A Direct Approach for Handling Contextual Bandits with Latent State Dynamics

Zhen LI [1] Gilles Stoltz[2,3]

[1] BNP Paribas Corporate and Institutional Banking

[2]Université Paris-Saclay, CNRS, Inria, Laboratoire de mathématiques d'Orsay

[3]HEC Paris

Motivation: Beyond Standard Linear Contextual Bandits

A standard *Linear Contextual Bandits* assumes that the observed context contains all reward-relevant state information:

$$\mathbb{E}\big[r_t(a) \mid \mathbf{x}_t, \text{past}\big] = \varphi(a, \mathbf{x}_t)^\top \theta_\star.$$

This assumption is too optimistic when the environment has latent states h_t.

- Latent states may drive both the context distribution and the reward model;
- The learner observes $\mathbf{x}_t$ and the reward of the chosen action, but never observes h_t.
- The latent state may switch frequently over time.

Research question. Can we design a contextual-bandit algorithm with sublinear regret when latent states evolve according to an HMM, rewards are continuous, and rewards depend jointly on the latent state, the context, and the action?

Learning Protocol: HMM-Generated Contexts

Latent-state dynamics: Hidden Markov Model (HMM).

$$h_1 \sim \boldsymbol{\pi}, \qquad \mathbf{x}_t \mid h_t \sim \nu_{h_t}, \qquad h_{t+1} \mid h_t \sim \mathbf{M}_{h_t,\cdot}$$

For rounds $t = 1, 2, \ldots, T$:

1. Hidden state $h_t \in [H]$ evolves according to the HMM, but is not observed;
2. Context $\mathbf{x}_t \in \mathcal{X}$ is observed, with $\mathbf{x}_t \sim \nu_{h_t}$;
3. The learner chooses an action $a_t \in \mathcal{A}$;
4. The learner observes only the reward $r_t(a_t)$.

Context-only belief.

$$\mathbf{b}_t(h) = \mathbb{P}(h_t = h \mid \mathbf{x}_{1:t}).$$

This posterior uses the context sequence only, not the reward feedback.

Belief-oracle benchmark. An oracle knows the HMM and reward parameters, and plays

$$a_t^\star \in \operatorname*{argmax}_{a \in \mathcal{A}} \sum_{h \in [H]} \mathbf{b}_t(h)\, \boldsymbol{\varphi}(a, \mathbf{x}_t)^\top \boldsymbol{\theta}_h^\star$$

The pseudo-regret compares the learner against this oracle in cumulative belief-expected reward.

Simplified Model: Belief-Dependent Rewards

Nelson et al., 2022 consider a belief-dependent reward model. In their setting, contexts are used to infer the latent state h_t, but do not enter the reward model directly:

$$r'_t(a) = \sum_{h=1}^{H} \mathbf{b}_t(h)\theta^\star_h + \eta'_t(a).$$

We extend this simplified model by allowing rewards to depend on both the belief and the observed context:

$$r'_t(a) = \sum_{h\in[H]} \mathbf{b}_t(h)\, \boldsymbol{\varphi}(a, \mathbf{x}_t)^\top \boldsymbol{\theta}^\star_h + \eta'_t(a).$$

Why this is easier. With known beliefs, this is a standard linear contextual bandit with feature vector $\mathbf{b}_t \otimes \boldsymbol{\varphi}(a, \mathbf{x}_t)$. With estimated beliefs, the standard LinUCB analysis (Abbasi-Yadkori et al., 2011) carries over up to an additional belief-estimation error term.

Contrib. #1: sharper treatment of the simplified model.

- We estimate the beliefs $\widehat{\mathbf{b}}_t$ from contexts using a spectral method, with confidence bounds;
- We design a LinUCB-style algorithm using the estimated beliefs and prove the high-probability regret bound $R_T = \widetilde{\mathcal{O}}(T^{3/4})$;
- The extra $T^{1/4}$ factor comes from estimating HMM beliefs; with known beliefs, we recover the usual $\widetilde{\mathcal{O}}(\sqrt{T})$ rate.

Complex Model: State-Dependent Rewards and Main Challenges

The paper's main model keeps the realized latent state in the reward model:

$$r_t(a) = \varphi(a, \mathbf{x}_t)^\top \boldsymbol{\theta}^\star_{h_t} + \eta_t(a).$$

This model is more natural: the latent states directly select the reward parameters. However, it no longer admits a direct reduction to a standard linear contextual bandits. For any fixed action a, the context-only belief gives

$$\mathbb{E}\big[r_t(a) \mid \mathbf{x}_{1:t}\big] = \sum_{h \in [H]} \mathbf{b}_t(h) \varphi(a, \mathbf{x}_t)^\top \boldsymbol{\theta}^\star_h,$$

The key difficulty: adaptivity

The equality above need not hold for the adaptive action a_t: a_t is chosen using past rewards, which carry information about past latent states. Hence a_t is not measurable with respect to $\sigma(\mathbf{x}_{1:t})$, and in general

$$\mathbb{P}(h_t = h \mid \mathbf{x}_{1:t}, a_t) \neq \mathbf{b}_t(h).$$

This is the central difficulty in the complex state-dependent model.

