Optimal Batched Linear Bandits

Xuanfei Ren[*], **Tianyuan Jin**[†], **Pan Xu**[§]

[*]University of Science and Technology of China
[†]National University of Singapore
[§]Duke University

The Forty-first International Conference on Machine Learning

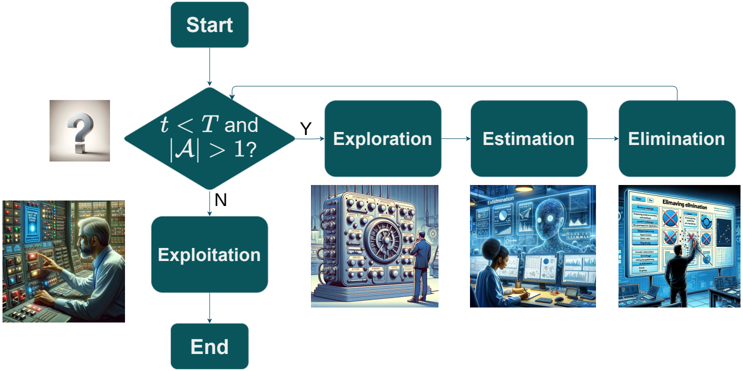
Outline

1. Problem Setup
2. Research Goal
3. Algorithm and Analysis

Outline

Overview

Optimal Batched Linear Bandits



Start

$t < T$ and $|\mathcal{A}| > 1$?

Y

Exploration

Estimation

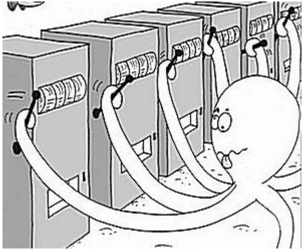
Elimination

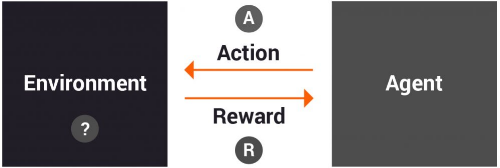
N

Exploitation

End

Bandit Problem

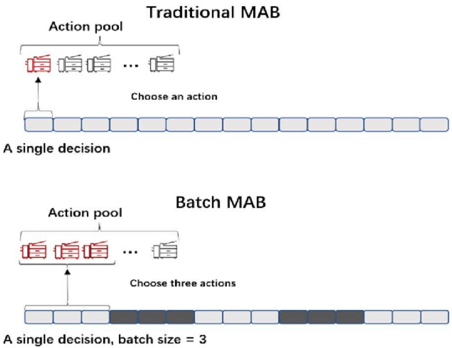




This is a fully sequential decision problem!

Batch Mode

Batch mode The player commits to a sequence of actions (*a batch of actions*) and observes the rewards *after all actions in that sequence are played*.



Problem Setting

Batched linear bandits

Notations:

- T time horizon
- $\mathcal{X}$ a fixed set of K actions
- θ^* an unknown parameter
- Rewards: $y_t = \langle x_t, \theta^* \rangle + \varepsilon_t$
- Regret: $R_T = E[\max_{x \in X} \Sigma_{t=1}^{T} \langle x - x_t, \theta^* \rangle]$
- Batch complexity: number of batches

Our goal is to design batched algorithms that achieve optimal regret and batch complexity in different senses.

Existing Lower Bound

Asymptotic lower bound For an allocation $\alpha \in \mathbb{R}^k_{\geq 0}$ over actions we define the associated covariance matrix $H(\alpha) = \sum_{x \in X} \alpha(x) x x^T$. Let c^* be the solution to the following convex program,

$$c^*(\theta^*) \triangleq \inf_{\alpha \in \mathbb{R}^k_{\geq 0}} \sum_{x \in X} \alpha(x) \Delta(x) \quad \text{s.t.} \quad \|x\|^2_{H^{-1}(\alpha)} \leq \frac{\Delta_x^2}{2}, \forall x \in \mathcal{X}^- := \mathcal{X} - \{x^*\}, \qquad (1)$$

