Non-Stationary Latent Auto-Regressive Bandits

Anna L. Trella, Walter Dempsey, Asim H. Gazi, Ziping Xu, Finale Doshi-Velez, Susan A. Murphy

Keywords: bandit algorithms, non-stationarity

Summary

For the non-stationary multi-armed bandit (MAB) problem, many existing methods allow a general mechanism for the non-stationarity, but rely on a budget for the non-stationarity that is sub-linear to the total number of time steps T. In many real-world settings, however, the mechanism for the non-stationarity can be modeled, but there is no budget for the non-stationarity. We instead consider the non-stationary bandit problem where the reward means change due to a latent, auto-regressive (AR) state. We develop Latent AR LinUCB (LARL), an online linear contextual bandit algorithm that does not rely on the non-stationary budget, but instead forms predictions of reward means by implicitly predicting the latent state. The key idea is to reduce the problem to a linear dynamical system which can be solved as a linear contextual bandit. In fact, LARL approximates a steady-state Kalman filter and efficiently learns system parameters online. We provide an interpretable regret bound for LARL with respect to the level of non-stationarity in the environment. LARL achieves sub-linear regret in this setting if the noise variance of the latent state process is sufficiently small with respect to T. Empirically, LARL outperforms various baseline methods in this non-stationary bandit problem.

Contribution(s)

1. This paper introduces Latent AR LinUCB (LARL), an online algorithm designed for non-stationary MABs where the non-stationarity is due to a latent, auto-regressive (AR) state. LARL forms predictions of reward means by implicitly predicting the latent state using past rewards and actions. This strategy can be seen as an approximation of a steady-state Kalman filter with ground-truth system parameters.
   Context: The setting we consider is motivated by real-world applications where the non-stationary mechanism can be modeled by a latent state, but there is no budget for the non-stationarity. Existing approaches that consider similar settings rely on the latent state being discrete (Hong et al., 2020; Nelson et al., 2022) or require knowing the ground truth parameters or quality historical data to recover parameters (Liu et al., 2023; Chen et al., 2024).
2. We present an interpretable regret bound for LARL against the dynamic oracle. The regret bound allows practitioners to interpret the performance of LARL with respect to the level of non-stationarity in the environment and the complexity of learning parameters online.
   Context: Sub-linear regret with respect to the dynamic oracle is only possible in environments with a budget for non-stationarity that is sub-linear in T. For example, Besbes et al. (2014) assume a finite constant (variation budget) of how much the mean rewards can change over time and Garivier & Moulines (2011) assume a finite number of changes to the mean reward. We show that in our setting, LARL achieves sub-linear regret if the noise variance on the latent state process is sufficiently small with respect to T.
3. We demonstrate that LARL can outperform (achieve lower regret) against various stationary and non-stationary baselines in the non-stationary bandit environment where reward means change due to a latent AR state.
   Context: We consider cumulative regret across time and pairwise comparisons of methods in terms of total cumulative regret. To offer a fair comparison, baseline methods were only considered if they implemented an online learning strategy. For large values of k, the performance of LARL approaches the performance of baseline methods because the algorithm needs to fit more parameters, and thus requires more data to learn effectively.

Non-Stationary Latent Auto-Regressive Bandits

Anna L. Trella[1], Walter Dempsey[2], Asim H. Gazi[1], Ziping Xu[1], Finale Doshi-Velez[1], Susan A. Murphy[1]
annatrella@g.harvard.edu, wdem@umich.edu, agazi@seas.harvard.edu, zipingxu@fas.harvard.edu, finale@seas.harvard.edu, samurphy@fas.harvard.edu

[1]**School of Engineering and Applied Sciences, Harvard University, Cambridge, MA USA**
[2]**Department of Biostatistics, University of Michigan, Ann Arbor, MI USA**

Abstract

For the non-stationary multi-armed bandit (MAB) problem, many existing methods allow a general mechanism for the non-stationarity, but rely on a budget for the non-stationarity that is sub-linear to the total number of time steps T. In many real-world settings, however, the mechanism for the non-stationarity can be modeled, but there is no budget for the non-stationarity. We instead consider the non-stationary bandit problem where the reward means change due to a latent, auto-regressive (AR) state. We develop Latent AR LinUCB (LARL), an online linear contextual bandit algorithm that does not rely on the non-stationary budget, but instead forms predictions of reward means by implicitly predicting the latent state. The key idea is to reduce the problem to a linear dynamical system which can be solved as a linear contextual bandit. In fact, LARL approximates a steady-state Kalman filter and efficiently learns system parameters online. We provide an interpretable regret bound for LARL with respect to the level of non-stationarity in the environment. LARL achieves sub-linear regret in this setting if the noise variance of the latent state process is sufficiently small with respect to T. Empirically, LARL outperforms various baseline methods in this non-stationary bandit problem.

1 Introduction

In the classical formulation of the stochastic multi-armed bandit (MAB) problem (Lattimore & Szepesvári, 2020), the rewards are assumed to be independently and identically drawn from a fixed distribution. In the non-stationary formulation (Auer et al., 2002), the reward means, instead, change over time. While many existing approaches for non-stationary bandits allow an arbitrary mechanism for the non-stationarity, they rely on some budget to the non-stationarity that is sub-linear to the total number of time steps T. For example, in Besbes et al. (2014) there is a variation budget for the amount of change in the mean rewards, and in Garivier & Moulines (2011) there is a budget for the number of changes. In contrast, for many real-world applications, the non-stationarity mechanism can be modeled as a latent state with temporal dependencies, but with restless non-stationarity. For example, in mobile health applications, bandit algorithms are used to optimize notifications to maximize users' health outcomes (rewards). User burden from using the app is an evolving latent process with temporal dependencies and can cause the health outcome to decline over time (non-stationarity).

Motivated by these real-world settings, we study a non-stationary bandit problem with a realistic source of non-stationarity. In this problem, changes in the mean reward of the arms over time are due to some latent, auto-regressive (AR) state of order k. This problem is represented by the graphical model in Figure 1. Such a latent state causes smooth changes to the mean rewards as opposed to abrupt changes; however, the variation budget or the budget on the number of changes could scale linearly with T.

Our approach to solving the non-stationary bandit problem in Figure 1 leverages the graphical structure and reduces the problem instead to the well-studied problem of linear dynamical systems (Section 3.3); we then show that the linear dynamical system can be solved as a linear contextual bandit (Section 3.4). By leveraging the structure of the non-stationarity, we can offer a finer theoretical analysis and design a more specific algorithm that can outperform general non-stationary algorithms.

Contributions. We propose Latent AR LinUCB or LARL (Algorithm 1), an online linear contextual bandit algorithm that maintains good reward predictions by using past history to predict the current latent state and to learn parameters online. The reward model maintained by LARL can be seen as an approximation to the steady-state Kalman filter with access to ground-truth system parameters. We present an interpretable regret bound for LARL against the dynamic oracle (Theorem 4.2). In our setting, LARL achieves sub-linear regret if the noise variance of the latent state process is sufficiently small with respect to the total number of time steps T. We validate in simulation studies (Section 5) that LARL outperforms various baseline methods in the non-stationary latent AR environment.

2 Related Works

2.1 Non-Stationary Bandits

Non-stationary bandits (Auer et al., 2002) extend the standard bandit problem to one with reward means changing over time. Many approaches have been proposed for non-stationary bandits including change point detection (Mellor & Shapiro, 2013), sliding window (Garivier & Moulines, 2011; Cheung et al., 2019; Trovo et al., 2020), restarting (Besbes et al., 2014; Viappiani, 2013), and discounting the effect of past observations (Kocsis & Szepesvári, 2006; Garivier & Moulines, 2011; Raj & Kalyani, 2017). A majority of these methods were developed for an arbitrary mechanism for controlling the non-stationarity, but rely on a budget for the amount of changes that is sub-linear in T. For example, there is a total budget for the amount of change (Besbes et al., 2014) or to the number of changes (Garivier & Moulines, 2011). In contrast, our setting has a specific mechanism controlling the non-stationarity (i.e., a latent AR state), but the budget for the non-stationarity could scale linearly with T. Others have also formulated non-stationarity through a latent state. These works propose methods for maintaining a posterior belief over the latent state and acting according to it (Hong et al., 2020; Nelson et al., 2022). Nelson et al. (2022) rely on the latent state being discrete, as the dimension of the linear contextual bandit is the cardinality of the set of latent state values. This approach is not applicable in our setting where the latent state is continuous. Hong et al. (2020) use particle filtering to sample from the joint posterior of the latent state and model parameters. However, their approach would be computationally challenging in our setting, as the number of particles needed increases with the number of latent state values. Also similar to our work is Gornet et al. (2022) which implements a similar reduction to a linear contextual bandit; however, their algorithm heavily relies on exogenous context to predict the latent state, which is not present in our setting.

2.2 AR Bandits

Autoregressive (AR) processes (Brockwell & Davis, 2009) are studied extensively across many fields to model temporal dependencies in many real-world processes. Some bandit formulations involve rewards evolving auto-regressively, rather than a latent state, with fixed reward means and known AR order k (Bacchiocchi et al., 2022). Due to the fixed reward means, these settings are *stationary* bandits, which allows for the direct application of standard linear bandit theory. While some have proposed similar settings where the non-stationarity is dictated by an AR process (Liu et al., 2023; Chen et al., 2024), these papers assume either access to ground-truth parameters or quality offline data to learn these parameters. Instead, we are interested in learning AR and reward parameters online. Furthermore, the predictive sampling method in Liu et al. (2023) is developed for the Bayesian framework and their main goal is to compare to traditional Thompson sampling. At first glance, one may notice that the AR(1) setting presented in Chen et al. (2024) is very similar. However, their method relies on a stronger assumption that the agent observes the true mean reward for the same

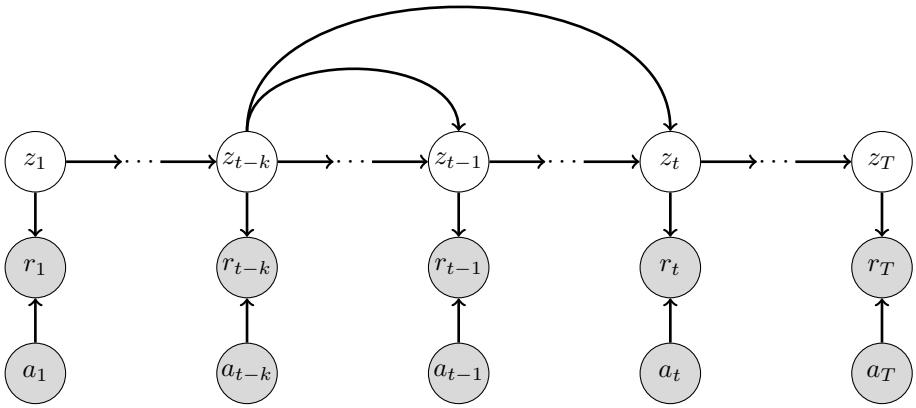


Figure 1: Graphical Model for Non-Stationary Latent Auto-regressive Bandits.

action taken at the previous time step. In our setting, the agent never observes the exact mean reward for any action.

3 Problem Setting

3.1 Notation

We introduce the following notation used throughout the paper. For some vector $v \in \mathbb{R}^d$, we use $\|v\| = \sqrt{\langle v, v \rangle}$ to denote the L2-norm of v and $v^\top$ to denote the transpose of v. $\mathbf{1}$ denotes a vector of all 1s and e_j denotes the standard basis vector with 1 in the jth component and 0s elsewhere. For a square matrix $M \in \mathbb{R}^{d \times d}$, M^{-1} denotes the inverse of M. $\lambda_{\max}(M)$ is the largest eigenvalue of M. $\|M\|_{\text{op}} = \sigma_{\max}(M)$ is the operator norm or largest singular value of M. If M is positive semi-definite (PSD), then $M^{1/2}$ denotes the square root of M such that $M^{1/2}M^{1/2} = M$, and $\|v\|_M^2 = v^\top M v$ denotes the square of the weighted L2-norm of v. We use $[i, j]$ to denote the set of positive integers from i to j, inclusive. $\mathbb{I}[\cdot]$ denotes the indicator function. $\mathcal{H}_{t-1}$ is the entire history of information observed up to, but not including, time t.

