Regret Analysis of Multi-task Representation Learning for Linear-Quadratic Adaptive Control

Bruce D. Lee[*1], **Leonardo F. Toso**[*2], **Thomas T. Zhang**[*1], **James Anderson**[2], **Nikolai Matni**[1]

[1]Department of Electrical and Systems Engineering, University of Pennsylvania
[2]Department of Electrical Engineering, Columbia University
{brucele, ttz2, nmatni}@seas.upenn.edu, {lt2879, james.anderson}@columbia.edu

Abstract

Representation learning is a powerful tool that enables learning over large multitudes of agents or domains by enforcing that all agents operate on a shared set of learned features. However, many robotics or controls applications that would benefit from collaboration operate in settings with changing environments and goals, whereas most guarantees for representation learning are stated for static settings. Toward rigorously establishing the benefit of representation learning in dynamic settings, we analyze the regret of multi-task representation learning for linear-quadratic control. This setting introduces unique challenges. Firstly, we must account for and balance the misspecification introduced by an approximate representation. Secondly, we cannot rely on the parameter update schemes of single-task online LQR, for which least-squares often suffices, and must devise a novel scheme to ensure sufficient improvement. We demonstrate that for settings where exploration is "benign", the regret of any agent after T timesteps scales with the square root of T/H, where H is the number of agents. In settings with "difficult" exploration, the regret scales as the square root of the input dimension times the parameter dimension multiplied by T, plus a term which scales with T to the three quarters divided by H to the one fifth. In both cases, by comparing to the minimax single-task regret, we see a benefit of a large number of agents. Notably, in the difficult exploration case, by sharing a representation across tasks, the effective task-specific parameter count can often be small. Lastly, we validate the trends we predict.

1 Introduction

Many modern applications of robotics and controls involve simultaneous control over a large number of agents. For example, robot fleet learning, in which fleets of robots performing diverse tasks share information to learn more effectively, has demonstrated impressive success in recent years (Brohan et al. 2022; Wang et al. 2023b). One of the technologies that enables this success is *transfer learning*, in which dynamics models or control policies built upon learned compressed features (also known as *representation learning*) that are broadly useful for ensuing tasks of interest. Existing work which characterizes the generalization capabilities of transfer learning largely considers static environments, where data from an agent's completed task is aggregated with data from other agents to learn the shared features offline, rather than during task execution. However, it is often relevant to have a fleet of agents adapt quickly to a changing environment, e.g. a team of drones flying in close proximity adapting to weather conditions, or a team of legged robots adapting to changing terrain conditions. In such settings, the agents must communicate to adjust shared features online.

In this work, we rigorously study such approaches for online fleet learning with dynamical systems in the analytically tractable setting of adaptive linear-quadratic (state-feedback) control. Adaptive linear-quadratic control has emerged as a benchmark for learning to control dynamical systems using online data. This consists of a learner interacting with an unknown linear system

$$x_{t+1} = A_\star x_t + B_\star u_t + w_t, \quad t \geq 1, \tag{1}$$

with state x_t, input u_t, and noise w_t assuming values in $\mathbb{R}^{d_{\mathsf{X}}}$, $\mathbb{R}^{d_{\mathsf{U}}}$, and $\mathbb{R}^{d_{\mathsf{X}}}$, respectively. The learner is evaluated by its incurred *regret*, which compares the cost incurred by playing the learner for T time steps against the cost attained by the optimal LQR controller. Prior work typically studies regret of a single dynamical system of the form (1). In this work, we study a setting where there are $H \gg 1$ distinct systems which share an unknown d_θ-dimensional dynamics basis. Each agent aims to minimize their individual linear-quadratic control objective; however, by communicating they may more efficiently learn the shared dynamics basis matrices. The questions we address are the following:

- What are the requisite algorithmic elements that enable simultaneous online control of *multiple* systems?
- What are the benefits of sharing a representation across agents compared with learning individual models?

1.1 Related Work

Fleet Learning: Fleet Policy Learning involves obtaining datasets from diverse robot interactions, studied in offline reinforcement learning (Kumar et al. 2022) and multi-task behavior cloning (Brohan et al. 2022, 2023; Goyal et al. 2023). Challenges include data communication, storage, and training scalability. Weight merging frameworks have been proposed (Wang et al. 2023b), focusing on skill aggregation and adaptive communication for changing environments, similar

[*]These authors contributed equally.

to federated learning settings (Collins et al. 2021; Ma et al. 2022; Tan et al. 2022).

