶ Also achieves first-order bound

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Are we there yet?

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Suppose, with high probability,

$$|\langle \Psi(\pi), \theta_t \rangle - \langle \hat{\Psi}(\pi), \theta_t \rangle| \leq \epsilon \quad \text{for } \pi \in \{\pi^*, \pi_t\}, t = 1, \dots, T$$

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Suppose, with high probability,

$$|\langle \Psi(\pi), \theta_t \rangle - \langle \hat{\Psi}(\pi), \theta_t \rangle| \leq \epsilon \quad \text{for } \pi \in \{\pi^*, \pi_t\}, t = 1, \dots, T$$

This would resolve the following **remaining issues**:

1. Controlling bias:

$$|\mathbb{E}_{\mathcal{A}_t}[\langle \pi_t(\mathcal{A}_t), \theta_t \rangle] - \langle y_t, \theta_t \rangle| = |\langle \Psi(\pi_t), \theta_t \rangle - \langle \hat{\Psi}(\pi_t), \theta_t \rangle| \leq \epsilon$$

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Suppose, with high probability,

$$|\langle \Psi(\pi), \theta_t \rangle - \langle \hat{\Psi}(\pi), \theta_t \rangle| \leq \epsilon \quad \text{for } \pi \in \{\pi^*, \pi_t\}, t = 1, \dots, T$$

This would resolve the following **remaining issues**:

1. Controlling bias:

$$|\mathbb{E}_{\mathcal{A}_t}[\langle \pi_t(\mathcal{A}_t), \theta_t \rangle] - \langle y_t, \theta_t \rangle| = |\langle \Psi(\pi_t), \theta_t \rangle - \langle \hat{\Psi}(\pi_t), \theta_t \rangle| \leq \epsilon$$

2. Linear bandit gives regret bound w.r.t.  $y^* \in \hat{\Omega}$  instead of  $\Omega$ :

$$\begin{aligned} \mathbb{E} \left[ \sum_{t=1}^T (\langle y_t, \theta_t \rangle - \langle y^*, \theta_t \rangle) \right] &\geq \mathbb{E} \left[ \sum_{t=1}^T (\langle \hat{\Psi}(\pi_t), \theta_t \rangle - \langle \hat{\Psi}(\pi^*), \theta_t \rangle) \right] \\ &\geq \mathbb{E} \left[ \sum_{t=1}^T (\langle \Psi(\pi_t), \theta_t \rangle - \langle \Psi(\pi^*), \theta_t \rangle) \right] - 2T\epsilon \end{aligned}$$

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Suppose, with high probability,

$$|\langle \Psi(\pi), \theta_t \rangle - \langle \hat{\Psi}(\pi), \theta_t \rangle| \leq \epsilon \quad \text{for } \pi \in \{\pi^*, \pi_t\}, t = 1, \dots, T$$

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}}[\pi(\mathcal{A})] \qquad \hat{\Psi}(\pi) = \frac{1}{N} \sum_{i=1}^N \pi(\tilde{\mathcal{A}}_i)$$

## Lemma (Uniform Convergence over Linear Policies)

Let  $\pi_\phi(\mathcal{A}) := \arg \min_{a \in \mathcal{A}} \langle a, \phi \rangle$  be a linear policy. Then, w.p.  $\geq 1 - \delta$ ,

$$\sup_{\phi} |\langle \Psi(\pi_\phi), \theta_t \rangle - \langle \hat{\Psi}(\pi_\phi), \theta_t \rangle| \leq 2\sqrt{\frac{2d \ln(NK^2)}{N}} + \sqrt{\frac{2 \ln(4/\delta)}{N}}.$$

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Suppose, with high probability,

$$|\langle \Psi(\pi), \theta_t \rangle - \langle \hat{\Psi}(\pi), \theta_t \rangle| \leq \epsilon \quad \text{for } \pi \in \{\pi^*, \pi_t\}, t = 1, \dots, T$$

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}}[\pi(\mathcal{A})] \qquad \hat{\Psi}(\pi) = \frac{1}{N} \sum_{i=1}^N \pi(\tilde{\mathcal{A}}_i)$$