3.2 Non-Stationary Latent Auto-regressive Bandits

We consider a non-stationary multi-armed bandit environment (Definition 3.1) where the true underlying reward depends on some latent state $z_t \in \mathbb{R}$ that evolves according to an auto-regressive (AR) process of order at most k. See Figure 1 for the graphical structure.

Definition 3.1. (Non-Stationary Latent Auto-regressive Bandit) Let $\mathcal{A} \subset \mathbb{N}$ be the action space and initial latent states $[z_0, ..., z_{k-1}] \sim \mathcal{N}_k(\mu_0, \Sigma_0)$. The interaction between the environment and the agent is as follows. For every time step $t \in [k, T]$:

1. The environment generates latent state z_t of the form:

$$z_t = \gamma_0 + \sum_{j=1}^{k} \gamma_j z_{t-j} + \xi_t, \quad \xi_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_z^2), \tag{1}$$

   where $\gamma_0, \gamma_1, ..., \gamma_k \in \mathbb{R}$.

2. The agent selects action $a_t \in \mathcal{A}$ without observing z_t.
3. The environment then generates reward r_t given latent state z_t and action a_t, $r_t(a_t)$, where:

$$r_t(a) = \mu_a + \beta_a z_t + \epsilon_t(a), \quad \epsilon_t(a) \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \beta_a^2 \sigma_r^2) \tag{2}$$

where $\mu_a, \beta_a, \epsilon_t(a) \in \mathbb{R}$ depends on the action a and $\epsilon_t(a)$ is independent across actions and time steps.

4. The agent observes r_t.

Notice that in our setting, z_t is not impacted by the action selected in the previous time step. Also, notice that in Equation 2 the noise variance of the reward has an exact structure. This structure is needed to simplify the reduction in Lemma 3.2. But in practice, the algorithm (Algorithm 1) we present later does not require this noise structure. To solve this non-stationary bandit problem, a natural approach is to form good predictions of z_t and therefore good predictions of the reward means for each action. However, with no exogenous context available, the only observations one is given is the current history $\mathcal{H}_{t-1}$ consisting of past actions and rewards. We will see in later sections that the method we develop can still perform well in such an environment, despite having limited information.

3.3 Connecting Latent AR Bandits With Linear Dynamical Systems

To assist with the reduction to a linear contextual bandit in Section 3.4, we first show that the latent AR bandit environment in Definition 3.1 is a specific case of a linear dynamic system (LDS) with Gaussian noise. See Appendix C for a review of linear dynamical systems.

Lemma 3.2. ***(Linear Dynamical System)*** *The latent state process (Equation 1) and the reward function (Equation 2) in Definition 3.1 form a special case of a linear dynamical system with Gaussian noise. The system has state vector $\vec{z}_t \in \mathbb{R}^k$ which incorporates the most recent k latent state realizations and measurement $y_t = \frac{r_t - \mu_{a_t}}{\beta_{a_t}} \in \mathbb{R}$.*

$$\vec{z}_t = \Gamma \vec{z}_{t-1} + w_t, \quad w_t \sim \mathcal{N}_k(\gamma_0 e_1, W) \tag{3}$$

$$y_t = C\vec{z}_t + v_t, \quad v_t \sim \mathcal{N}(0, \sigma_r^2) \tag{4}$$

$$r_t(a) = c_a^\top \vec{z}_t + \mu_a + \epsilon_t(a), \quad \epsilon_t(a) \sim \mathcal{N}(0, \beta_a^2 \sigma_r^2) \tag{5}$$

where

$$\vec{z}_t := \begin{bmatrix} z_t & z_{t-1} & \cdots & z_{t-k+1} \end{bmatrix}^\top \in \mathbb{R}^k$$

See Appendix A.1 for exact forms for Γ, W, C, c_a.

Proof. See Appendix A.1. □

Lemma 3.2 shows that we can rewrite the process as a linear dynamical system with Gaussian noise with a specific form for the measurement model. Since this LDS is in companion form (Bellman & Åström, 1970), the LDS satisfies structural identifiability (Bellman & Åström, 1970), and therefore is observable (Assumption C.1).

A natural approach to predicting the mean reward for each action is first to predict $\vec{z}_t$. Since Assumption C.1 holds, one may be motivated to use the steady-state Kalman filter (Appendix C.2) to infer $\vec{z}_t$. Given the ground-truth system parameters $\Gamma, C, \gamma_0, W, \sigma_r^2$, the Kalman filter prediction $\tilde{z}_t$ is the optimal (least mean square) estimate for latent state $\vec{z}_t$ (Kailath et al., 2000). However, we do not assume agents have access to ground-truth parameters or quality batch data for learning them offline (Ljung, 1999). While one can learn system parameters online (Annaswamy & Fradkov, 2021; Subbarao et al., 2016), we show in the following sections that it is not required to explicitly learn system parameters for forming good mean reward predictions. Instead, the reduction to a linear contextual bandit allows us to implicitly learn system parameters by learning a single reward parameter (Lemma 3.3) and to leverage the well-established theory on linear bandits for analyzing regret.

3.4 Reduction to a Linear Contextual Bandit

To re-frame the problem as a linear contextual bandit, we use the converted LDS (Lemma 3.2) to show that the reward (Equation 2) can be re-written in a linear form of past history and the steady-state Kalman filter prediction of state. This is a modified version of the decomposition in Gornet et al. (2022).

Lemma 3.3. *(Linear Contextual Bandit Reduction) Let $\mathcal{H}_{t-1} := \sigma(a_1, r_1, ..., a_{t-1}, r_{t-1})$ be the smallest σ-algebra generated by the current history observed up to time t, $\tilde{z}_t := z_{t|t-1} = \mathbb{E}[\vec{z}_t|\mathcal{H}_{t-1}]$ be the steady-state Kalman filter for $\vec{z}_t$ and let $\tilde{r}_t(a) = \mathbb{E}[r_t(a)|\mathcal{H}_{t-1}] = c_a^\top \tilde{z}_t + \mu_a$. For a choice of $s > 0$ that controls the number of past time steps to include in the context, there exists some $\theta_a \in \mathbb{R}^{2s\cdot|\mathcal{A}|+1}$ where the reward (Equation 2) for action a is:*

$$r_t(a) = \Phi_t(s)^\top \theta_a + b_t(a, s) + \varepsilon_{a;t} \tag{6}$$

where

$$\Phi_t(s) := \Phi(R_t, A_t) = \begin{bmatrix} R_t & A_t & 1 \end{bmatrix}^\top \in \mathbb{R}^{2s\cdot|\mathcal{A}|+1} \tag{7}$$

$$b_t(a, s) := \langle c_a, (\Gamma - \Gamma K C)^s \tilde{z}_{t-s} \rangle \tag{8}$$

$$\varepsilon_{a;t} := r_t(a) - \tilde{r}_t(a) = \langle c_a, \vec{z}_t - \tilde{z}_t \rangle + \epsilon_t(a) \sim \mathcal{N}(0, c_a^\top P c_a + \beta_a^2 \sigma_r^2) \tag{9}$$

$$R_t := \begin{bmatrix} r_{t-s} e_{a_{t-s}}^\top & \cdots & r_{t-1} e_{a_{t-1}}^\top \end{bmatrix}^\top \in \mathbb{R}^{s\cdot|\mathcal{A}|}$$

$$A_t := \begin{bmatrix} e_{a_{t-s}}^\top & \cdots & e_{a_{t-1}}^\top \end{bmatrix}^\top \in \mathbb{R}^{s\cdot|\mathcal{A}|}$$

Proof. See Appendix A.2. □

Equation (6) shows a standard linear contextual bandit problem with an additional bias term $b_t(a, s)$. $\Phi_t(s)$ is the current context obtained from a feature mapping of the s most recent previous actions and rewards; θ_a incorporates the underlying LDS parameters and is the parameter to learn. Notice that $\varepsilon_{a;t}$ is independent of history and therefore has mean 0 conditioned on history $\mathcal{H}_{t-1}$.

Lemma 3.3 justifies the rationale of solving the non-stationary bandit problem through a contextual bandit algorithm, say LinUCB. By selecting s, one is selecting the number of recent time steps used to predict the current mean reward. If one had access to ground-truth θ_a and dynamically sets $s = t$, then such an agent's performance is the same as an agent that predicts the mean reward using a steady-state Kalman filter (Appendix C.2) with ground-truth parameters. Recall that with the true underlying parameters, the Kalman filter estimate $\tilde{z}_t$ is the optimal estimate for latent state z_t. However, because we do not assume access to ground-truth parameters and must learn parameters online, one must set s to balance bias and variance, which allows for a good approximation of the Kalman filter estimate. This bias-variance trade-off, controlled by s, also appears in the regret bound (Theorem 4.2) presented later. This linear bandit reduction is also desirable because an agent only needs to specify a value for s and does not need to know the ground-truth AR order k, the initial state $\vec{z}_0$, nor the noise variances in practice for forming an estimator for θ_a.

3.5 Assumptions

We make the following regularity assumptions on the environment.

Assumption 3.4. (Stability and Boundedness of the AR Process) For the parameters $\gamma_0, \gamma_1, ..., \gamma_k$ of the latent AR process, $|\sum_{j=1}^{k} \gamma_j| < 1$ and $|\gamma_0| \leq c$ for some $c < +\infty$

Assumption 3.5. (Bounded Reward Parameter) For every action $a \in \mathcal{A}$, $\|\theta_a\| \leq S_a$ for $S_a \in \mathbb{R}^+$

Assumption 3.4 is standard for AR processes. Assumption 3.5 is standard for theoretical results for linear bandits (Abbasi-Yadkori et al., 2011). More importantly, Assumption 3.4 implies the stability of state transition matrix Γ, a common assumption for LDS (Bertsekas, 2012). Namely, the parameters $\gamma_1, ..., \gamma_k$ of the latent AR process form Γ such that $|\lambda_{\max}(\Gamma)| < 1$. The stability of Γ is needed for the bias term $b_t(a, s)$ in Equation 6 to decrease with large s as $|\lambda_{\max}(\Gamma)| < 1 \implies |\lambda_{\max}(\Gamma - \Gamma KC)| < 1$ (Anderson & Moore, 2005).

3.6 Regret

We define regret with respect to the dynamic oracle (Besbes et al., 2014), the standard choice in the non-stationary bandit settings. The dynamic oracle observes all information in the environment (including z_t) and then acts optimally with that information. The oracle therefore knows the true reward means $\mathbb{E}[r_t(a)|\vec{z}_t]$ at every time step t for each action a and selects the optimal action for every t: $a_t^* = \arg\max_{a \in \mathcal{A}} \mathbb{E}[r_t(a)|\vec{z}_t]$. The regret with respect to the dynamic oracle is:

$$\text{Regret}(T; \pi) = \sum_{t=1}^{T} \mathbb{E}[r_t(a_t^*) - r_t(a_t)|\vec{z}_t] \tag{10}$$

where a_t is the action selected by the algorithm at time step t following policy π.

Achieving sub-linear regret against the dynamic oracle is not guaranteed without assuming vanishing non-stationarity in the environment. Since our latent AR setting does not make this assumption, our goal is to provide an interpretable regret bound with respect to the non-stationarity in the environment. For a full discussion of the regret definition and the difficulty in achieving sub-linear regret, see Appendix D.