Multi-Task Learning (in Dynamical Systems): Multi-task learning has been extensively studied in machine learning (Baxter 2000). Recent works highlight the benefits of shared representations in iid learning for generalization (Du et al. 2020; Tripuraneni, Jordan, and Jin 2020) and efficient algorithms (Collins et al. 2021; Vaswani 2024; Thekumparampil et al. 2021; Tripuraneni, Jin, and Jordan 2021). However, these works often do not address the complexities of data from dynamical systems. In dynamical systems, multi-task settings often involve agents sharing a parameter space with task-specialization arising from perturbations, as seen in Model-agnostic meta-learning (MAML) (Finn, Abbeel, and Levine 2017).[1] Both model-free federated learning of the linear-quadratic regulator with heterogeneous data (Wang et al. 2023a) and MAML for linear-quadratic control (Toso et al. 2024) have been explored, yet they only achieve optimality up to a heterogeneity bias. Imposing a shared basis for all dynamics matrices (Modi et al. 2021) ensures error decreases as data increases. Analogous settings have been studied in imitation learning (Zhang et al. 2023a; Guo et al. 2023). Most relevant to our work is Zhang et al. (2024), which addresses iid representation learning shortcomings for linear system-identification, a component of which is adapted in our proposed algorithm.

Regret Analysis of Adaptive Control: Our setting and analysis builds off recent work that attempts to provide finite sample guarantees for adaptive control by controlling the *regret* of the learning algorithm. While adaptive control has a rich history beginning with autopilot development for high-performance aircraft in the 1950s (Gregory 1959), finite sample regret analysis of adaptive control arose much later (Abbasi-Yadkori and Szepesvári 2011). Subsequent work (Dean et al. 2018; Cohen, Koren, and Mansour 2019; Mania, Tu, and Recht 2019) has introduced algorithms that yield $\sqrt{T}$ regret, and are computationally feasible. Simchowitz and Foster (2020) establish corresponding lower bounds, indicating that a rate of $\sqrt{d_{\mathsf{U}}^2 d_{\mathsf{X}} T}$ is optimal for completely unknown systems. Improved regret bounds of $\texttt{poly}(\log T)$ are achievable when either $A^\star$ or $B^\star$ is known (Cassel, Cohen, and Koren 2020; Jedra and Proutiere 2022). The aforementioned work studies adaptive control in a setting where the noise is zero-mean and stochastic. Alternative formulations of the adaptive LQR problem consider bounded adversarial disturbances (Hazan, Kakade, and Singh 2020; Simchowitz, Singh, and Hazan 2020) and settings where there is misspecification between the underlying data generating process and the model class (Ghai et al. 2022; Lee, Rantzer, and Matni 2024). Our work extends analogous regret analysis to the multi-agent setting.

1.2 Contribution

We propose and analyze fleet linear-quadratic adaptive control in a setting where multiple linear systems driven by dynamics in the span of d_θ common basis matrices can communicate to drastically improve their individual control objectives. We propose such an algorithm and analyze the regret incurred, uncovering an interesting transition distinguishing the difficulty of the problem:

- When the system specific parameters are "benign" to identify, our proposed scheme incurs a regret of

$$R_T = \tilde{\mathcal{O}}\Big(\sqrt{T/H}\Big),$$

  where H is the number of communicating agents. When there are many agents, this is drastically lower than the regret $\mathcal{O}(\sqrt{d_{\mathsf{X}} d_{\mathsf{U}}^2 T})$ incurred if each agent had to learn to control its respective system without communication.
- When the system-specific parameters are challenging to identify, our proposed algorithm incurs a regret of at most

$$R_T = \tilde{\mathcal{O}}\left(\sqrt{d_{\mathsf{U}} d_\theta}\sqrt{T} + \frac{T^{3/4}}{H^{1/5}}\right).$$

  When T is moderate, or if the number of agents H is large, this can demonstrate a marked gain over the single-agent setting. However, when T is large, the $T^{3/4}$ term dominates, which arises due to the mismatch between the difficulty of parameter identification and the misspecification of the learned basis directions.

In order to establish such guarantees, we propose and analyze a new algorithm that synthesizes tools from regret analysis of misspecified linear system identification and algorithmic analysis of multi-task linear regression. In particular, the multi-agent setting introduces unique challenges:

- Due to the approximate representation at any given timestep, the problem is misspecified. Therefore, in addition to the standard explore-commit tradeoff, we must account for improving the representation.
- Whereas for prior work in the stochastic single-agent setting least-squares–whose optimization and generalization is well-understood–suffices algorithmically, such an analog is not well-posed for the multiple agent setting.

Numerical experiments validate our theory and demonstrate the value of shared representations in learning to control.

Notation: The spectral norm is denoted $\|A\|$, and the Frobenius norm is denoted $\|A\|_F$. We denote the condition number as $\kappa(A) \triangleq \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}$. For $f, g : D \to \mathbb{R}$, we write $f \lesssim g$ if for some $c > 0$, $f(x) \le c g(x) \, \forall x \in D$. We denote the solutions to the discrete Lyapunov and Riccati equations by $\texttt{dlyap}(A, Q)$ and $\texttt{DARE}(A, B, Q, R)$, respectively.