Contrib. #2: We formalize this adaptivity gap and design a staged LinUCB algorithm with sublinear regret for the latent-state-dependent reward model.

Why Standard LinUCB on Beliefs Is Not Enough

A natural idea is to estimate the beliefs $\widehat{\mathbf{b}}_t$ and run standard LinUCB with the belief-weighted feature $\widehat{\mathbf{b}}_t \otimes \varphi(a, \mathbf{x}_t)$. This works for the belief-dependent model, but not directly for the state-dependent model.

Where the standard proof breaks.

- The action a_t is selected using past rewards;
- Past rewards carry information about past latent states;
- Hence the Gram matrix and reward estimates may become statistically entangled with latent-state fluctuations.

Algorithmic principle.

Estimate beliefs from contexts only + update reward parameters only periodically.

The first choice avoids feeding reward-action information into the HMM filter. The second creates stages in which actions are measurable with respect to a controlled filtration:

$$\mathcal{U}_t = \sigma\left(\mathbf{x}_{1:t},\ \left(\widehat{\boldsymbol{\theta}}_{s\ell}\right)_{s \leqslant s_t - 1}\right).$$

Proposed Algorithm: Staged LinUCB on Estimated Beliefs

A belief-estimation subroutine $\mathcal{B}$ (spectral method Anandkumar et al., 2012) returns context-only beliefs $\widehat{\mathbf{b}}_t$ from $\mathbf{x}_{1:t}$, with

$$\left\| \widehat{\mathbf{b}}_t - \mathbf{b}_t \right\|_1 \leqslant U_{\text{belief}}(t, \delta), \qquad \sum_{t \in [T]} U_{\text{belief}}(t, \delta) = \widetilde{\mathcal{O}}\left(T^{1/2}\right).$$

At stage endpoints, estimate the stacked reward parameter $\boldsymbol{\theta}^\star$:

$$G_t \stackrel{\text{def}}{=} \sum_{\tau=1}^{t} \left(\widehat{\mathbf{b}}_\tau \otimes \boldsymbol{\varphi}(a_\tau, \mathbf{x}_\tau)\right)\left(\widehat{\mathbf{b}}_\tau \otimes \boldsymbol{\varphi}(a_\tau, \mathbf{x}_\tau)\right)^\top + \lambda \mathbf{I}_{dH}\,,$$
$$\widehat{\boldsymbol{\theta}}_t = G_t^{-1} \sum_{\tau=1}^{t} \left(\widehat{\mathbf{b}}_\tau \otimes \boldsymbol{\varphi}(a_\tau, \mathbf{x}_\tau)\right) r_\tau(a_\tau)\,.$$

During stage s, keep $\widehat{\boldsymbol{\theta}}_{(s-1)\ell}$ frozen and choose

$$a_t \in \operatorname*{argmax}_{a \in \mathcal{A}} \sum_{h \in [H]} \widehat{\mathbf{b}}_t(h) \boldsymbol{\varphi}(a, \mathbf{x}_t)^\top \widehat{\boldsymbol{\theta}}_{(s-1)\ell, h} + \varepsilon_{t,a}.$$

Why staging? Actions in a stage use only reward estimates computed before the stage begins. This prevents rewards collected inside the current stage from immediately affecting action selection. Hence a_t is measurable with respect to the controlled filtration $\mathcal{U}_t$.

Regret Bound and Main Error Terms

High-probability regret bound. For the state-dependent model, under the HMM forgetting condition, the proposed algorithm achieves

$$\boxed{R_T = \widetilde{\mathcal{O}}(T^{7/8})}$$

with high probability.

Where does the rate come from? Optimism reduces the regret analysis to confidence radii, but the estimation error now contains three components:

$$\text{belief error} \quad + \quad \text{reward noise} \quad + \quad \text{latent state fluctuation}.$$

A key step is controlling the hardest component of the latent-state fluctuation, namely the gap between

$$\sum_{t=1}^{T} \varphi(a_t, \mathbf{x}_t)^\top \boldsymbol{\theta}^\star_{h_t} \qquad \text{and} \qquad \sum_{t=1}^{T} \varphi(a_t, \mathbf{x}_t)^\top \sum_{h \in [H]} \mathbb{P}(h_t = h \mid \mathcal{U}_t)\, \boldsymbol{\theta}^\star_h.$$

This gap is bounded by combining Markov's inequality with exponential forgetting of the HMM.

Reference

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in Neural Information Processing Systems (NeurIPS'11)*, 24, 2011.

Animashree Anandkumar, Daniel Hsu, and Sham M. Kakade. A method of moments for mixture models and hidden Markov models. In *Proceedings of the 25th Annual Conference on Learning Theory (COLT'2012)*, volume 23 of PMLR, pages 33.1–33.34, 2012.

Elliot Nelson, Debarun Bhattacharjya, Tian Gao, Miao Liu, Djallel Bouneffouf, and Pascal Poupart. Linearizing contextual bandits with latent state dynamics. In *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence (UAI'22)*, volume 180 of PMLR, pages 1477–1487, 2022.