4 LinUCB Algorithm for Latent AR Bandits

We present our algorithm coined Latent AR Bandit LinUCB (LARL), shown in Algorithm 1. LARL is based on the LinUCB algorithm (Li et al., 2010; Abbasi-Yadkori et al., 2011) for linear contextual bandits, modified to handle our non-stationary environment. For a fixed choice of $s > 0$, LARL uses rewards and actions from the s most recent time steps to form current context $\Phi_t(s)$ (Equation 7). By carefully constructing the context $\Phi_t(s)$ this way, we can implicitly predict the latent state z_t and efficiently learn the parameters θ_a online.

As a review, LinUCB maintains regularized least squares (RLS) estimators for each action a at time step t:

$$\hat{\theta}_{a,t} = V_{a,t}^{-1} b_{a,t} \tag{11}$$

where $V_{a,t} = \lambda I + \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)\Phi_j(s)^\top$, and $b_{a,t} = \sum_{j=1}^{t} \mathbb{I}[a_j = a] r_j \Phi_j(s)$

For action-selection, LinUCB forms some confidence set $\mathcal{C}_{a,t-1}$ using the most recent RLS estimator $\hat{\theta}_{a,t-1}$ for every action a and selects the action with the highest confidence bound on its reward:

$$a_t = \arg\max_{a \in \mathcal{A}} \max_{\theta_a \in \mathcal{C}_{a,t-1}} \Phi_t(s)^T \theta_a \tag{12}$$

Algorithm 1 Latent AR LinUCB

Inputs: $V_{a,0} = \lambda I, b_{a,0} = \vec{0}, \theta_{a,0} = \vec{0}$ for all $a \in \mathcal{A}$, $s \in \mathbb{N}$
for $t = 1, 2, ..., T$ **do**
  Use rewards and actions from the most recent s time steps $r_{t-s}, ..., r_{t-1}, a_{t-s}, ..., a_{t-1}$ to form current context $\Phi_t(s)$ defined in Equation 7.
  For each action a, use most recent RLS estimator $\hat{\theta}_{a,t-1}$ to form confidence set $\mathcal{C}_{a,t-1}$.
  Select action a_t:

$$a_t = \arg\max_{a \in \mathcal{A}} \max_{\theta_a \in \mathcal{C}_{a,t-1}} \Phi_t(s)^T \theta_a$$

  Execute action a_t and observe r_t.
  Update history $\mathcal{H}_t = \{(\Phi(s)_{t'}, a_{t'}, r_{t'})\}_{t'=1}^t$
  Update RLS estimator for action a_t as in Equation 11:
  $\hat{\theta}_{a_t,t} = V_{a_t,t}^{-1} b_{a_t,t}$,
  $V_{a,t} = V_{a,0} + \sum_{j=1}^t \mathbb{I}[a_j = a] \Phi_j(s) \Phi_j(s)^\top$,
  $b_{a,t} = b_{a,0} + \sum_{j=1}^t \mathbb{I}[a_j = a] r_j \Phi_j(s)$
end for

Notably, our algorithm requires no knowledge of the ground-truth AR order k, initial state $\vec{z}_0$, nor noise variances of the state and reward processes. Notice also that for $s = 0$, Algorithm 1 reduces to the standard UCB method for stationary MABs.

To show theoretical results for LARL, we first show that for each action a, θ_a lies in the confidence set $\mathcal{C}_{a,t}$ for all t (Lemma 4.1). The radius of the confidence set formed by LARL is an enlarged version of the one presented in Abbasi-Yadkori et al. (2011) to account for the bias term. Finally, we use Lemma 4.1 to prove the regret bound in Theorem 4.2.

4.1 Confidence Set

Lemma 4.1. *[Confidence Set for Latent AR Bandits] Suppose Assumptions 3.4 and 3.5 holds. For given action a, with probability at least $1 - \delta$ where $\delta \in (0, 1)$, the true parameter θ_a (Definition 6) is in the confidence ellipsoid $\mathcal{C}_{a,t}$ centered at $\hat{\theta}_{a,t}$ (Equation 11), for all $t \in [T]$:*

$$\mathcal{C}_{a,t} := \{\theta_a \in \mathbb{R}^d \mid \|\hat{\theta}_{a,t} - \theta_a\|_{V_{a,t}} \leq \beta_{a,t}(\delta)\} \tag{13}$$

where

$$\beta_{a,t}(\delta) = R\sqrt{(2s|\mathcal{A}| + 1) \log\left(\frac{1 + n_{a,t} L(s, \delta/2)/\lambda}{\delta/2}\right)} + \sqrt{\lambda} S_a + \tau(a, s)_t \sqrt{\sum_{j=1}^t \mathbb{I}[a_j = a] b_j(a, s)^2} \tag{14}$$

where $n_{a,t} = \sum_{j=1}^t \mathbb{I}[a_j = a]$ and

$$\tau(a, s)_t = \sqrt{\sum_{j=1}^t \mathbb{I}[a_j = a] \|\Phi_j(s)\|^2_{V_{a,t}^{-1}}} \tag{15}$$

Proof. See Appendix A.3 □

4.2 Regret Bound

We now derive a regret bound for LARL (Algorithm 1). The main proof idea is to introduce an intermediate agent that knows the ground-truth parameters and uses the steady-state Kalman filter

prediction to select actions. We are able to add and subtract the mean reward obtained by such an agent to the instantaneous regret. The instantaneous regret therefore decomposes into the regret of the dynamic oracle against the intermediate agent and the regret of the intermediate agent against LARL.

Theorem 4.2. *Suppose all the assumptions mentioned in Lemma 4.1 hold. With probability at least $1-\delta$ where $\delta \in (0,1)$, the regret of Algorithm 1 in the non-stationary latent AR bandit environment (Definition 3.1) is bounded as follows:*

$$\textit{Regret}(T;\pi_{\textit{LARL}}) \leq 8\max_a \|c_a\| \sqrt{\frac{\sigma_z^2}{1-\sigma_{\max}(\Gamma)^2}} \sqrt{2(k+\log(3/\delta))} \cdot T$$
$$+2\beta_T(2\delta/3)\sqrt{\sum_{t=1}^{T} \|\Phi_t(s)\|^2_{V^{-1}_{a_t,t-1}}} \sqrt{T} + 2\sum_{t=1}^{T} \max_a |b_t(a,s)|$$

for $\beta_T(\delta') = \max_a \beta_{a,T-1}(\delta')$ (Equation 14)

Proof. See Appendix A.4 □

The first term of the regret represents the gap between the optimal decision-making based on the Kalman filter prediction $\tilde{z}_t$ and the dynamic oracle. This term also captures the non-stationarity of the environment as controlled by σ_z^2, the noise variance of the latent AR process. The second term represents the complexity of learning to behave like the optimal policy based on Kalman filtering. The third term captures the bias in the reward function.

The first term captures how the regret rate is ultimately dependent on σ_z^2. For a fixed T in environments where $\sigma_{\max}(\Gamma) \leq 1-\epsilon$ for $\epsilon > 0$ and $\sigma_z^2 = T^{c-2}$ for some constant $c < 2$, our algorithm achieves sub-linear regret. For example, if $\sigma_z^2 = \frac{1}{T}$, then the regret is on the order of $\sqrt{T}$. This is congruent with a variety of other non-stationary bandit formulations where the non-stationarity budget is sub-linear in T. See Appendix D.1 for more details. However, if $c \geq 2$, then σ_z^2 is too large and the regret is not sub-linear. This is because for large σ_z^2, even with ground-truth parameters, the most optimal prediction $\tilde{z}_t$ used for predicting $\tilde{r}_t(a)$, can be far away from the realization of z_t and therefore r_t.

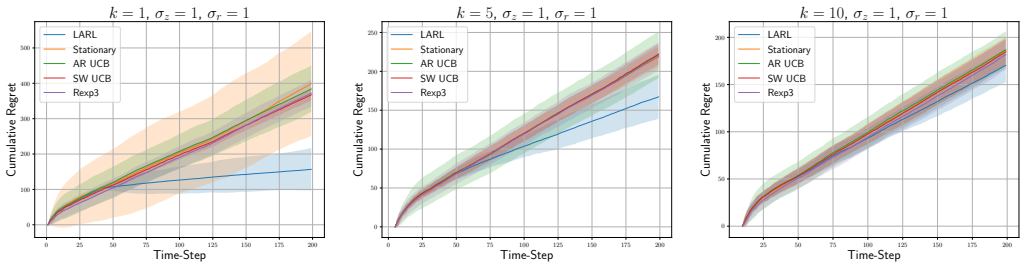
The second and third terms describe the bias-variance trade-off the algorithm designer makes with the choice of s. For large s the bias decreases, as $s \to \infty \implies b_t(a,s) \to 0$ for any a, however, the dimensionality of $\Phi_t(s)$ increases (Lemma A.1) which increases the variance. See Appendix E.1 for simulations that verify this trade-off empirically.

5 Experiments

Through simulations, we highlight how our proposed algorithm LARL (Algorithm 1) can outperform various stationary and non-stationary baselines. We assess performance based on cumulative regret with respect to the dynamic oracle (Section 3.6). Additional experiments and results can be found in Appendix E.

For all experiments, we set $T = 200$. We consider 2 actions $\mathcal{A} = \{0,1\}$, reward parameters $\mu_0, \mu_1 = [0,0], \beta_0, \beta_1 = [-1,1], \gamma_0 = 0, \sigma_z = 1$, and $\sigma_r = 1$. $\gamma_1, ..., \gamma_k$ are drawn randomly from a uniform distribution and post-processed to ensure Assumption 3.4 holds. We consider environments that vary by the AR order k. For each environment variant, we simulate 100 Monte-Carlo trials and in each trial, the k values in the initial state $\vec{z}_0$ are drawn randomly in every trial.

We test the performance of LARL against various stationary and non-stationary baselines. The competing baselines are: (a) “Stationary”, standard UCB which treats the environment as a stationary multi-armed bandit, (b) “AR UCB" (Bacchiocchi et al., 2022), the UCB algorithm developed for stationary AR environments, (c) “SW UCB” (Garivier & Moulines, 2011), the Sliding Window UCB algorithm, (d) “Rexp3” (Besbes et al., 2014), which runs the Exp3 algorithm with restarts.



(a) (b) (c)

Figure 2: Our algorithm LARL (blue), with s chosen using BIC after a period of pure exploration, consistently achieves lower cumulative regret (Equation 10) over time against various baseline methods. Line is the average and shaded region is $\pm$ standard deviation across 100 Monte Carlo simulated trials.

To prove theoretical results, we left s arbitrary. To select s in simulations, we implement an "exploration" period that selects actions randomly up to time step t'. Then using the data collected during the exploration period, we choose s based on the Bayesian Information Criterion (BIC) and commit to that s for the rest of the time-steps. For experiments, we let $t' = \lfloor T/5 \rfloor$. All baseline algorithms that use a UCB-based strategy (including LARL), use regularization parameter $\lambda = 1$ in each environment.

5.1 Results

Figure 2 shows cumulative regret over time. Figure 3 shows pairwise comparisons between algorithms in terms of total cumulative regret. Our method LARL consistently outperforms baseline algorithms developed for stationary and non-stationary environments. Stationary and AR UCB were developed for stationary environments with fixed reward means and cannot adapt to the non-stationarity of the reward means. Although developed for a non-stationary environments, SW UCB and Rexp3 perform similarly to the stationary baselines because SW UCB assumes the mean rewards remain constant over epochs and Rexp3 assumes the non-stationarity has bounded total variation. In the latent AR environment (Definition 3.1), these assumptions are not guaranteed as the non-stationary budget can be linear in T. As k increases, the performance of LARL approaches the performance of baseline methods because the algorithm needs to fit more parameters, and thus requires more data to learn effectively.