2 Problem Formulation

2.1 System and Data Assumptions

Consider H systems with dynamics defined by

$$x_{t+1}^{(h)} = A_\star^{(h)} x_t^{(h)} + B_\star^{(h)} u_t^{(h)} + w_t^{(h)}, \quad t \ge 1, \qquad (2)$$

for $h \in [H]$. We suppose that each rollout starts from initial state $x_1^{(h)} = 0$ for $h \in [H]$, and that that the noise $w_t^{(h)}$ has iid elements that are mean zero and σ^2-sub-Gaussian for some $\sigma^2 \in \mathbb{R}$ with $\sigma^2 \ge 1$ (Vershynin 2018). We

[1] This differs from our setting, where agents share a representation function with task-specialization derived from linear functions of the representation.

additionally assume that the noise has identity covariance: $\mathbf{E}\left[w_t^{(h)} w_t^{(h),\top}\right] = I$.[2] We suppose the dynamics matrices admit the decomposition

$$\begin{bmatrix} A_\star^{(k)} & B_\star^{(k)} \end{bmatrix} = \mathsf{vec}^{-1}\left(\Phi_\star \theta_\star^{(k)}\right), \tag{3}$$

where $\Phi_\star \in \mathbb{R}^{d_{\mathsf{X}}(d_{\mathsf{X}}+d_{\mathsf{U}}) \times d_\theta}$ is a column-orthonormal matrix that contains an optimal set of d_θ (vectorized) basis matrices in $\mathbb{R}^{d_{\mathsf{X}}(d_{\mathsf{X}}+d_{\mathsf{U}})}$, and $\theta_\star^{(k)} \in \mathbb{R}^{d_\theta}$ are agent-specific parameters. The operator vec^{-1} maps a vector in $\mathbb{R}^{d_{\mathsf{X}}(d_{\mathsf{X}}+d_{\mathsf{U}})}$ into a matrix in $\mathbb{R}^{d_{\mathsf{X}} \times (d_{\mathsf{X}}+d_{\mathsf{U}})}$ by stacking contiguous length-d_{X} blocks of a vector (top-to-bottom) into columns of a matrix (left-to-right). We can equivalently write this as a linear combination of basis matrices:

$$\begin{bmatrix} A_\star^{(k)} & B_\star^{(k)} \end{bmatrix} = \sum_{i=1}^{d_\theta} \theta_{\star,i}^{(k)} \begin{bmatrix} \Phi_{\star,i}^A & \Phi_{\star,i}^B \end{bmatrix},$$

where $\begin{bmatrix} \Phi_{\star,i}^A & \Phi_{\star,i}^B \end{bmatrix} = \mathsf{vec}^{-1} \Phi_{\star,i}$ and $\Phi_{\star,i}$ is the i^{th} column of $\Phi_\star$. This decomposition of the data generating process is a natural extension of the low-rank linear representations considered in (Du et al. 2020; Zhang et al. 2023a, 2024) to the setting of multiple related dynamical systems with shared structure determined by $\Phi_\star$. A version of this model for autonomous systems was considered by (Modi et al. 2021, Assumption 3) for multi-task system identification.

2.2 Control Objective

The goal of the learners is to interact with system (2) while keeping the total cumulative cost small, where the system specific cumulative cost for system h is defined for matrices $Q \succeq I$ and $R = I$ as[3]

$$C_T^{(h)} \triangleq \sum_{t=1}^{T} c_t^{(h)}, \text{ and } c_t^{(h)} \triangleq x_t^{(h),\top} Q x_t^{(h)} + u_t^{(h),\top} R u_t^{(h)}.$$

To define an algorithm that keeps the cost small, we first introduce the infinite horizon LQR cost:

$$\mathcal{J}^{(h)}(K) \triangleq \limsup_{T \to \infty} \frac{1}{T} \mathbf{E}^K C_T^{(h)}, \tag{4}$$

where the superscript K denotes evaluation under the state-feedback controller $u_t^{(h)} = K x_t^{(h)}$. To ensure that there exists a controller such that (4) is finite, we assume $(A_\star^{(h)}, B_\star^{(h)})$ is stabilizable for all $h \in [H]$. Under this assumption, (4) is minimized by the LQR controller $K_\infty(A_\star^{(h)}, B_\star^{(h)})$, where

$$K_\infty(A, B) \triangleq -(B^\top P_\infty(A, B) B + R)^{-1} B^\top P_\infty(A, B) A,$$
$$P_\infty(A, B) \triangleq \mathtt{DARE}(A, B, Q, R).$$