## Lemma (Uniform Convergence over Linear Policies)

Let  $\pi_\phi(\mathcal{A}) := \arg \min_{a \in \mathcal{A}} \langle a, \phi \rangle$  be a linear policy. Then, w.p.  $\geq 1 - \delta$ ,

$$\sup_{\phi} |\langle \Psi(\pi_\phi), \theta_t \rangle - \langle \hat{\Psi}(\pi_\phi), \theta_t \rangle| \leq 2\sqrt{\frac{2d \ln(NK^2)}{N}} + \sqrt{\frac{2 \ln(4/\delta)}{N}}.$$

- Union bound over  $t = 1, \dots, T$  is cheap:  $\delta/T$  instead of  $\delta$

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Suppose, with high probability,

$$|\langle \Psi(\pi), \theta_t \rangle - \langle \hat{\Psi}(\pi), \theta_t \rangle| \leq \epsilon \quad \text{for } \pi \in \{\pi^*, \pi_t\}, t = 1, \dots, T$$

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}}[\pi(\mathcal{A})] \qquad \hat{\Psi}(\pi) = \frac{1}{N} \sum_{i=1}^N \pi(\tilde{\mathcal{A}}_i)$$

## Lemma (Uniform Convergence over Linear Policies)

Let  $\pi_\phi(\mathcal{A}) := \arg \min_{a \in \mathcal{A}} \langle a, \phi \rangle$  be a linear policy. Then, w.p.  $\geq 1 - \delta$ ,

$$\sup_{\phi} |\langle \Psi(\pi_\phi), \theta_t \rangle - \langle \hat{\Psi}(\pi_\phi), \theta_t \rangle| \leq 2\sqrt{\frac{2d \ln(NK^2)}{N}} + \sqrt{\frac{2 \ln(4/\delta)}{N}}.$$

- ▶ Union bound over  $t = 1, \dots, T$  is cheap:  $\delta/T$  instead of  $\delta$
- ▶ We know  $\pi^*$  is always a linear policy, but algorithm's choices  $\pi_t$  may not be! **Problem!**

# Controlling the Difference between $\Psi$ and $\hat{\Psi}$

Suppose, with high probability,

$$|\langle \Psi(\pi), \theta_t \rangle - \langle \hat{\Psi}(\pi), \theta_t \rangle| \leq \epsilon \quad \text{for } \pi \in \{\pi^*, \pi_t\}, t = 1, \dots, T$$

$$\Psi(\pi) = \mathbb{E}_{\mathcal{A}}[\pi(\mathcal{A})] \qquad \hat{\Psi}(\pi) = \frac{1}{N} \sum_{i=1}^N \pi(\tilde{\mathcal{A}}_i)$$

## Lemma (Uniform Convergence over Linear Policies)

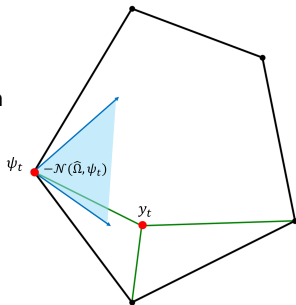
Let  $\pi_\phi(\mathcal{A}) := \arg \min_{a \in \mathcal{A}} \langle a, \phi \rangle$  be a linear policy. Then, w.p.  $\geq 1 - \delta$ ,

$$\sup_{\phi} |\langle \Psi(\pi_\phi), \theta_t \rangle - \langle \hat{\Psi}(\pi_\phi), \theta_t \rangle| \leq 2\sqrt{\frac{2d \ln(NK^2)}{N}} + \sqrt{\frac{2 \ln(4/\delta)}{N}}.$$

- ▶ Union bound over  $t = 1, \dots, T$  is cheap:  $\delta/T$  instead of  $\delta$
- ▶ We know  $\pi^*$  is always a linear policy, but algorithm's choices  $\pi_t$  may not be! **Problem!**
- ▶ Can we solve this by extending to uniform convergence over all policies  $\pi$ ? No, does not hold! So need to ensure  $\pi_t$  is linear policy.

# Restricting $\pi_t$ to be a Linear Policy

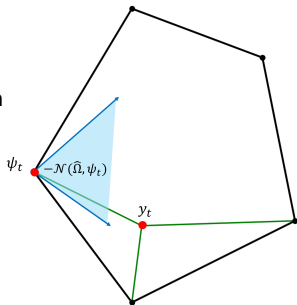
- ▶  $\hat{\Omega} \subset \mathbb{R}^d$  is a polytope
- ▶ Lemma: for every vertex  $v$  of  $\hat{\Omega}$ , there exists a linear policy  $\pi_\phi$  that maps to it:  $\hat{\Psi}(\pi_\phi) = v$ .
  - ▶ In fact, this holds for any interior point  $\phi$  of the negative normal cone  $-\mathcal{N}(\hat{\Omega}, v)$  at  $v$ .