6 Conclusion and Future Work

In this paper, we study the non-stationary bandit problem where the mean rewards of actions change over time due to a latent, AR process of order k. We propose a new online algorithm, LARL, that leverages the structure of the non-stationary mechanism in this setting. LARL employs the key idea that this non-stationary bandit can be reduced to a linear dynamical system and solved using a linear contextual bandit with a thoughtful design of the context space. This reduction motivates LARL's linear contextual bandit strategy to implicitly predict the current latent state z_t and efficiently learn parameters online. Furthermore, with a choice of hyperparameter s that trades off bias and variance, one can view the reward model of LARL as a reasonable approximation of a steady-state Kalman filter with ground-truth parameters.

6.1 Future Work

A natural extension is to generalize the present work to contextual bandits with the inclusion of exogenous, observed context features. Although we have proposed an initial approach for selecting

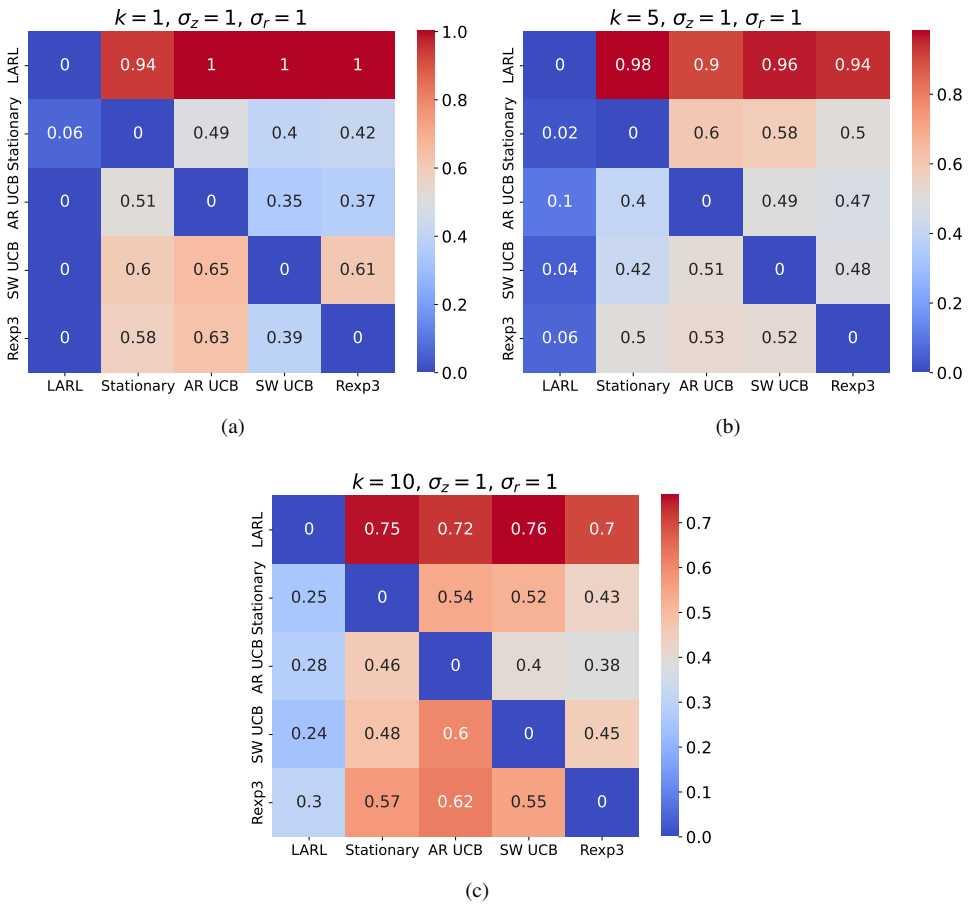


Figure 3: Pairwise comparisons between algorithms in the three variants of the simulation environment where $k = 1, 5, 10$, respectively. Each cell shows the proportion of 100 Monte-Carlo repetitions where the algorithm listed in the row achieved lower cumulative regret than the algorithm listed in the column. Our algorithm LARL (top row) consistently outperforms baseline methods in pairwise comparison.

hyperparameter s, one can explore other approaches such as running an ensemble of agents with different s or finding an optimal s as a function of environment parameters k, T, σ_r, and σ_z. Lastly, to make our ideas clear, we formulated the latent state as a scalar, but one can consider a generalization of our setting where the latent state is multi-dimensional.

Broader Impact Statement

This paper presents work whose goal is to advance the field of reinforcement learning for use in real-world problems such as digital health. The assumptions we make in the paper may be valid for some of these domains and not others.

Acknowledgments

This research was funded by NIH/NIDCR grant UH3DE028723, the Center for Methodologies for Adapting and Personalizing Prevention, Treatment and Recovery Services for SUD and HIV P50 DA054039, and the National Science Foundation grant no. IIS-1750358. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not

necessarily reflect the views of the National Science Foundation. SAM holds concurrent appointments at Harvard University and as an Amazon Scholar. This paper describes work performed at Harvard University and is not associated with Amazon.

References

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

Brian DO Anderson and John B Moore. *Optimal filtering*. Courier Corporation, 2005.

Anuradha M. Annaswamy and Alexander L. Fradkov. A historical perspective of adaptive control and learning. *Annual Reviews in Control*, 52:18–41, 2021. ISSN 1367-5788. DOI: https://doi.org/10.1016/j.arcontrol.2021.10.014. URL https://www.sciencedirect.com/science/article/pii/S1367578821000894.

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

Francesco Bacchiocchi, Gianmarco Genalti, Davide Maran, Marco Mussi, Marcello Restelli, Nicola Gatti, and Alberto Maria Metelli. Autoregressive bandits. *arXiv preprint arXiv:2212.06251*, 2022.

Ror Bellman and Karl Johan Åström. On structural identifiability. *Mathematical biosciences*, 7(3-4): 329–339, 1970.

Dimitri Bertsekas. *Dynamic programming and optimal control: Volume I*, volume 4. Athena scientific, 2012.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Stochastic multi-armed-bandit problem with non-stationary rewards. *Advances in neural information processing systems*, 27, 2014.

Peter J Brockwell and Richard A Davis. *Time series: theory and methods*. Springer science & business media, 2009.

Qinyi Chen, Negin Golrezaei, and Djallel Bouneffouf. Non-stationary bandits with auto-regressive temporal dependency. *Advances in Neural Information Processing Systems*, 36, 2024.

Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Learning to optimize under non-stationarity. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1079–1087. PMLR, 2019.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *International Conference on Algorithmic Learning Theory*, pp. 174–188. Springer, 2011.

Jonathan Gornet, Mehdi Hosseinzadeh, and Bruno Sinopoli. Stochastic multi-armed bandits with non-stationary rewards generated by a linear dynamical system. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 1460–1465. IEEE, 2022.

Joey Hong, Branislav Kveton, Manzil Zaheer, Yinlam Chow, Amr Ahmed, Mohammad Ghavamzadeh, and Craig Boutilier. Non-stationary latent bandits. *arXiv preprint arXiv:2012.00386*, 2020.

Thomas. Kailath, Ali H. Sayed, and Babak. Hassibi. *Linear estimation*. Prentice Hall, first edit edition, 2000. ISBN 9780130224644.

Levente Kocsis and Csaba Szepesvári. Discounted ucb. In *2nd PASCAL Challenges Workshop*, volume 2, pp. 51–134, 2006.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pp. 661–670, 2010.

Yueyang Liu, Xu Kuang, and Benjamin Van Roy. Non-stationary bandit learning via predictive sampling, 2023.

Lennart Ljung. *System Identification: Theory for the User*. Prentice Hall, second edi edition, 1999. ISBN 978-0136566953.

Joseph Mellor and Jonathan Shapiro. Thompson sampling in switching environments with bayesian online change detection. In *Artificial intelligence and statistics*, pp. 442–450. PMLR, 2013.

Elliot Nelson, Debarun Bhattacharjya, Tian Gao, Miao Liu, Djallel Bouneffouf, and Pascal Poupart. Linearizing contextual bandits with latent state dynamics. In *Uncertainty in Artificial Intelligence*, pp. 1477–1487. PMLR, 2022.

Vishnu Raj and Sheetal Kalyani. Taming non-stationary bandits: A bayesian approach. *arXiv preprint arXiv:1707.09727*, 2017.

Alessandro Rinaldo and Shenghao Wu. Lecture notes in advanced statistical theory. https://www.stat.cmu.edu/~arinaldo/Teaching/36709/S19/Scribed_Lectures/Feb21_Shenghao.pdf, 2019.

Sam Roweis and Zoubin Ghahramani. A unifying review of linear gaussian models. *Neural computation*, 11(2):305–345, 1999.

Kamesh Subbarao, Pavan Nuthi, and Ghassan Atmeh. Reinforcement learning based computational adaptive optimal control and system identification for linear systems. *Annual Reviews in Control*, 42:319–331, 2016. ISSN 1367-5788. DOI: https://doi.org/10.1016/j.arcontrol.2016.09.021. URL https://www.sciencedirect.com/science/article/pii/S1367578816300517.

Francesco Trovo, Stefano Paladino, Marcello Restelli, and Nicola Gatti. Sliding-window thompson sampling for non-stationary settings. *Journal of Artificial Intelligence Research*, 68:311–364, 2020.

Paolo Viappiani. Thompson sampling for bayesian bandits with resets. In *Algorithmic Decision Theory: Third International Conference, ADT 2013, Bruxelles, Belgium, November 12-14, 2013, Proceedings 3*, pp. 399–410. Springer, 2013.

A Proofs

A.1 Proof of Lemma 3.2

We first present exact forms for W, Γ, C, c_a.

$$W := \text{diag}(\sigma_z^2, 0, \cdots, 0) \in \mathbb{R}^{k\times k}$$

$$\Gamma = \begin{bmatrix} \gamma_1 & \gamma_2 & \cdots & \gamma_k \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix} \in \mathbb{R}^{k\times k}$$

$$C = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{1\times k}$$

$$c_a := \begin{bmatrix} \beta_a & 0 & \cdots & 0 \end{bmatrix}^\top \in \mathbb{R}^k$$

Proof. For the latent state evolution, it is fairly straightforward to compute that Equation 3 is equivalent to Equation 1 with the definitions of $\vec{z}_t, \Gamma$, and w_t. Notice that the second entry in $\vec{z}_t$ corresponds to the latent state value z_t. Similarly, for the reward function, we can directly compute that Equation 5 is equivalent to Equation 2 with the definition of $c_a^\top$.