In paper *The End of Optimism* [LS17], it is stated that any consistent algorithm π for the linear bandit setting with has regret $R_T(\theta^*, \pi)$ at least

$$\liminf_{T \to \infty} \frac{R_T(\theta^*, \pi)}{\log(T)} \geq c^*(\theta^*).$$

Outline

Our Goal

Non-asymptotic

- minimax optimal regret, best instance-dependent regret bound
- batch complexity that matches existing lower bound, instance-dependent batch complexity

Our Goal

Non-asymptotic

- minimax optimal regret, best instance-dependent regret bound
- batch complexity that matches existing lower bound, instance-dependent batch complexity

Asymptotic $T \to \infty$

- asymptotically optimal regret
- asymptotically optimal batch complexity

Our Goal

Non-asymptotic

- minimax optimal regret, best instance-dependent regret bound
- batch complexity that matches existing lower bound, instance-dependent batch complexity

Asymptotic $T \to \infty$

- asymptotically optimal regret
- asymptotically optimal batch complexity

The first algorithm for linear bandits that simultaneously achieves the minimax and asymptotic optimality in regret with the corresponding optimal batch complexities!

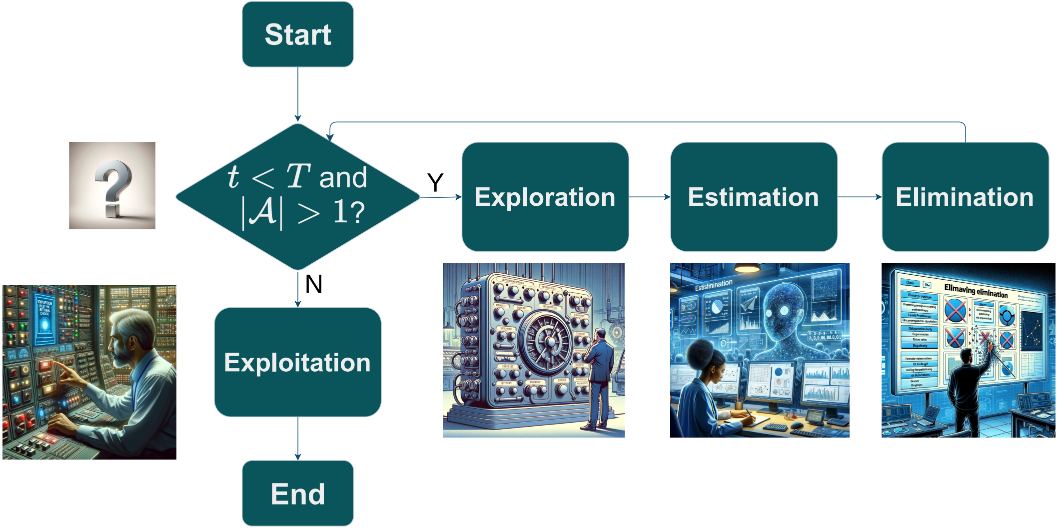
Related Work Comparison

Algorithm	Non-asymptotic setting		Asymptotic setting	
	Worst-case regret	Batch complexity	Asymptotic regret	Batch complexity
[AYPS11]	$\tilde{O}(d\sqrt{T})$	$O(\log T)$	-	-
[EKMM21]	$\tilde{O}(\sqrt{dT})$	$O(\log T)$	-	-
[RYZ21]	$\tilde{O}(\sqrt{dT})$	$O(\log\log T)$	-	-
[HYF23]	$\tilde{O}(\sqrt{dT})$	$O(\log\log T)$	-	-
Lower bound [GHRZ19]	$\Omega(\sqrt{dT})$	$\Omega(\log\log T)$	-	-
[LS17]	-	-	Optimal	Sequential
OSSB [CMP17]	-	-	Optimal	Sequential
OAM [HLS20]	-	-	Optimal	Sequential
SOLID [TPRL20]	$\tilde{O}((d+\log K)\sqrt{T})$	Sequential	Optimal	Sequential
IDS [KLVS21]	$\tilde{O}(d\sqrt{T})$	$\geq O(d^4 \log^4 T/\Delta_{\min}^2)$	Optimal	$\geq O(\log^4 T)$
Batch lower bound	-	-	Optimal	3
E^4(Our Algorithm)	$\tilde{O}(\sqrt{dT})$	$O(\log\log T)$	Optimal	3

Outline

Outline

Algorithm Framework: E^4



Start

$t < T$ and $|\mathcal{A}| > 1$?