We define the shorthands $P_\star^{(h)} \triangleq P_\infty(A_\star^{(h)}, B_\star^{(h)})$ and $K_\star^{(h)} \triangleq K_\infty(A_\star^{(h)}, B_\star^{(h)})$ for all $h \in [H]$. To characterize the infinite-horizon LQR cost of an arbitrary stabilizing controller K, we additionally define the solution $P_K^{(h)}$ to the Lyapunov equation for the closed loop system under an arbitrary K where $\rho(A_\star^{(h)} + B_\star^{(h)} K) < 1$:

$$P_K^{(h)} \triangleq \mathtt{dlyap}(A_\star^{(h)} + B_\star^{(h)} K, Q + K^\top R K).$$

For a controller K satisfying $\rho(A_\star^{(h)} + B_\star^{(h)} K) < 1$, $\mathcal{J}^{(h)}(K) = \mathrm{tr}(P_K^{(h)})$. We have that $P_{K_\star^{(h)}}^{(h)} = P_\star^{(h)}$.

The infinite horizon LQR controller provides a baseline level of performance that our learner cannot surpass in the limit as $T \to \infty$. We quantify the performance of our learning algorithm by comparing the cumulative cost $C_T^{(h)}$ to the scaled infinite horizon cost attained by the LQR controller if the system matrices $\begin{bmatrix} A_\star^{(h)} & B_\star^{(h)} \end{bmatrix}$ were known:

$$\mathcal{R}_T^{(h)} \triangleq C_T^{(h)} - T \mathcal{J}^{(h)}(K_\star^{(h)}). \tag{5}$$

This metric has previously been considered for adaptive control of a single system (Abbasi-Yadkori and Szepesvári 2011). The above formulation casts the goal of the learner as interacting with each system (2) to maximize the information required for control while simultaneously regulating each system to minimize $\mathcal{R}_T^{(h)}$. The learner uses its history of interaction with each system to do so by constructing dynamics models, e.g. by determining estimates $\hat{A}^{(h)}$ and $\hat{B}^{(h)}$. It may then use these estimates as part of a *certainty equivalent* (CE) design by synthesizing controllers $\hat{K}^{(h)} = K_\infty(\hat{A}^{(h)}, \hat{B}^{(h)})$. It is known from prior work that if the model estimate is sufficiently close to the true dynamics, then the excess cost of playing the controller $\hat{K}^{(h)}$ is bounded by its parameter estimation error (Mania, Tu, and Recht 2019; Simchowitz and Foster 2020).

Lemma 2.1 (Theorem 3 of (Simchowitz and Foster 2020)). *Define* $\varepsilon^{(h)} \triangleq \frac{\|P_\star^{(h)}\|^{-10}}{3000}$. *If* $\left\| \begin{bmatrix} \hat{A}^{(h)} & \hat{B}^{(h)} \end{bmatrix} - \begin{bmatrix} A_\star^{(h)} & B_\star^{(h)} \end{bmatrix} \right\|_F^2 \le \varepsilon^{(h)}$, *then*

$$\mathcal{J}^{(h)}(\hat{K}^{(h)}) - \mathcal{J}^{(h)}(K_\star^{(h)}) \le$$
$$142 \left\| P_\star^{(h)} \right\|^8 \left\| \begin{bmatrix} \hat{A}^{(h)} & \hat{B}^{(h)} \end{bmatrix} - \begin{bmatrix} A_\star^{(h)} & B_\star^{(h)} \end{bmatrix} \right\|_F^2.$$

3 Algorithm Description

Our proposed algorithm, Algorithm 1, is a CE algorithm similar to those proposed by Cassel, Cohen, and Koren (2020); Lee, Rantzer, and Matni (2024), which we extend to the multi-task representation learning setting. The algorithm takes a stabilizing controller $K_0^{(h)}$ for each system h as an input, in addition to an initial epoch length τ_1, an exploration sequence σ_k^2 for $k \in [k_{\mathsf{fin}}]$, state and controller bounds x_b and K_b, an initial representation estimate Φ_0, and a number of gradient steps N to run on the representation per epoch. Starting from the initial controllers, Algorithm 1 follows a

[2] Noise that enters the process through a non-singular matrix S can be addressed by rescaling the dynamics by S^{-1}.

[3] Generalizing to arbitrary $Q \succ 0$ and $R \succ 0$ can be performed by scaling the cost and changing the input basis. One can also generalize by allowing Q and R to differ among systems, but we use a single value for expositional convenience.