All that is left is to show the equivalence of the measurement model y_t. Then by construction:

$$y_t = \frac{r_t - \mu_{a_t}}{\beta_{a_t}} = z_t + \frac{\epsilon_t(a_t)}{\beta_{a_t}} \sim \mathcal{N}(z_t, \sigma_r^2)$$

□

A.2 Proof of Lemma 3.3

Proof. Recall that due to Assumption C.1, the Kalman gain matrix K_t converges and the steady-state Kalman filter update and prediction can be combined into a single step (Gornet et al., 2022). Equation 19 in our setting becomes:

$$\tilde{z}_t = \Gamma \tilde{z}_{t-1} + \Gamma K(y_{t-1} - C\tilde{z}_{t-1}) = (\Gamma - \Gamma KC)\tilde{z}_{t-1} + \Gamma K y_{t-1} \tag{16}$$

Since $r_t(a) = \tilde{r}_t(a) + (r_t(a) - \tilde{r}_t(a)) = \tilde{r}_t(a) + \varepsilon_{a;t}$, it suffices to show that $\tilde{r}_t(a) = \Phi_t(s)^\top \theta_a + b_t(a, s)$. We first show that $\tilde{r}_t(a) = G_a^\top Y_t + \mu_a + \langle c_a, (\Gamma - \Gamma KC)^s \tilde{z}_{t-s} \rangle$ where

$$G_a := \begin{bmatrix} c_a^\top (\Gamma - \Gamma KC)^{s-1} \Gamma K & \cdots & c_a^\top \Gamma K \end{bmatrix}^\top \in \mathbb{R}^s, \quad Y_t := \begin{bmatrix} y_{t-s} & \cdots & y_{t-1} \end{bmatrix}^\top \in \mathbb{R}^s$$

Recall by definition:

$$\tilde{r}_t(a) = \mathbb{E}[r_t(a) | \mathcal{H}_{t-1}] = \mu_a + c_a^\top \tilde{z}_t$$

Using Equation 16, we can continuously unravel $\tilde{z}_t$ until the sth time step before:

$$\tilde{r}_t(a) = c_a^\top (\Gamma - \Gamma KC)\tilde{z}_{t-1} + c_a^\top \Gamma K y_{t-1} + \mu_a$$

$$= c_a^\top (\Gamma - \Gamma KC)^2 \tilde{z}_{t-2} + c_a^\top (\Gamma - \Gamma KC)\Gamma K y_{t-2} + c_a^\top \Gamma K y_{t-1} + \mu_a$$

$$= \cdots$$

$$= c_a^\top (\Gamma - \Gamma KC)^s \tilde{z}_{t-s} + c_a^\top (\Gamma - \Gamma KC)^{s-1} \Gamma K y_{t-s} + \cdots + c_a^\top (\Gamma - \Gamma KC)\Gamma K y_{t-2} + c_a^\top \Gamma K y_{t-1} + \mu_a$$

$$= G_a^\top Y_t + \mu_a + \langle c_a, (\Gamma - \Gamma KC)^s \tilde{z}_{t-s} \rangle$$

$$= G_a^\top Y_t + \mu_a + b_t(a, s)$$

Now let $g_a^j := c_a^\top (\Gamma - \Gamma KC)^j \Gamma K \in \mathbb{R}$. Then:

$$G_a^\top Y_t = y_{t-s} g_a^{s-1} + \cdots + y_{t-1} g_a^0$$

Using the definition of y_t,

$$G_a^\top Y_t = r_{t-s} \frac{g_a^{s-1}}{\beta_{a_{t-s}}} + \cdots + r_{t-1} \frac{g_a^0}{\beta_{a_{t-1}}} - \frac{\mu_{a_{t-s}} g_a^{s-1}}{\beta_{a_{t-s}}} - \cdots - \frac{\mu_{a_{t-1}} g_a^0}{\beta_{a_{t-1}}}$$

$$= R_t^\top \tilde{\beta}_a - A_t^\top \tilde{\mu}_a$$

where

$$R_t := \begin{bmatrix} r_{t-s} e_{a_{t-s}}^\top & \cdots & r_{t-1} e_{a_{t-1}}^\top \end{bmatrix}^\top \in \mathbb{R}^{s \cdot |\mathcal{A}|}$$

$$\tilde{\beta}_a = \begin{bmatrix} \frac{g_a^{s-1}}{\beta_1} & \dots & \frac{g_a^{s-1}}{\beta_{|\mathcal{A}|}} & \dots & \frac{g_a^0}{\beta_1} & \dots & \frac{g_a^0}{\beta_{|\mathcal{A}|}} \end{bmatrix}^\top \in \mathbb{R}^{s \cdot |\mathcal{A}|}$$

$$A_t := \begin{bmatrix} e_{a_{t-s}}^\top & \cdots & e_{a_{t-1}}^\top \end{bmatrix}^\top \in \mathbb{R}^{s \cdot |\mathcal{A}|},$$

$$\tilde{\mu}_a = \begin{bmatrix} \frac{\mu_1 g_a^{s-1}}{\beta_1} & \dots & \frac{\mu_{|\mathcal{A}|} g_a^{s-1}}{\beta_{|\mathcal{A}|}} & \dots & \frac{\mu_1 g_a^0}{\beta_1} & \dots & \frac{\mu_{|\mathcal{A}|} g_a^0}{\beta_{|\mathcal{A}|}} \end{bmatrix}^\top \in \mathbb{R}^{s \cdot |\mathcal{A}|}$$

One can verify that by these definitions, for $\theta_a = \begin{bmatrix} \tilde{\beta}_a & \tilde{\mu}_a & \mu_a \end{bmatrix}^\top$

$$\Phi_t(s)^\top \theta_a = \Phi(R_t, A_t)^\top \theta_a = R_t^\top \tilde{\beta}_a - A_t^\top \tilde{\mu}_a + \mu_a$$

Therefore we have shown:

$$r_t(a) = \Phi_t(s)^\top \theta_a + b_t(a, s) + \varepsilon_{a;t}$$

□

The proof for the confidence set lemma (Lemma 4.1) requires lemmas A.1 and A.3.

Lemma A.1. *[Bound on Context] Suppose Assumption 3.4 holds, then there exists some constant $L(s, \delta_r)$ such that $\|\Phi_t(s)\|^2 \leq L(s, \delta_r)$ for all t with probability at least $1 - \delta_r$, where $\delta_r \in (0, 1)$.*

Proof. Notice that by construction, $\|R_t\|^2 = R_t^\top R_t = \sum_{j=1}^s r_{t-j}^2$ and $\|A_t\|^2 = A_t^\top A_t = s$. So:

$$\|\Phi_t(s)\|^2 = \sum_{j=1}^s r_{t-j}^2 + s + 1$$

Since every r_{t-j} is Gaussian with variance $\beta_{a_{t-j}}^2 \sigma_r^2$, every r_{t-j} is also sub-Gaussian with parameter $\beta_{a_{t-j}} \sigma_r$. Therefore by Lemma B.5,

$$r_{t-j} < \mu_{a_{t-j}} + \beta_{a_{t-j}} \mathbb{E}[z_{t-j}] + \sqrt{2\beta_{a_{t-j}}^2 \sigma_r^2 \log(1/\delta_r)}) = R_{\max}(\delta_r)$$

with probability at least $1 - \delta_r$. Since Assumption 3.4 holds, we know that the mean of the AR process $\mathbb{E}[z_{t-j}]$ is always bounded.

Therefore,

$$\|\Phi_t(s)\|^2 < s(R_{\max}(\delta_r)^2 + 1) + 1 = L(s, \delta_r)$$

□

For proofs of Lemma A.3 and Theorem 4.2 we need the following lemma.

Lemma A.2. *Let all the information observed up to and including time $t - 1$ be encoded in the filtration $\mathcal{H}_{t-1} := \sigma(a_1, r_1, ..., a_{t-1}, r_{t-1})$ and $\vec{z}_t$ as defined in Equation 3 and $\tilde{z}_t = \mathbb{E}[\vec{z}_t | \mathcal{H}_{t-1}]$ be the steady-state Kalman filter for $\vec{z}_t$. Then $\vec{z}_t - \tilde{z}_t | \mathcal{H}_{t-1} \sim \mathcal{N}(\vec{0}, P)$ and $\vec{z}_t - \tilde{z}_t \sim \mathcal{N}(\vec{0}, P)$*

Proof. First notice that by construction of the steady-state Kalman filter, $\vec{z}_t|\mathcal{H}_{t-1} \sim \mathcal{N}(\tilde{z}_t, P)$ and $\tilde{z}_t|\mathcal{H}_{t-1}$ is a constant (i.e., not random). Therefore,

$$\vec{z}_t - \tilde{z}_t|\mathcal{H}_{t-1} = \vec{z}_t|\mathcal{H}_{t-1} - \tilde{z}_t|\mathcal{H}_{t-1} \sim \mathcal{N}(\vec{0}, P)$$

Since $\mathcal{N}(\vec{0}, P)$ is a fixed distribution and P does not depend on $\mathcal{H}_{t-1}$, this implies that $\vec{z}_t - \tilde{z}_t \sim \mathcal{N}(\vec{0}, P)$ as well. □

Lemma A.3. *[Noise Process Property] Let all the information observed up to and including time $t-1$ be encoded in the filtration $\mathcal{H}_{t-1} := \sigma(a_1, r_1, ..., a_{t-1}, r_{t-1})$. For any given a, the noise process $\{\varepsilon_{a;t}\}_t$ from Equation 6 is a martingale difference sequence given filtration $\mathcal{H}_{t-1}$ and is conditionally R-subgaussian for some constant $R \geq 0$,*

$$\forall t \geq 1, \mathbb{E}[\varepsilon_{a;t}|\mathcal{H}_{t-1}] = 0$$
$$\forall \alpha \in \mathbb{R}, \mathbb{E}[e^{\alpha\varepsilon_{a;t}}|\mathcal{H}_{t-1}] \leq exp(\alpha^2 R^2/2)$$

Proof. Fix some $a \in \mathcal{A}$. We first show that $\mathbb{E}[\varepsilon_{a;t}|\mathcal{H}_{t-1}] = 0$.

$$\mathbb{E}[\varepsilon_{a;t}|\mathcal{H}_{t-1}] = \mathbb{E}[\langle c_a, \vec{z}_t - \tilde{z}_t\rangle + \epsilon_t(a)|\mathcal{H}_{t-1}]$$

$$= c_a^\top(\mathbb{E}[\vec{z}_t - \tilde{z}_t|\mathcal{H}_{t-1}]) + \mathbb{E}[\epsilon_t(a)|\mathcal{H}_{t-1}]$$

$$= c_a^\top \mathbb{E}[\vec{z}_t - \tilde{z}_t|\mathcal{H}_{t-1}] \quad (\epsilon_t(a) \text{ is independent of } \mathcal{H}_{t-1} \text{ and } \mathbb{E}[\epsilon_t(a)] = 0)$$

$$= c_a^\top \vec{0} = 0 \quad (\text{Lemma A.2})$$

To prove that $\varepsilon_{a;t}$ is conditionally R-subgaussian for some R, we first show that $\varepsilon_{a;t}|\mathcal{H}_{t-1} \sim \mathcal{N}(0, c_a^\top P c_a + \sigma_r^2)$. Notice that $c_a^\top(\vec{z}_t - \tilde{z}_t)|\mathcal{H}_{t-1} \sim \mathcal{N}(0, c_a^\top P c_a)$ by Lemma A.2 and $\epsilon_t(a)|\mathcal{H}_{t-1} \sim \mathcal{N}(0, \sigma_r^2)$ since $\epsilon_t(a)$ is independent of $\mathcal{H}_{t-1}$. Therefore:

$$\varepsilon_{a;t}|\mathcal{H}_{t-1} = c_a^\top(\vec{z}_t - \tilde{z}_t)|\mathcal{H}_{t-1} + \epsilon_t(a)|\mathcal{H}_{t-1} \sim \mathcal{N}(0, c_a^\top P c_a + \beta_a^2\sigma_r^2)$$

The moment generating function (MGF) for the normal random variable $\varepsilon_{a;t}$ is:

$$M_{\varepsilon_{a;t}}(\alpha) = \mathbb{E}[e^{\alpha\varepsilon_{a;t}}] = \exp(\alpha^2(c_a^\top P c_a + \beta_a^2\sigma_r^2)/2) \;\; \forall \alpha \in \mathbb{R}$$

Then:

$$\mathbb{E}[e^{\alpha\varepsilon_{a;t}}|\mathcal{H}_{t-1}] = \mathbb{E}[e^{\alpha\varepsilon_{a;t}}] \quad (\text{as shown above, } \varepsilon_{a;t} \text{ is independent of } \mathcal{H}_{t-1})$$

$$= \exp(\alpha^2(c_a^\top P c_a + \beta_a^2\sigma_r^2)/2)$$

$$\leq \exp(\alpha^2 R^2/2) \;\; \forall \alpha \in \mathbb{R}$$

for some $R^2 \geq c_a^\top P c_a + \beta_a^2\sigma_r^2$ □

A.3 Proof of Lemma 4.1

Proof. First notice that for all actions a,

$$\hat{\theta}_{a,t} - \theta_a = V_{a,t}^{-1} \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s) r_j - \theta_a$$

$$= V_{a,t}^{-1} \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)(\Phi_j(s)^\top \theta_a + b_j(a,s) + \varepsilon_{a;t}) - \theta_a$$

$$= V_{a,t}^{-1} \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)(\Phi_j(s)^\top \theta_a + b_j(a,s) + \varepsilon_{a;t}) - V_{a,t}^{-1}(\lambda I + \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)\Phi_j(s)^\top)\theta_a$$

$$= V_{a,t}^{-1} \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s) b_j(a,s) + V_{a,t}^{-1} \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)\varepsilon_{a;j} - \lambda V_{a,t}^{-1}\theta_a$$