Y

Exploration

Estimation

Elimination

N

Exploitation

End

Sampling Rule

For any $\Delta \in [0, \infty)^k$ define $w(\Delta) \in [0, \infty]^k$ to be a solution to the optimisation problem

$$\min_{w \in [0,\infty]^k} \sum_{\boldsymbol{x} \in \mathcal{X}} w_x \Delta(\boldsymbol{x})$$

$$\text{s.t.} \quad \|\boldsymbol{x}\|^2_{H_w^{-1}} \leq \frac{\Delta_{\boldsymbol{x}}^2}{2}, \forall \boldsymbol{x} \in \mathcal{X},$$

where $H_w = \sum_{x \in \mathcal{X}} w_x x x^T$.

Sampling rule: use estimators to calculate $w(\hat{\Delta})$, then sample according to this proportion.

Stopping Rule

Chernoff's Stopping Rule (Generalized likelihood ratio test):

If we find the best arm with probability at least $1 - 1/T$, then stop to commit.

Define

$$Z(t) = \min_{x \neq \hat{x}^*} \frac{\hat{\Delta}_x^2}{2\|\hat{x}^* - x\|_{H_t^{-1}}^2}$$

$$\tau = \inf\left\{t \in \mathbb{N}^* : Z(t) \geq \beta(\delta, t) \text{ and } \sum_{s=1}^{t} x_t x_t^T \geq cI_d\right\},$$

where τ is a stopping time.

Choosing proper threshold β to make $\mathbb{P}\left(\tau < \infty, {\theta^*}^T(x^* - \hat{x}_\tau^*) > 0\right) \leq \delta$.

Uniform Exploration for Linear Bandits: D-Optimal Design

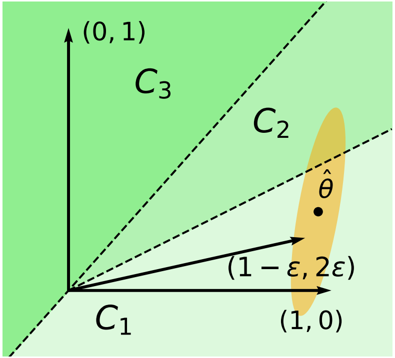
Definition D-optimal design sampling allocation is given by:

$$\min_{\pi} \max_{x \in \mathcal{X}} \|x\|^2_{H_\pi^{-1}}, \quad H_\pi = \sum_{x \in \mathcal{X}} \pi_x \cdot x x^\top.$$

Pulling arms according to this special design leads to good concentration results like:

$$|\langle \hat{\theta} - \theta^*, x \rangle| \leq \sqrt{d \log(1/\delta)/T_\varepsilon}, \quad \forall x \in \mathcal{X}$$

where the total pulling number is $\Theta(T_\varepsilon)$.