Algorithm 1 Shared-Representation Certainty-Equivalent Control with Continual Exploration

Input: Stabilizing controllers $K_0^{(h)}$ for $h \in [H]$, initial epoch length τ_1, number of epochs k_{fin}, exploration sequence $\sigma_1^2, \sigma_2^2, \sigma_3^2, \dots \sigma_{k_{\mathsf{fin}}}^2$, state bound x_b, controller bound K_b, initial representation estimate Φ_0, gradient steps per epoch N

Init: $\hat{K}_1^{(h)} \leftarrow K_0^{(h)}, \tau_0 \leftarrow 0, T \leftarrow \tau_1 2^{k_{\mathsf{fin}}-1}, \hat{\Phi}_1 \leftarrow \Phi_0$.

for $k = 1, 2, \dots, k_{\mathsf{fin}}$ **do**

  for $h = 1, \dots, H$ **(in parallel) do**

    for $t = \tau_{k-1}, \tau_{k-1}+1, \dots, \tau_k$ **do**

      if $\|x_t^{(h)}\|^2 \geq x_b^2 \log T$ or $\|\hat{K}_k^{(h)}\| \geq K_b$ **then**

        Abort and play $K_0^{(h)}$ forever

      Play $u_t^{(h)} = \hat{K}_k^{(h)} x_t^{(h)} + \sigma_k g_t^{(h)}$,

      where $g_t^{(h)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$

    $\hat{\theta}_k^{(h)} \leftarrow \texttt{LS}(\hat{\Phi}_k, x^{(h)}_{\tau_{k-1}:\lceil \frac{3}{2}\tau_{k-1}\rceil}, u^{(h)}_{\tau_{k-1}:\lceil \frac{3}{2}\tau_{k-1}\rceil})$

    $\begin{bmatrix} \hat{A}_k^{(h)} & \hat{B}_k^{(h)} \end{bmatrix} \leftarrow \mathsf{vec}^{-1}\left(\hat{\Phi}_k \hat{\theta}_k^{(h)}\right)$

    $\hat{K}_{k+1}^{(h)} \leftarrow K_\infty(\hat{A}_k^{(h)}, \hat{B}_k^{(h)})$

  $\hat{\Phi}_{k+1} \leftarrow \texttt{DFW}(\hat{\Phi}_k, x^{(1:H)}_{\lceil \frac{3}{2}\tau_{k-1}\rceil:\tau_k}, u^{(1:H)}_{\lceil \frac{3}{2}\tau_{k-1}\rceil:\tau_k}, N)$

  $\tau_{k+1} \leftarrow 2\tau_k$

Algorithm 2 Least squares: $\texttt{LS}(\hat{\Phi}, x_{1:t+1}, u_{1:t})$

Input: Basis $\hat{\Phi}$, state data $x_{1:t+1}$, input data $u_{1:t}$

Return: $\hat{\theta}$, where

$$\hat{\theta} = \Lambda^\dagger \left(\sum_{s=1}^{t} \hat{\Phi}^\top \left(\begin{bmatrix} x_s \\ u_s \end{bmatrix} \otimes I_{d_{\mathsf{x}}} \right) x_{s+1} \right) \quad \text{and}$$

$$\Lambda = \sum_{s=1}^{t} \hat{\Phi}^\top \left(\begin{bmatrix} x_s \\ u_s \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix}^\top \otimes I_{d_{\mathsf{x}}} \right) \hat{\Phi}.$$

doubling epoch approach. During each epoch, each agent plays their current controller with exploratory noise added with scale determined by the exploration sequence. Each agent then uses the collected data to estimate its dynamics $\begin{bmatrix} \hat{A}^{(h)} & \hat{B}^{(h)} \end{bmatrix}$ by running least-squares (Algorithm 2), fixing the current representation estimate $\hat{\Phi}$.[4] This is used to synthesize a new CE controller $\hat{K}^{(h)} = K_\infty(\hat{A}^{(h)}, \hat{B}^{(h)})$. At the end of each epoch, the agents engage in a round of N representation updates (Algorithm 3), in which they update their estimate for the shared basis using local data and communicate to take the average of their estimates. To analyze expected regret it is necessary to prevent catastrophic failures even under unlikely failure events. For this reason, the algorithm checks the state and controller norm against the supplied bounds x_b and K_b at the start of each interaction round, and aborts the CE scheme if either is too large.

[4]This does not allow updating the model at arbitrary times and throws away earlier data. This eases the analysis, but may be undesirable. Such undesirable characteristics have been removed in single task expected regret analysis (Jedra and Proutiere 2022).

Algorithm 3 De-bias & Feature Whiten: $\texttt{DFW}(\hat{\Phi}, x_{1:t}^{(1:H)}, u_{1:t}^{(1:H)}, N)$

Input: Representation estimate $\hat{\Phi}$, state data $x_{1:t+1}^{(1:H)}$, input data $u_{1:t}^{(1:H)}$, gradient steps N, step-size η

Split trajectories into segments of length t_1 and t_2, $N(t_1 + t_2) \leq t$.

for $n = 1, \dots, N$ **do**

  for $h = 1, \dots, H$ **in parallel do**

    Est. weights: $\hat{\theta}_n^{(h)} \leftarrow \texttt{LS}(\hat{\Phi}_n, \{x_s^{(h)}, u_s^{(h)}\}_{s \in [t_1]})$.