For the first term,

$$\left\| V_{a,t}^{-1} \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s) b_j(a,s) \right\|_{V_{a,t}} = \left\| \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s) b_j(a,s) \right\|_{V_{a,t}^{-1}}$$

$$\leq \sum_{j=1}^{t} \mathbb{I}[a_j = a] \|\Phi_j(s) b_j(a,s)\|_{V_{a,t}^{-1}} \quad \text{(generalized triangle-inequality)}$$

$$= \sum_{j=1}^{t} \mathbb{I}[a_j = a] \sqrt{b_j(a,s)^2 \Phi_j(s)^\top V_{a,t}^{-1} \Phi_j(s)}$$

$$\leq \sqrt{\sum_{j=1}^{t} \mathbb{I}[a_j = a] b_j(a,s)^2} \sqrt{\sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)^\top V_{a,t}^{-1} \Phi_j(s)} \quad \text{(Cauchy-Schwartz)}$$

$$= \tau(a,s)_t \sqrt{\sum_{j=1}^{t} \mathbb{I}[a_j = a] b_j(a,s)^2}$$

where $\tau(a,s)_t = \sqrt{\sum_{j=1}^{t} \mathbb{I}[a_j = a] \|\Phi_j(s)\|_{V_{a,t}^{-1}}^2}$

For the second term,

$$\left\| V_{a,t}^{-1} \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)\varepsilon_{a;j} \right\|_{V_{a,t}} = \left\| \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)\varepsilon_{a;j} \right\|_{V_{a,t}^{-1}}$$

By Lemma A.3, since the noise process $\varepsilon_{a;j}$ satisfies the assumptions of Theorem B.1,

$$\left\| \sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)\varepsilon_{a;j} \right\|_{V_{a,t}^{-1}} \leq \sqrt{2R^2 \log\left(\frac{\det(V_{a,t})^{1/2} \det(\lambda I)^{-1/2}}{\delta_\beta} \right)}$$

with probability at-least $1-\delta_\beta$. Using Lemma B.2 (determinant-trace inequality),

$$\det(V_{a,t}) \leq \left(\lambda + \frac{n_{a,t}L(s,\delta_r)}{2s|\mathcal{A}|+1}\right)^{2s|\mathcal{A}|+1}$$

where $n_{a,t} := \sum_{j=1}^{t} \mathbb{I}[a_j = a]$ and $\|\Phi_j(s)\|^2 \leq L(s,\delta_r)$ for all $j \in [t]$ because of Lemma A.1.

$$\implies \left\|\sum_{j=1}^{t} \mathbb{I}[a_j = a]\Phi_j(s)\varepsilon_{a;j}\right\|_{V_{a,t}^{-1}} \leq R\sqrt{(2s|\mathcal{A}|+1)\log\left(\frac{1+n_{a,t}L(s,\delta_r)/\lambda}{\delta_\beta}\right)}$$

For the third term,

$$\|\lambda V_{a,t}^{-1}\theta_a\|_{V_{a,t}} = \lambda\|\theta_a\|_{V_{a,t}^{-1}} \leq \sqrt{\lambda}\|\theta_a\| \leq \sqrt{\lambda}S_a$$

since $\|\theta_a\|^2_{V_{a,t}^{-1}} \leq \frac{1}{\lambda_{\min}(V_{a,t})}\|\theta_a\|^2 \leq \frac{1}{\lambda}\|\theta_a\|^2$ and using Assumption 3.5.

Using generalized triangle inequality

$$\implies \|\hat{\theta}_{a,t} - \theta_a\|_{V_{a,t}} \leq \left\|V_{a,t}^{-1}\sum_{j=1}^{t}\mathbb{I}[a_j = a]\Phi_j(s)b_j(a,s)\right\|_{V_{a,t}} + \left\|V_{a,t}^{-1}\sum_{j=1}^{t}\mathbb{I}[a_j = a]\Phi_j(s)\varepsilon_{a;j}\right\|_{V_{a,t}} + \left\|\lambda V_{a,t}^{-1}\theta_a\right\|_{V_{a,t}}$$

$$\leq R\sqrt{(2s|\mathcal{A}|+1)\log\left(\frac{1+n_{a,t}L(s,\delta_r)/\lambda}{\delta_\beta}\right)} + \sqrt{\lambda}S_a + \tau(a,s)_t\sqrt{\sum_{j=1}^{t}\mathbb{I}[a_j = a]b_j(a,s)^2}$$

Finally, for readability, we let $\delta_\beta = \delta_r = \delta/2$. Therefore with probability at least $1-\delta$,

$$\|\hat{\theta}_{a,t} - \theta_a\|_{V_{a,t}} \leq R\sqrt{(2s|\mathcal{A}|+1)\log\left(\frac{1+n_{a,t}L(s,\delta/2)/\lambda}{\delta/2}\right)} + \sqrt{\lambda}S_a + \tau(a,s)_t\sqrt{\sum_{j=1}^{t}\mathbb{I}[a_j = a]b_j(a,s)^2}$$

□

A.4 Proof of Theorem 4.2

Proof. Using Assumption 3.5 with Lemma 4.1, it suffices to prove the bound on the event that true parameter $\theta_a \in \mathcal{C}_{a,t}$ (Equation 13) for $\forall a \in \mathcal{A}$ and $t \in [T]$. Recall the regret in our setting (Equation 10) is:

$$\text{Regret}(T;\pi) = \sum_{t=1}^{T}\mathbb{E}[r_t(a_t^*) - r_t(a_t)|\vec{z}_t]$$

where a_t^* is the optimal action and a_t is the action selected by Algorithm 1 at time step t.

To assist with the proof, we consider an intermediate agent that knows the ground-truth parameters and therefore has the exact steady-state Kalman filter prediction $\tilde{z}_t$ and selects actions $\tilde{a}_t = \arg\max_a \ \tilde{r}_t(a)$.

Let $\Delta_t := \mathbb{E}[r_t(a_t^*) - r_t(a_t)|\vec{z}_t]$ denote the instantaneous regret at time t. Then:

$$\Delta_t = c_{a_t^*}^\top \vec{z}_t + \mu_{a_t^*} - c_{a_t}^\top \vec{z}_t - \mu_{a_t}$$

$$= (c_{a_t^*}^\top \vec{z}_t + \mu_{a_t^*} - c_{\tilde{a}_t}^\top \tilde{z}_t - \mu_{\tilde{a}_t}) + (c_{\tilde{a}_t}^\top \tilde{z}_t + \mu_{\tilde{a}_t} - c_{a_t}^\top \vec{z}_t - \mu_{a_t})$$

First notice that:

$$c_{a_t^*}^\top \vec{z}_t + \mu_{a_t^*} - c_{\tilde{a}_t}^\top \tilde{z}_t - \mu_{\tilde{a}_t} = c_{a_t^*}^\top \vec{z}_t - c_{a_t^*}^\top \tilde{z}_t + c_{a_t^*}^\top \tilde{z}_t + \mu_{a_t^*} - c_{\tilde{a}_t}^\top \tilde{z}_t - \mu_{\tilde{a}_t}$$

$$= c_{a_t^*}^\top (\vec{z}_t - \tilde{z}_t) + \tilde{r}_t(a_t^*) - \tilde{r}_t(\tilde{a}_t)$$

$$\leq c_{a_t^*}^\top (\vec{z}_t - \tilde{z}_t)$$

since $\tilde{r}_t(a_t^*) - \tilde{r}_t(\tilde{a}_t) \leq 0$ by the action-selection strategy of the intermediate agent.

Next notice that:

$$c_{\tilde{a}_t}^\top \tilde{z}_t + \mu_{\tilde{a}_t} - c_{a_t}^\top \vec{z}_t - \mu_{a_t} = c_{\tilde{a}_t}^\top \tilde{z}_t + \mu_{\tilde{a}_t} + (c_{a_t}^\top \tilde{z}_t - c_{a_t}^\top \tilde{z}_t) - c_{a_t}^\top \vec{z}_t - \mu_{a_t}$$

$$= \tilde{r}_t(\tilde{a}_t) - \tilde{r}_t(a_t) - c_{a_t}^\top (\vec{z}_t - \tilde{z}_t)$$

$$\implies \Delta_t \leq (c_{a_t^*} - c_{a_t})^\top (\vec{z}_t - \tilde{z}_t) + \tilde{r}_t(\tilde{a}_t) - \tilde{r}_t(a_t)$$

We first focus on $\tilde{r}_t(\tilde{a}_t) - \tilde{r}_t(a_t)$. Recall by Lemma 3.3 that $\tilde{r}_t(a) = \Phi_t(s)^\top \theta_a + b_t(a, s)$. Then:

$$\tilde{r}_t(\tilde{a}_t) - \tilde{r}_t(a_t) = \Phi_t(s)^\top \theta_{\tilde{a}_t} + b_t(\tilde{a}_t, s) - \Phi_t(s)^\top \theta_{a_t} - b_t(a_t, s)$$

Now let $\theta'_{a,t} = \max\{\mathcal{C}_{a,t-1}\}$ denote the max value of the confidence set $\mathcal{C}_{a,t-1}$ constructed at time t. Notice that:

$$\Phi_t(s)^\top \theta_{\tilde{a}_t} \leq \Phi_t(s)^\top \theta'_{\tilde{a}_t,t} \leq \Phi_t(s)^\top \theta'_{a_t,t}$$

where the first inequality is because $\theta_{\tilde{a}_t} \in \mathcal{C}_{\tilde{a}_t,t-1}$ and $\theta'_{\tilde{a}_t,t} = \max\{\mathcal{C}_{\tilde{a}_t,t-1}\}$, and the second inequality is by the action-selection strategy of Algorithm 1 (i.e., $a_t = \arg\max_a \ \Phi_t(s)^\top \theta'_{a,t}$). $\implies$

$$\tilde{r}_t(\tilde{a}_t) - \tilde{r}_t(a_t) \leq \Phi_t(s)^\top (\theta'_{a_t,t} - \theta_{a_t}) + b_t(\tilde{a}_t, s) - b_t(a_t, s)$$

$$\leq \|\Phi_t(s)\|_{V_{a,t-1}^{-1}} \|\theta'_{a_t,t} - \theta_{a_t}\|_{V_{a_t,t-1}} + b_t(\tilde{a}_t, s) - b_t(a_t, s)$$

(by Cauchy-Schwartz and $\|\cdot\|_{V_{a_t,t-1}^{-1}} \leq \|\cdot\|_{V_{a,t-1}}$)

$$\leq 2\|\Phi_t(s)\|_{V_{a_t,t-1}^{-1}} \beta_{a_t,t-1}(\delta') + b_t(\tilde{a}_t, s) - b_t(a_t, s)$$

$$\leq 2\|\Phi_t(s)\|_{V_{a_t,t-1}^{-1}} \beta_{a_t,t-1}(\delta') + 2\max_a |b_t(a, s)|$$

We now focus on $(c_{a_t^*} - c_{a_t})^\top (\vec{z}_t - \tilde{z}_t)$.