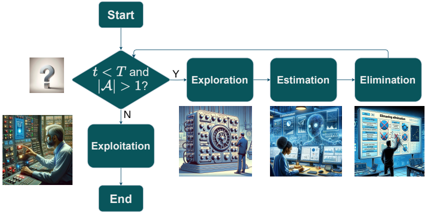
Algorithm Design

The performance of our algorithm in each batch:

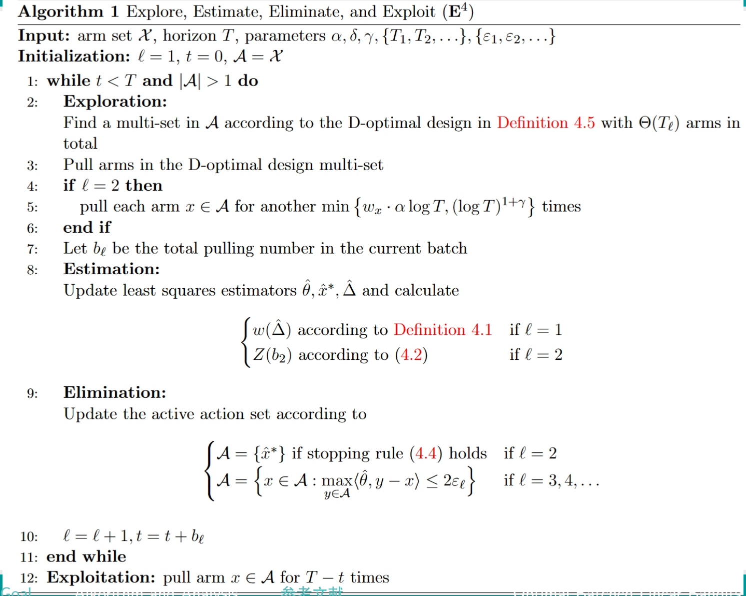
1. Exploration: D-optimal design;
   Estimation: calculate sampling proportion w
2. Exploration: D-optimal design and according to the proportion;
   Estimation: calculate stopping statistics Z;
   Elimination: stopping rule
3. Exploration: D-optimal design; Elimination:

$$\mathcal{A} = \left\{ x \in \mathcal{A} : \max_{y \in \mathcal{A}} \langle \hat{\theta}, y - x \rangle \leq 2\varepsilon_\ell \right\}$$

4. Repeat step 3 until $|\mathcal{A}| = 1$ or $t = T$
5. Exploitation: Commit to the estimated best arm



Algorithm design



Algorithm 1 Explore, Estimate, Eliminate, and Exploit ($\mathbf{E}^4$)

```
Input: arm set $\mathcal{X}$, horizon $T$, parameters $\alpha, \delta, \gamma, \{T_1, T_2, \ldots\}, \{\varepsilon_1, \varepsilon_2, \ldots\}$
Initialization: $\ell = 1$, $t = 0$, $\mathcal{A} = \mathcal{X}$
while $t < T$ and $|\mathcal{A}| > 1$ do
    Exploration:
    Find a multi-set in $\mathcal{A}$ according to the D-optimal design in Definition 4.5 with $\Theta(T_\ell)$ arms in total
    Pull arms in the D-optimal design multi-set
    if $\ell = 2$ then
        pull each arm $x \in \mathcal{A}$ for another $\min\left\{w_x \cdot \alpha \log T, (\log T)^{1+\gamma}\right\}$ times
    end if
    Let $b_\ell$ be the total pulling number in the current batch
    Estimation:
    Update least squares estimators $\hat{\theta}, \hat{x}^*, \hat{\Delta}$ and calculate
        $\begin{cases} w(\hat{\Delta}) \text{ according to Definition 4.1} & \text{if } \ell = 1 \\ Z(b_2) \text{ according to (4.2)} & \text{if } \ell = 2 \end{cases}$
    Elimination:
    Update the active action set according to
        $\begin{cases} \mathcal{A} = \{\hat{x}^*\} \text{ if stopping rule (4.4) holds} & \text{if } \ell = 2 \\ \mathcal{A} = \left\{x \in \mathcal{A} : \max_{y \in \mathcal{A}} \langle \hat{\theta}, y - x \rangle \le 2\varepsilon_\ell \right\} & \text{if } \ell = 3, 4, \ldots \end{cases}$
    $\ell = \ell + 1, t = t + b_\ell$
end while
Exploitation: pull arm $x \in \mathcal{A}$ for $T - t$ times
```