# Restricting $\pi_t$ to be a Linear Policy

- ▶  $\hat{\Omega} \subset \mathbb{R}^d$  is a polytope
- ▶ Lemma: for every vertex  $v$  of  $\hat{\Omega}$ , there exists a linear policy  $\pi_\phi$  that maps to it:  $\hat{\Psi}(\pi_\phi) = v$ .
  - ▶ In fact, this holds for any interior point  $\phi$  of  $\psi_t$  the negative normal cone  $-\mathcal{N}(\hat{\Omega}, v)$  at  $v$ .
- ▶ By Carathéodory's theorem,  $y_t$  is a convex combination of  $m \leq d + 1$  vertices  $v_1, \dots, v_m$ :

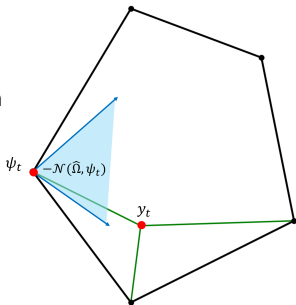
$$y_t = \sum_{j=1}^m \lambda_j v_j \quad \text{for } \lambda = (\lambda_1, \dots, \lambda_m) \in \Delta_m \quad (1)$$



# Restricting $\pi_t$ to be a Linear Policy

- ▶  $\hat{\Omega} \subset \mathbb{R}^d$  is a polytope
- ▶ Lemma: for every vertex  $v$  of  $\hat{\Omega}$ , there exists a linear policy  $\pi_\phi$  that maps to it:  $\hat{\Psi}(\pi_\phi) = v$ .
  - ▶ In fact, this holds for any interior point  $\phi$  of  $\psi_t$  of the negative normal cone  $-\mathcal{N}(\hat{\Omega}, v)$  at  $v$ .
- ▶ By Carathéodory's theorem,  $y_t$  is a convex combination of  $m \leq d + 1$  vertices  $v_1, \dots, v_m$ :

$$y_t = \sum_{j=1}^m \lambda_j v_j \quad \text{for } \lambda = (\lambda_1, \dots, \lambda_m) \in \Delta_m \quad (1)$$



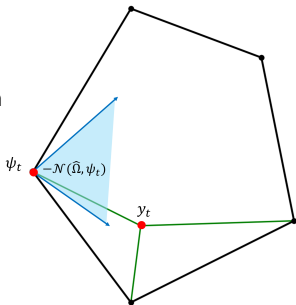
## Solution

Instead of playing  $y_t$ , sample one of the vertices  $v_1, \dots, v_m$  according to  $\lambda$  and play the corresponding linear policy  $\pi_\phi \rightarrow$  same expected loss.

# Restricting $\pi_t$ to be a Linear Policy

- ▶  $\hat{\Omega} \subset \mathbb{R}^d$  is a polytope
- ▶ Lemma: for every vertex  $v$  of  $\hat{\Omega}$ , there exists a linear policy  $\pi_\phi$  that maps to it:  $\hat{\Psi}(\pi_\phi) = v$ .
  - ▶ In fact, this holds for any interior point  $\phi$  of  $\psi_t$  of the negative normal cone  $-\mathcal{N}(\hat{\Omega}, v)$  at  $v$ .
- ▶ By Carathéodory's theorem,  $y_t$  is a convex combination of  $m \leq d + 1$  vertices  $v_1, \dots, v_m$ :

$$y_t = \sum_{j=1}^m \lambda_j v_j \quad \text{for } \lambda = (\lambda_1, \dots, \lambda_m) \in \Delta_m \quad (1)$$



## Solution

Instead of playing  $y_t$ , sample one of the vertices  $v_1, \dots, v_m$  according to  $\lambda$  and play the corresponding linear policy  $\pi_\phi \rightarrow$  same expected loss.

- ▶ Computation: We can both find the decomposition (1) and the interior point  $\phi$  of the normal cone in  $\text{poly}(d, N, C)$  time, because we can construct an efficient separation oracle for  $\hat{\Omega}$ .