First,

$$(c_{a_t^*} - c_{a_t})^\top (\vec{z}_t - \tilde{z}_t) \leq \|c_{a_t^*} - c_{a_t}\| \|\vec{z}_t - \tilde{z}_t\| \quad \text{(Cauchy-Schwartz)}$$

$$\leq 2 \max_a \|c_a\| \|\vec{z}_t - \tilde{z}_t\|$$

By Lemma A.2, we know that $\vec{z}_t - \tilde{z}_t \sim \mathcal{N}_k(\vec{0}, P)$. By Lemma B.3, $\vec{z}_t - \tilde{z}_t$ is sub-Gaussian with parameter $\sigma^2 = \|P\|_{\text{op}}$. Finally by Theorem B.4, with probability at least $1 - \delta_z$ for $\delta_z \in (0, 1)$,

$$\|\vec{z}_t - \tilde{z}_t\| \leq 4\sqrt{\|P\|_{\text{op}} k} + 2\sqrt{\|P\|_{\text{op}} \log(1/\delta_z)}$$

$$\leq 4\sqrt{\|P\|_{\text{op}}}(\sqrt{k} + \sqrt{\log(1/\delta_z)})$$

$$\leq 4\sqrt{\|P\|_{\text{op}}}\sqrt{2(k + \log(1/\delta_z))} \quad (\sqrt{a} + \sqrt{b} \leq \sqrt{2(a+b)} \text{ for } a, b \in \mathbb{R}_{\geq 0})$$

We bound $\|P\|_{\text{op}}$. Recall that for the steady-state Kalman filter, $P = \Gamma P \Gamma^\top + W - \Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top$.

Using triangle inequality,

$$\|P\|_{\text{op}} = \|\Gamma P \Gamma^\top - \Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top\|_{\text{op}} + \|W\|_{\text{op}}$$

We now show that $\|\Gamma P \Gamma^\top - \Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top\|_{\text{op}} \leq \|\Gamma P \Gamma^\top\|_{\text{op}}$. To do so, we show that the following three matrices are positive semi-definite (PSD): (1) $A = \Gamma P \Gamma^\top$, (2) $B = \Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top$, and (3) $A - B$.

A is PSD because P is PSD and using Lemma B.6.

We now show B is PSD. Consider some vector $v \in \mathbb{R}^k$. Then $v^\top B v = v^\top (\Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top) v = (CP\Gamma^\top v)^\top (CPC^\top + V)^{-1} CP\Gamma^\top v = (CP\Gamma^\top v)^2 (CPC^\top + V)^{-1}$.

Now since $C = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{1 \times k}$, $CPC^\top = P_{11} \geq 0$ since P is PSD so diagonal entries are non-negative. Also $V = \frac{\sigma_r^2}{\min_a \beta_a^2} \implies (CPC^\top + V)^{-1} = \frac{1}{P_{11} + \frac{\sigma_r^2}{\min_a \beta_a^2}} \geq 0$.

We now show $A - B$ is PSD. Since P is PSD we know that there exists a PSD matrix $P^{1/2}$ such that $P = P^{1/2} P^{1/2}$. Therefore $A - B = \Gamma P \Gamma^\top - \Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top = \Gamma P^{1/2} (I - P^{1/2} C^\top (CP^{1/2} P^{1/2} C^\top + V)^{-1} CP^{1/2}) P^{1/2} \Gamma^\top = \Gamma P^{1/2} (I - \nu(\nu^\top \nu + V)^{-1} \nu^\top) P^{1/2} \Gamma^\top$ for $\nu = P^{1/2} C^\top \in \mathbb{R}^k$.

Notice that $\nu(\nu^\top \nu + V)^{-1} \nu^\top = \frac{\nu\nu^\top}{\nu^\top \nu + V} = \frac{\frac{\nu}{\|\nu\|} \frac{\nu^\top}{\|\nu\|}}{\frac{v^\top v}{\|\nu\|^2} + \frac{V}{\|\nu\|^2}} = \frac{\nu' \nu'^\top}{1 + \frac{V}{\|\nu\|^2}}$ where $\nu' = \nu / \|\nu\|$. Since ν' is a unit vector, $\nu' \nu'^\top$ has only one non-zero eigenvalue which is 1, which implies $\frac{\nu' \nu'^\top}{1 + \frac{V}{\|\nu\|^2}}$ has one non-zero eigenvalue which is $\frac{1}{1 + \frac{V}{\|\nu\|^2}}$. Furthermore this implies that $I - \frac{\nu' \nu'^\top}{1 + \frac{V}{\|\nu\|^2}}$ has eigenvalues either 1 or $1 - \frac{1}{1 + \frac{V}{\|\nu\|^2}} \geq 0$. Therefore, $I - \nu(\nu^\top \nu + V)^{-1} \nu^\top$ is PSD and so is $\Gamma P^{1/2} (I - \nu(\nu^\top \nu + V)^{-1} \nu^\top) P^{1/2} \Gamma^\top$ by Lemma B.6.

Since we have shown A, B, and $A - B$ are PSD, then by Lemma B.7, $\|\Gamma P \Gamma^\top - \Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top\|_{\text{op}} \leq \|\Gamma P \Gamma^\top\|_{\text{op}}$.

$\implies$

$$\|P\|_{\text{op}} \leq \|\Gamma P \Gamma^\top\|_{\text{op}} + \|W\|_{\text{op}}$$

Now,

$$\|\Gamma P \Gamma^\top\|_{\text{op}} \leq \|\Gamma\|_{\text{op}} \|P\|_{\text{op}} \|\Gamma^\top\|_{\text{op}} = \sigma_{\max}(\Gamma)^2 \|P\|_{\text{op}}$$

since A and $A^\top$ have the same singular values for any matrix A.

Also in our setting, W is a matrix of all 0s except for σ_z^2 in the first diagonal entry, so $\|W\|_{\text{op}} = \sigma_z^2$. So,

$$\|P\|_{\text{op}} \leq \sigma_{\max}(\Gamma)^2 \|P\|_{\text{op}} + \sigma_z^2 \implies \|P\|_{\text{op}} \leq \frac{\sigma_z^2}{1 - \sigma_{\max}(\Gamma)^2}$$

Therefore, at every t, the instantaneous regret is bounded as so:

$$\Delta_t \leq 8 \max_a \|c_a\| \sqrt{\frac{\sigma_z^2}{1 - \sigma_{\max}(\Gamma)^2}} \sqrt{2(k + \log(1/\delta_z))} + 2\|\Phi_t(s)\|_{V^{-1}_{a_t,t-1}} \beta_{a_t,t-1}(\delta') + 2 \max_a |b_t(a,s)|$$

So,

$$\text{Regret}(T; \pi_{\text{LARL}}) = \sum_{t=1}^{T} \Delta_t$$

$$\leq 8 \max_a \|c_a\| \sqrt{\frac{\sigma_z^2}{1 - \sigma_{\max}(\Gamma)^2}} \sqrt{2(k + \log(1/\delta_z))} \cdot T + 2 \sum_{t=1}^{T} \|\Phi_t(s)\|_{V^{-1}_{a_t,t-1}} \beta_{a_t,t-1}(\delta') + 2 \sum_{t=1}^{T} \max_a |b_t(a,s)|$$

Now,

$$2 \sum_{t=1}^{T} \|\Phi_t(s)\|_{V^{-1}_{a_t,t-1}} \beta_{a_t,t-1}(\delta') \leq 2\beta_T(\delta') \sqrt{\sum_{t=1}^{T} \|\Phi_t(s)\|^2_{V^{-1}_{a_t,t-1}}} \sqrt{T}$$

where $\beta_T(\delta') = \max_a \beta_{a,T-1}(\delta')$ and $\sum_{t=1}^{T} \|\Phi_t(s)\|_{V^{-1}_{a_t,t-1}} \leq \sqrt{T \cdot \sum_{t=1}^{T} \|\Phi_t(s)\|^2_{V^{-1}_{a_t,t-1}}}$ by variant of Cauchy-Schwartz.

For readability, let $\delta_z = \delta_b = \delta_r = \delta/3$. Then w.p. at-least $1 - \delta$,

$$\text{Regret}(T; \pi_{\text{LARL}}) \leq 8 \max_a \|c_a\| \sqrt{\frac{\sigma_z^2}{1 - \sigma_{\max}(\Gamma)^2}} \sqrt{2(k + \log(3/\delta))} \cdot T + 2\beta_T(2\delta/3) \sqrt{\sum_{t=1}^{T} \|\Phi_t(s)\|^2_{V^{-1}_{a_t,t-1}}} \sqrt{T} + 2 \sum_{t=1}^{T} \max_a |b_t(a,s)|$$

□

B Auxillary Theorems and Lemmas

Theorem B.1. *(Theorem 1 in Abbasi-Yadkori et al. (2011)) Let $\{\mathcal{F}_t\}_{t=0}^{\infty}$ be a filtration. Let $\{\eta_t\}_{t=1}^{\infty}$ be a real-valued stochastic process such that η_t is $\mathcal{F}_t$-measurable and η_t is conditionally R-sub-Gaussian for some $R \geq 0$. Namely:*

$$\forall\, \lambda \in \mathbb{R} \quad \mathbb{E}[e^{\lambda \eta_t} | \mathcal{F}_{t-1}] \leq \exp\left(\frac{\lambda^2 R^2}{2}\right)$$

Let $\{x_t\}_{t=1}^{\infty}$ be an $\mathbb{R}^d$-valued stochastic process such that x_t is $\mathcal{F}_{t-1}$-measurable. Assume that $V_0 \in \mathbb{R}^{d\times d}$ is a positive definite matrix. For any $t \geq 1$, define:

$$V_t = V_0 + \sum_{j=1}^{t} x_j x_j^\top \quad S_t = \sum_{j=1}^{t} \eta_j x_j$$

Then for any $\delta > 0$, with probability atleast $1-\delta$, for all $t \geq 1$,

$$\|S_t\|^2_{V_t^{-1}} \leq 2R^2 \log\left(\frac{det(V_t)^{1/2} det(V_0)^{-1/2}}{\delta}\right)$$

Lemma B.2. *(Lemma 10 Determinant-Trace Inequality in Abbasi-Yadkori et al. (2011)) Suppose $x_1, ..., x_t \in \mathbb{R}^d$ and $\|x_j\| \leq L \; \forall \; j \in [t]$. Let $V_t = \lambda I + \sum_{j=1}^{t} x_j x_j^\top$ for some $\lambda > 0$. Then:*

$$det(V_t) \leq \left(\lambda + \frac{tL^2}{d}\right)^d$$

Lemma B.3. *(Lemma 8.2 in Rinaldo & Wu (2019)) Let $X \in \mathbb{R}^d$ be a random vector that is normally distributed $X \sim \mathcal{N}(0, \Sigma)$. Then X is a sub-Gaussian random vector with parameter $\|\Sigma\|_{op}$.*

Theorem B.4. *(Theorem 8.3 in Rinaldo & Wu (2019)) $X \in \mathbb{R}^d$ be a sub-Gaussian random vector with parameter σ^2, then with probability at least $1-\delta$ for $\delta \in (0,1)$:*

$$\|X\| \leq 4\sigma\sqrt{d} + 2\sigma\sqrt{\log(1/\delta)}$$

Lemma B.5. *(Sub-Gaussian upper tail bound) Let X be sub-Gaussian with variance proxy σ^2. Then for any $\delta \in [0,1]$ we have:*

$$\mathbb{P}(X - \mathbb{E}[X] < \sqrt{2\sigma^2 \log(1/\delta)}) \leq 1-\delta$$

Lemma B.6. *If $B \in \mathbb{R}^{m\times m}$ is positive semi-definite, then for any matrix $A \in \mathbb{R}^{n\times m}$, $ABA^\top$ is also positive semi-definite.*

Lemma B.7. *If matrices A, B, and $A-B$ are positive semi-definite, then $\|A-B\|_{op} \leq \|A\|_{op}$.*

C A Review of Linear Dynamical Systems with Gaussian Noise

We provide a brief review of discrete-time linear dynamical systems (LDS) with Gaussian noise (Roweis & Ghahramani, 1999).