Outline

1. Problem Setup
2. Research Goal
3. Algorithm and Analysis
   - Design
   - Theoretical Analysis
   - Empirical Results

Main Results: Regret and Batch Complexity

Define

$$\mathcal{T}_1 = \{T_1 = (\log T)^{1/2}, T_2 = (\log T)^{1/2}, T_3 = (\log T)^{1+\gamma}, T_\ell = T^{1-\frac{1}{2^{\ell-3}}}, \ell \geq 4\}.$$

- When $\{T_\ell\}_{\ell=1}^{\infty} = \mathcal{T}_1$, our algorithm achieves

$$Regret(T) \leq \tilde{O}(\sqrt{dT}),$$

  with at most $O(\log \log T)$ batches.

Main Results: Regret and Batch Complexity

Define

$$\mathcal{T}_2 = \{T_1 = (\log T)^{1/2}, T_2 = (\log T)^{1/2}, T_3 = (\log T)^{1+\gamma}, T_\ell = d\log(kT^2)\cdot 2^{\ell-3}, \ell \geq 4\}.$$

- When $\{T_\ell\}_{\ell=1}^{\infty} = \mathcal{T}_2$, our algorithm achieves $\tilde{O}(\sqrt{dT})$ regret and

$$Regret(T) \leq O\left((\log T)^{1+\gamma} + \frac{d\log(KT)}{\Delta_{\min}}\right) = \tilde{O}\left(\frac{d}{\Delta_{\min}}\right),$$

  with at most $O(\log T)$ batches and in expectation $O(\log(1/\Delta_{\min}))$ batches.

Asymptotic Results

- In the asymptotic setting, when $T \to \infty$, our algorithm with $\{T_\ell\}_{\ell=1}^{\infty}$ equaling $\mathcal{T}_1$ or $\mathcal{T}_2$ achieves asymptotic optimality defined above, i.e.,

$$\limsup_{T\to\infty} \frac{Regret(T)}{\log(T)} \leq c^*,$$

with 3 batches in expectation.

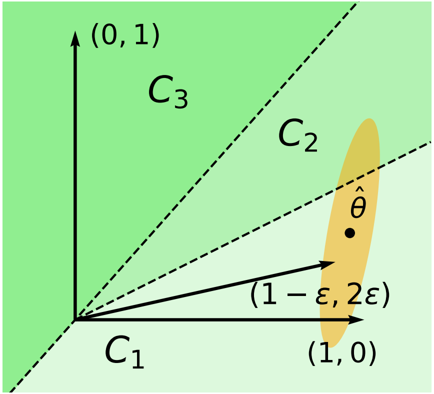
Main Results: Batch Complexity Lower Bound

We prove:
Theorem (Batch complexity lower bound) If an algorithm achieves asymptotic optimality, then on all bandit instances, it must have at least 3 batches in expectation as $T \to \infty$.

The batch complexity of our algorithm matches this lower bound!

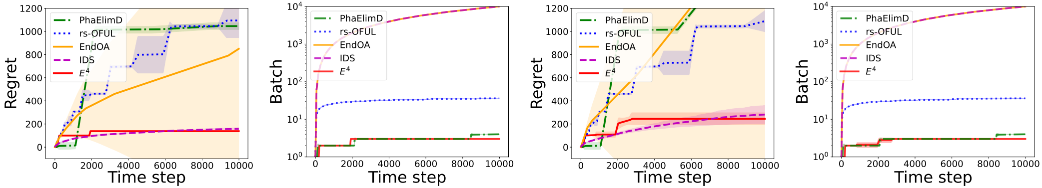
Outline

Experiments



"End of Optimism" instance.

Experiments



(a) $d = 2, K = 3, \epsilon = 0.01$ (b) $d = 2, K = 3, \epsilon = 0.01$ (c) $d = 2, K = 3, \epsilon = 0.2$ (d) $d = 2, K = 3, \epsilon = 0.2$

Regret and Batch Analysis: "End of Optimism" instances ($d = 2, K = 3$).