# Putting It All Together

## Theorem

*Given access to an  $\alpha$ -misspecification robust linear bandit algorithm, we obtain*

$$R_T(\pi) = \tilde{O}\left(d\sqrt{T} + \alpha Td\sqrt{\frac{\log(NKT)}{N}}\right).$$

- ▶ We had  $\alpha = \sqrt{d}$  for an efficient algorithm

# Putting It All Together

## Theorem

*Given access to an  $\alpha$ -misspecification robust linear bandit algorithm, we obtain*

$$R_T(\pi) = \tilde{O}\left(d\sqrt{T} + \alpha Td\sqrt{\frac{\log(NKT)}{N}}\right).$$

- ▶ We had  $\alpha = \sqrt{d}$  for an efficient algorithm

**With simulator access:** Take  $N$  large enough to make the second term negligible:

$$R_T(\pi) = \tilde{O}(d\sqrt{T})$$

# Putting It All Together

## Theorem

*Given access to an  $\alpha$ -misspecification robust linear bandit algorithm, we obtain*

$$R_T(\pi) = \tilde{O}\left(d\sqrt{T} + \alpha Td\sqrt{\frac{\log(NKT)}{N}}\right).$$

- ▶ We had  $\alpha = \sqrt{d}$  for an efficient algorithm

**With simulator access:** Take  $N$  large enough to make the second term negligible:

$$R_T(\pi) = \tilde{O}(d\sqrt{T})$$

**Without simulator access:**

- ▶ Run in epochs of lengths  $2^i$  for  $i = 0, 1, 2, 3, \dots$
- ▶ In epoch  $i$  use  $N = \Theta(2^i)$  samples from all previous epochs to construct  $\hat{\Omega}$

$$R_T(\pi) = \tilde{O}(d\sqrt{T} + \alpha d\sqrt{T \log(KT)})$$

# Conclusion

## Highlights:

- ▶ First algorithm for this setting that handles combinatorial action sets efficiently
- ▶ Efficient reduction from contextual to (misspecified) non-contextual linear bandits
- ▶ Handle resulting misspecification in linear bandit algorithm

## Open Questions:

- ▶ Improve computation to match Neu and Valko [2014]? For semi-bandit feedback, they only require a linear optimization oracle for each action set instead of a polynomial number of constraints.
- ▶ Improve regret to  $\tilde{O}(d\sqrt{T})$  with polynomial-time algorithm without a simulator?
- ▶ Improve first-order bound to  $\tilde{O}(\sqrt{pKL^*})$  in initial setting?

# References I

- Y. Dai, H. Luo, C.-Y. Wei, and J. Zimmert. Refined regret for adversarial MDPs with linear function approximation. In *International Conference on Machine Learning*, pages 6726–6759. PMLR, 2023.
- O. A. Hanna, L. Yang, and C. Fragouli. Contexts can be cheap: Solving stochastic contextual bandits with linear bandit algorithms. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 1791–1821. PMLR, 2023.
- S. Ito, S. Hirahara, T. Soma, and Y. Yoshida. Tight first-and second-order regret bounds for adversarial linear bandits. *Advances in Neural Information Processing Systems*, 33:2028–2038, 2020.
- H. Liu, C.-Y. Wei, and J. Zimmert. Bypassing the simulator: Near-optimal adversarial linear contextual bandits. *Advances in Neural Information Processing Systems*, 36, 2023.
- H. Liu, A. Tajdini, A. Wagenmaker, and C.-Y. Wei. Corruption-robust linear bandits: Minimax optimality and gap-dependent misspecification. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.
- G. Neu and J. Olkhovskaya. Efficient and robust algorithms for adversarial linear contextual bandits. In J. Abernethy and S. Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 3049–3068. PMLR, 09–12 Jul 2020. URL <https://proceedings.mlr.press/v125/neu20b.html>.

# References II

- G. Neu and M. Valko. Online combinatorial optimization with stochastic decision sets and adversarial losses. *Advances in Neural Information Processing Systems*, 27, 2014.
- J. Olkhovskaya, J. Mayo, T. van Erven, G. Neu, and C.-Y. Wei. First-and second-order bounds for adversarial linear contextual bandits. *Advances in Neural Information Processing Systems*, 36, 2023.