    Compute local rep. update $\overline{\Phi}_n^{(h)}$ (6) on $s \in [t_2]$.

  Avg. global rep $\hat{\Phi}^n, \_ \leftarrow \texttt{thin\_QR}(\frac{1}{H}\sum_{h=1}^{H} \bar{\Phi}_n^{(h)})$.

Return: $\hat{\Phi}_+ \leftarrow \hat{\Phi}_N$

A key subtlety and contribution of our algorithm is how the parameters are updated (Algorithm 1 and 3). In the single-agent setting, the optimal dynamics matrix $\begin{bmatrix} \hat{A} & \hat{B} \end{bmatrix}$ follows from least squares, halving the parameter error with each doubling epoch (Simchowitz and Foster 2020). However, in our multi-agent setting, least squares is neither implementable nor optimal, motivating the need for an alternative subroutine to reduce representation error between epochs. Existing linear representation learning algorithms often assume iid isotropic Gaussian data $x_i^{(h)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$ (Collins et al. 2021; Thekumparampil et al. 2021; Tripuraneni, Jin, and Jordan 2021), which is violated in our setting where states converge to their respective stationary distributions. A recent algorithm, `De-bias & Feature Whiten (DFW)`, proposed by Zhang et al. (2023b), addresses many similar issues in a related multi-task representation learning problem, which we adapt for our setting. Beyond its guarantees (see Section 3.1), `DFW` enables distributed optimization of a shared linear representation across data sources with non-identical distributions and temporally dependent covariates. Additionally, `DFW` does not require communication of raw data between agents; instead, each agent communicates its updated representation, enabling federated implementation. During each `DFW` iteration $n \in [N]$, each agent estimates its local parameters via least-squares using the current representation $\hat{\Phi}_{n-1}$ (see Algorithm 2). Then, each agent uses another portion of its data to compute its local representation descent step:

$$\begin{aligned} \nabla_{\Phi,n}^{(h)} &\triangleq \nabla_\Phi \sum_{t \in D_n} \left\| x_{t+1}^{(h)} - \mathsf{vec}^{-1}\left(\hat{\Phi}\hat{\theta}_n^{(h)}\right) \begin{bmatrix} x_t^{(h)} \\ u_t^{(h)} \end{bmatrix} \right\|^2 \\ \hat{\Sigma}_n^{(h)} &\triangleq \sum_{t \in D_n} \left(\begin{bmatrix} x_t^{(h)} \\ u_t^{(h)} \end{bmatrix} \begin{bmatrix} x_t^{(h)} \\ u_t^{(h)} \end{bmatrix}^\top \otimes I_{d_{\mathsf{x}}} \right) \\ \overline{\Phi}_n^{(h)} &\leftarrow \hat{\Phi}_{n-1} - \eta \, (\hat{\Sigma}_n^{(h)})^{-1} \nabla_{\Phi,n}^{(h)}. \end{aligned} \tag{6}$$

The updated local representations from each agent are then averaged and orthonormalized, and transmitted back to each agent for the next iteration (see Algorithm 3, line 7).

3.1 Representation Error Guarantees

In this section, we motivate the roles of our representation update (Algorithm 3) and task-specific weight update (Algorithm 2) subroutines. Consider current representation estimate $\hat{\Phi}$ and data $(x_{1:t}^{(1:H)}, u_{1:t}^{(1:H)})$ generated from initial states $x_1^{(1)}, \ldots, x_1^{(H)}$, under stabilizing controllers $K^{(1)}, \ldots, K^{(H)}$ with exploratory noise $\sigma_u g_s^{(h)}$, $g_s^{(h)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_{d_{\mathsf{U}}})$ for $s \in [t]$, $h \in [H]$, and some $\sigma_u \in [0, 1]$. This can be seen as the general set-up for the data collected during an epoch of Algorithm 1. We want to establish the following:

1. Running `DFW` yields an updated representation whose error decomposes as a contraction of the previous representation's error plus a variance term that scales inversely with the amount of *total data* tH.
2. The error $\left\|\hat{\Phi}\hat{\theta}^{(h)} - \Phi_\star \theta_\star^{(h)}\right\|$ accrued by fitting the least-squares task-specific weights, holding the representation fixed, decomposes into a sum of least-squares error scaling inversely with t and the *representation error*.