Instance		**E**^{4}	**PhaElimD**	**rs-OFUL**	**EndOA**	**IDS**
$d = 2, K = 3, T = 10000$	$\epsilon = 0.01$	3.0 ± 0.0	4.0 ± 0.0	36.1 ± 0.3	-	-
	$\epsilon = 0.2$	3.0 ± 0.0	4.0 ± 0.0	37.0 ± 0.0	-	-
$d = 3, K = 5, T = 50000$	$\epsilon = 0.01$	3.0 ± 0.0	4.0 ± 0.0	61.0 ± 0.5	-	-
	$\epsilon = 0.2$	3.0 ± 0.0	4.0 ± 0.0	60.5 ± 0.8	-	-
$d = 5, K = 9, T = 100000$	$\epsilon = 0.01$	3.0 ± 0.0	4.0 ± 0.0	102.3 ± 0.9	-	-
	$\epsilon = 0.2$	3.0 ± 0.0	4.0 ± 0.0	101.8 ± 0.6	-	-

Batch Complexity Analysis: "End of Optimism" instances. Note that batch complexity of sequential algorithms like **EndOA** and **IDS** equals time horizon.

Experiments

Instance		**E^4**	**PhaElimD**	**rs-OFUL**	**EndOA**	**IDS**
$d = 2, K = 3, T = 10000$	$\epsilon = 0.01$	0.04	0.18	0.45	3.15	9.48
	$\epsilon = 0.2$	0.06	0.15	0.28	2.23	6.42
$d = 3, K = 5, T = 50000$	$\epsilon = 0.01$	0.12	0.71	1.47	3.17	30.22
	$\epsilon = 0.2$	0.15	0.76	1.60	3.87	13.86
$d = 5, K = 9, T = 100000$	$\epsilon = 0.01$	0.25	1.46	3.72	8.94	178.31
	$\epsilon = 0.2$	0.33	1.40	2.90	10.19	246.53

Runtime comparison (Unit: second per experiment).

Thank you!

Reference I

[AYPS11] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

[CMP17] Richard Combes, Stefan Magureanu, and Alexandre Proutiere. Minimal exploration in structured stochastic bandits. *Advances in Neural Information Processing Systems*, 30, 2017.

[EKMM21] Hossein Esfandiari, Amin Karbasi, Abbas Mehrabian, and Vahab Mirrokni. Regret bounds for batched bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7340–7348, 2021.

[GHRZ19] Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. Batched multi-armed bandits problem. *Advances in Neural Information Processing Systems*, 32, 2019.

Reference II

[HLS20] Botao Hao, Tor Lattimore, and Csaba Szepesvari. Adaptive exploration in linear contextual bandit. In *International Conference on Artificial Intelligence and Statistics*, pages 3536–3545. PMLR, 2020.

[HYF23] Osama A Hanna, Lin Yang, and Christina Fragouli. Contexts can be cheap: Solving stochastic contextual bandits with linear bandit algorithms. In Gergely Neu and Lorenzo Rosasco, editors, *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pages 1791–1821. PMLR, 12–15 Jul 2023.

[KLVS21] Johannes Kirschner, Tor Lattimore, Claire Vernade, and Csaba Szepesvári. Asymptotically optimal information-directed sampling. In *Conference on Learning Theory*, pages 2777–2821. PMLR, 2021.

[LS17] Tor Lattimore and Csaba Szepesvari. The end of optimism? an asymptotic analysis of finite-armed linear bandits. In *Artificial Intelligence and Statistics*, pages 728–737. PMLR, 2017.

Reference III

[RYZ21] Yufei Ruan, Jiaqi Yang, and Yuan Zhou. Linear bandits with limited adaptivity and learning distributional optimal design. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 74–87, 2021.

[TPRL20] Andrea Tirinzoni, Matteo Pirotta, Marcello Restelli, and Alessandro Lazaric. An asymptotically optimal primal-dual incremental algorithm for contextual linear bandits. *Advances in Neural Information Processing Systems*, 33:1417–1427, 2020.