C.1 Setting

A discrete-time, autonomous, LDS with Gaussian noise can be described with the following two equations:

$$\text{(State Evolution)} \quad \vec{z}_t = \Gamma \vec{z}_{t-1} + w_t, \tag{17}$$

$$\text{(Measurement Model)} \quad y_t = C\vec{z}_t + v_t, \tag{18}$$

where $\vec{z}_t \in \mathbb{R}^k$ is the (latent) state of the system with noise process $w_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu_z, W)$, $y_t \in \mathbb{R}$ is some measurement that is observable with noise process $v_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu_y, V)$, and Γ, C are constant matrices.

C.2 Steady-State Kalman Filter

With knowledge of $\Gamma, C, \mu_z, W, \mu_y, V$, Kalman filtering is the standard approach for predicting $\vec{z}_t$ using previous measurements $y_1, ..., y_{t-1}$, even if $\vec{z}_t$ is not observed. Namely, the optimal prediction (in the least mean square sense) for $\vec{z}_t$ would be $z_{t|t-1}$, where $z_{t|j} := \mathbb{E}[\vec{z}_t|\mathcal{F}_j]$ and $\mathcal{F}_j$ is the sigma algebra generated by previous measurements $y_1, ..., y_j$.

A standard assumption is that the LDS given by equations (17) and (18) is observable:

Assumption C.1. The observability matrix, $\mathcal{O} = \begin{bmatrix} C \\ C\Gamma \\ C\Gamma^2 \\ \vdots \\ C\Gamma^{k-1} \end{bmatrix} \in \mathbb{R}^{k \times k}$, is full rank.

Assumption C.1 leads to a steady-state solution (Ljung, 1999; Gornet et al., 2022) that is often employed in practice for applications involving numerous time steps (Kailath et al., 2000). For simplicity, let $\tilde{z}_t := z_{t|t-1}$. The steady-state Kalman filter is as follows, where the prediction and measurement update steps are combined:

$$\tilde{z}_t = \Gamma \tilde{z}_{t-1} + \mu_z + \Gamma K(y_{t-1} - C\tilde{z}_{t-1} - \mu_y) \tag{19}$$

$$K = PC^\top (CPC^\top + V)^{-1} \tag{20}$$

$$P = \Gamma P \Gamma^\top + W - \Gamma P C^\top (CPC^\top + V)^{-1} CP\Gamma^\top \tag{21}$$

The prediction of $\tilde{z}_t$ is recursively computed given the prediction from the previous time-step $\tilde{z}_{t-1}$ and the most recent measurement y_{t-1}. K is the steady-state Kalman gain matrix which acts as a weighting factor balancing the model's predictions with the discrepancy between the model's most recent prediction and measurement. The update for K is one that yields the minimum mean-square error estimate in the limit. P is the steady-state version of $P_t := \text{cov}(\vec{z}_t - z_{t|t})$, the error covariance for $\tilde{z}_t$. Assumption C.1 ensures that in the limit, P_t converges to some P, which implies the Kalman gain matrix converges to some K (Ljung, 1999; Gornet et al., 2022).

D Discussion on Regret

In standard stationary bandit settings, one often proves regret bounds with respect to a "standard oracle" that knows the true fixed reward means $\mu(a)$ for each action and selects the optimal action $a^* = \arg\max_{a \in \mathcal{A}} \mu(a)$. In non-stationary settings, where the mean rewards change over time, many works compare to an equivalent oracle called the dynamic oracle (Besbes et al., 2014). The dynamic oracle is one that knows the true reward means $\mu_t(a)$ at every time step t for each action and selects optimal action at every time step t, $a_t^* = \arg\max_{a \in \mathcal{A}} \mu_t(a)$.

Equivalently, the dynamic oracle is an oracle that observes all information in the environment and then acts optimally with that information. In the latent AR bandit setting, such an oracle knows the ground-truth parameters $\theta^* = [\gamma_0, \gamma_1, ..., \gamma_k, \mu_1, \beta_1, ..., \mu_{|\mathcal{A}|}, \beta_{|\mathcal{A}|}, \sigma_z, \sigma_r]$ and observes the latent process (i.e., the realization of z_t). With access to the ground-truth parameters and the realization of z_t, the oracle therefore knows the true reward means $\mu_t(a)$ at every time step t for each action. The oracle selects the optimal action for every t: $a_t^* = \arg\max_{a \in \mathcal{A}} \mu_t(a)$. For a policy π, the regret is defined as:

$$\text{Regret}(T; \pi) = \sum_{t=1}^{T} \mathbb{E}[r_t(a_t^*) - r_t(a_t)|\vec{z}_t]$$

where a_t is the action selected by the algorithm at time step t following policy π. The first term is the mean reward obtained by the oracle (i.e., reward obtained by selecting the most optimal action for that time-step) and the second term denotes the mean reward obtained by the agent (where the agent only has information from the history up to but not including time step t). With no other assumptions, it is impossible to achieve sub-linear regret with respect to this oracle in the latent AR setting.

D.1 Sub-linear Dynamic Regret

Sub-linear regret with respect to the dynamic oracle is only possible in environments with vanishing non-stationarity (i.e., there is a budget for the non-stationarity that is sub-linear in T). For example in Besbes et al. (2014), the non-stationarity is formulated by arbitrary changes to the mean rewards and they assume a finite variation budget V_T of how much the mean rewards can change over time. The regret bound for their method Rexp3 is on the order of $V_T^{1/3}T^{2/3}$. If V_T scales linearly with T, the regret of their method would be linear in T. Similarly in Garivier & Moulines (2011), they assume a finite number of changes to the mean reward or breakpoints Υ_T. The regret bound for their methods discounted UCB and sliding window UCB is on the order of $\sqrt{T\Upsilon_T}\log T$, where $\Upsilon_T = \mathcal{O}(T^\beta)$ for some $\beta \in [0, 1)$. If Υ_T scales linearly with T, then the regret for their approaches would also achieve linear regret. In our setting, σ_z^2, the noise variance on the latent state process, is the mechanism that controls the non-stationarity. We have shown in our main regret bound result (Theorem 4.2) that for fixed T, in environments where $\sigma_{\max}(\Gamma) \leq 1 - \epsilon$ for $\epsilon > 0$ and $\sigma_z^2 = T^{c-2}$ for some constant $c < 2$, our algorithm achieves sub-linear regret.

E Additional Experiments

E.1 Verifying the Bias Variance Trade-off

We verify the trade-off with the choice of s as depicted in Theorem 4.2. Recall that hyperparameter $s > 0$ dictates the number of recent time steps of history to incorporate into the context. We consider three environment variants where $k = 1, 5, 10$. In each environment variant, we run our algorithm with four different values of s, where $s = 1, 5, 10, 15$, compared to "Stationary", standard UCB which treats the environment as a stationary multi-armed bandit.

Figure 4 shows pairwise comparisons between Stationary and our algorithm LARL with various choices of s. We look at the proportion of times over 100 Monte-Carlo repetitions where the algorithm listed in the row achieved lower total cumulative regret than the algorithm listed in the column. Figure 5 shows cumulative regret over time. Even when s is not specifically tuned, our algorithm still outperforms Stationary; however the choice of s does dictate how much our algorithm excels. These simulations verify the bias-variance trade off with the choice of s as shown in the regret bound (Theorem 4.2). If s is chosen too large, the bias term is small but our algorithm's reward model is burdened with learning many parameters. If s is chosen too small, the bias term is large and our algorithm does not include enough history to inform the current prediction.

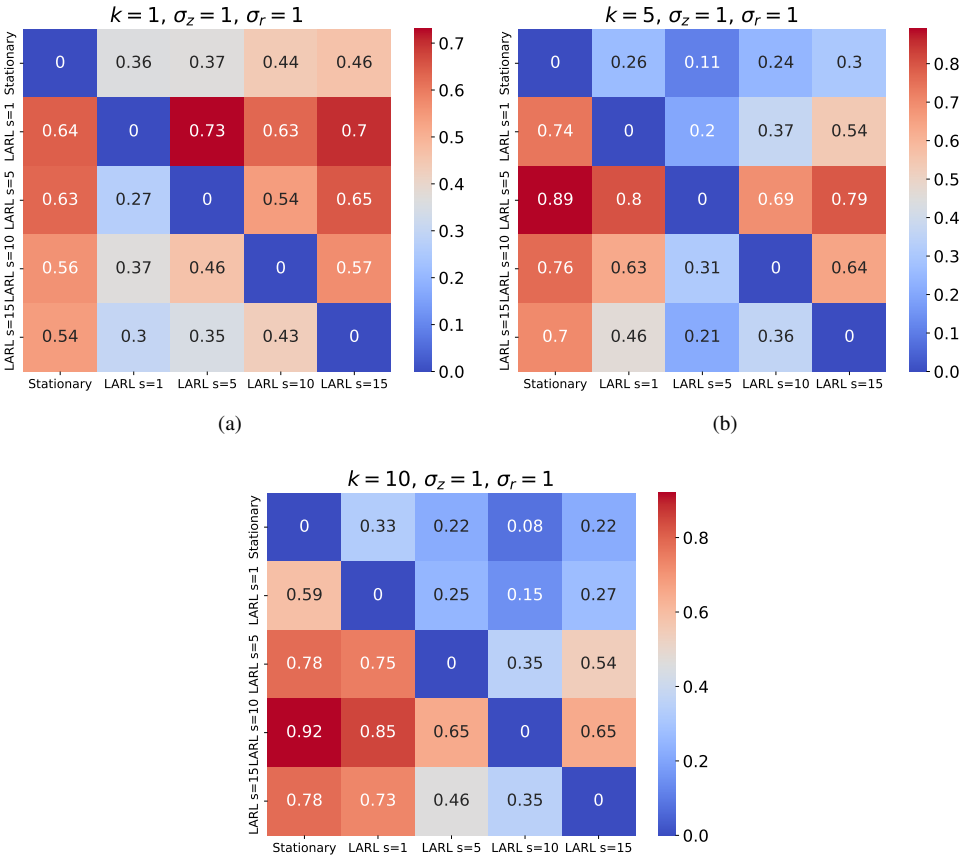


Figure 4: Pairwise comparisons between algorithms in the environment variants where $k = 1, 5, 10$, respectively. Each cell shows the proportion of 100 Monte-Carlo repetitions where the algorithm listed in the row achieved lower cumulative regret than the algorithm listed in the column. Even when s is not specifically tuned, our algorithm still outperforms Stationary.

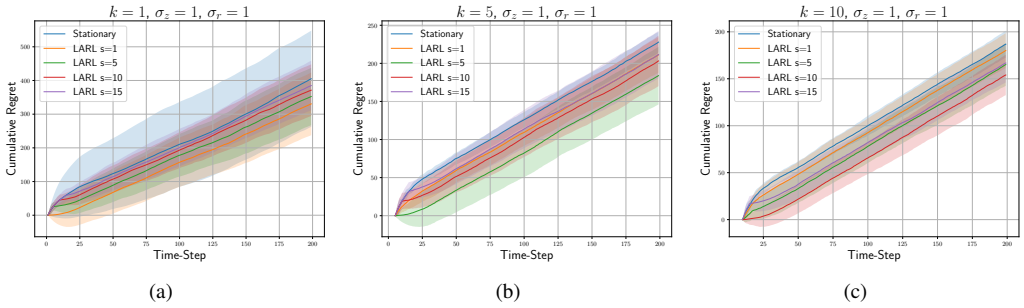


Figure 5: Cumulative regret (Equation 10) over time with varying choices of s for our algorithm Latent AR LinUCB (Algorithm 1). For a poor choice of s (either too small or too large compared to k) however, our algorithm performs similarly to the stationary. If s is too small, the reward model is under-parameterized. If s is too large, the reward model is over-parameterized. Line is the average and shaded region is $\pm$ standard deviation across Monte Carlo simulated trials.