These two guarantees together inform how to set the epoch length and exploratory noise strength σ_u to balance the explore-commit tradeoff for the ensuing regret analysis. To quantify the representation error, we consider the *subspace distance* between the spaces spanned by the columns of $\hat{\Phi}$ and $\Phi_\star$ (which are constrained to be column-orthonormal).

Definition 3.1 (Stewart and Sun (1990))**.** *For a given matrix with orthonormal columns Φ, let $\Phi_\perp$ be a matrix such that $[\Phi \quad \Phi_\perp]$ is an orthogonal matrix. Then, given another column-orthonormal matrix Φ', the* subspace distance *between Φ', Φ may be written $d(\Phi, \Phi') \triangleq \|\Phi_\perp^\top \Phi'\|$.*

For all dimensions of $\Phi_\star$ to be identifiable, assume the optimal weights $\theta_\star^{(1)}, \ldots, \theta_\star^{(H)}$ are full rank.

Assumption 3.1. *Consider $\Phi_\star$, $\{\theta_\star^{(h)}\}$ such that $\mathsf{vec}^{-1}(\Phi_\star \theta_\star^{(h)}) = \begin{bmatrix} A_\star^{(h)} & B_\star^{(h)} \end{bmatrix}$, $h = 1, \ldots, H$. We assume $\operatorname{rank}\left(\sum_{h=1}^H \theta_\star^{(h)} \theta_\star^{(h)\top}\right) = d_\theta$.*

We now state a bound on the improvement of the subspace distance after running Algorithm 3.

Theorem 3.1 (`DFW` guarantee, redux)**.** *Let Assumption 3.1 hold and fix $\delta \in (0, 1)$. Then, provided an appropriately chosen step-size $\eta > 0$, burn-in $t \geq \tau_{\mathsf{dfw}}$, and initial representation error $d(\hat{\Phi}, \Phi_\star) \leq d_{\mathsf{dfw}}$, with probability at least $1 - \delta$ running Algorithm 3 yields the following guarantee on the updated representation $\hat{\Phi} \to \hat{\Phi}_N$:*

$$d(\hat{\Phi}_N, \Phi_\star) \leq \rho^N d(\hat{\Phi}, \Phi^\star) + \frac{\overline{K}_{\mathsf{avg}}}{1 - \sqrt{2}\rho^N} \frac{\sqrt{N}}{\sigma_u \sqrt{tH}},$$

where

$$\rho = 1 - 0.897\eta \lambda_{\min}\left(\sum_{h=1}^H \theta_\star^{(h)} \theta_\star^{(h)\top}\right)$$

$$\overline{K}_{\mathsf{avg}} = \sqrt{\frac{1}{H}\sum_{h=1}^H \sigma^2 \|\theta_\star^{(h)}\|^2 (2 + \|K^{(h)}\|^2)} \cdot \operatorname{poly}(d_{\mathsf{X}}, d_{\mathsf{U}}, \log(H), \log(1/\delta)).$$

In particular, we have demonstrated that running `DFW` contracts the subspace distance by a factor of ρ^N, up to a variance factor. Notably, $\overline{K}_{\mathsf{avg}}$ serves as a task-averaged "noise-level", and the denominator of the variance factor scales *jointly* with the number of tasks H and data per task t. For downstream analysis, it suffices to choose a number of iterations N such that $\rho^N \leq 1/2$, i.e., $N \geq \log(2)/\log(1/\rho)$, which is independent of the size of the data. The subspace distance manifests in the error between the learned system parameters $\hat{\Phi}\hat{\theta}$ and the optimal $\Phi_\star \theta_\star$. In particular, given the output $\hat{\theta}$ of Algorithm 2, it can be shown (e.g. Theorem 5, (Lee, Rantzer, and Matni 2024)) that the parameter least squares error decomposes into a term scaling inversely with data and a term involving the subspace distance between $\hat{\Phi}$ and $\Phi_\star$.

Theorem 3.2 (Theorem 5 of Lee, Rantzer, and Matni (2024), informal)**.** *Consider running Algorithm 2 on the t data samples generated from a system of the form* (2) *for $t \geq \tau_{\mathsf{ls}}$, where τ_{ls} is a burn-in time. Then with probability at least $1 - \delta$,*

$$\left\|\hat{\Phi}\hat{\theta} - \Phi_\star \theta_\star\right\|^2 \lesssim \frac{\sigma^2 d_\theta \log(1/\delta)}{t \times \mathsf{excitation\ lvl}} + C_{\mathsf{sys}} \frac{d(\hat{\Phi}, \Phi_\star)^2}{\mathsf{excitation\ lvl}},$$

where C_{sys} is a constant that depends on the system (2)*, and* excitation lvl *characterizes the extent to which the the state is excited as required to identify the parameters θ.*

Formal statements of Theorem 3.1 and Theorem 3.2 are instantiated in the ensuing regret analysis and can be found in the appendix. We have thus established the desiderata stated at the beginning of the section. It remains to show that salient choices of epoch length and exploratory noise level in Algorithm 1 yield no-regret guarantees.

4 Regret Analysis

As previewed in the introduction, we consider two settings: one where the system-specific parameters $\theta_\star^{(h)}$ are easily identifiable given the representation, and one in which they are not. The setting where the system-specific parameters are easily identifiable corresponds to a situation in which excitation lvl from Theorem 3.2 is nonzero even when the input is determined by the optimal LQR controller. In both settings, we require that the bounds for the abort procedure (Line 6, Algorithm 1) are sufficiently large to ensure that the abort procedure occurs with small probability. To state the bounds, we introduce the following notation.

$$\Psi_{B_\star^{(h)}} \triangleq \max\left\{1, \left\|B_\star^{(h)}\right\|\right\}, \qquad \Psi_B^\vee \triangleq \max_{h=1,\ldots,H} \Psi_{B_\star^{(h)}}$$

$$\theta^\vee \triangleq \max_{h=1,\ldots,H} \left\|\theta_\star^{(h)}\right\|, \qquad P_0^\vee \triangleq \max_{h=1,\ldots,H} \left\|P_{K_0^{(h)}}^{(h)}\right\|$$

$$P_\star^\wedge \triangleq \min_{h=1,\ldots,H} \left\|P_{K_\star^{(h)}}^{(h)}\right\|, \qquad \varepsilon^\wedge \triangleq \min_{h=1,\ldots,H} \varepsilon^{(h)},$$

where $\varepsilon^{(h)}$ is as in Lemma 2.1.We introduce an assumption to make the analysis compact by enabling clean integration of the tail probabilities for computing expected regret.

Assumption 4.1. *We assume that*

$$x_b \geq 400 (P_0^\vee)^2 \Psi_B^\vee \sigma \sqrt{d_{\mathsf{X}} + d_{\mathsf{U}}}, \quad K_b \geq \sqrt{P_0^\vee}.$$

4.1 Not Easily Identifiable

In this setting, we do not make additional assumptions about the structure of $\Phi_\star$. We require an assumption ensuring that it is possible to obtain a stabilizing CE controller after the first epoch with high probability. To do so, we make an assumption about the size of the subspace distance of the representation estimate $\hat{\Phi}$ from $\Phi_\star$ after a single episode (leveraging the contraction of Theorem 3.1.) [5]

Assumption 4.2. *Define*

$$\beta_1 \triangleq C_{\beta,1}\sigma^4 (P_0^\vee)^{12}(\Psi_B^\vee)^8(\theta^\vee)^2(d_{\mathsf{X}}+d_{\mathsf{U}})\sqrt{\frac{d_\theta}{d_{\mathsf{U}}}},$$

$$\gamma_1 \triangleq \frac{1}{C_{\gamma,1}} \frac{\sigma_1^2}{x_b^2 (P_0^\vee)^5 \Psi_{B^\star}^2 \sqrt{\kappa\left(\sum_{h=1}^{d_\theta} \theta_\star^{(h)} \theta_\star^{(h),\top}\right)}}$$

for sufficiently large universal constants $C_{\beta,1}$ and $C_{\gamma,1}$. Let ρ be as in Theorem 3.1. We assume the initial subspace distance satisfies $d(\Phi_0, \Phi_\star) \le \min\left\{\frac{\varepsilon^\wedge}{4H^{2/5}\beta_1}, \gamma_1\right\}$.

This assumption leads to the following regret bound.

Theorem 4.1. *Consider applying Algorithm 1 with initial stabilizing controllers $K_0^{(1)}, \dots K_0^{(H)}$ for $T = \tau_1 2^{k_{\mathsf{fin}}-1}$ timesteps for some positive integers k_{fin}, and τ_1. Let $\tau_k = 2^k \tau_1$ for $k \in [k_{\mathsf{fin}}]$. Suppose that the exploration sequence supplied to the algorithm satisfies*

$$\sigma_k^2 = \max\left\{\tau_k^{-1/4} H^{-1/5},\ \sqrt{\frac{d_\theta}{d_{\mathsf{U}}\tau_k}},\ \rho^{(k-1)N} d(\Phi_0, \Phi_\star)\right\} \tag{7}$$

for $k \in [k_{\mathsf{fin}}]$, where ρ is the contraction rate of Theorem 3.1. Suppose the state bound x_b and the controller bound K_b satisfy Assumption 4.1 and that $N \ge \log(2)/\log(1/\rho)$. Additionally suppose that the weights satisfy Assumption 3.1. There exists a polynomial function $\mathtt{poly}_{\mathsf{warm}}$ such that if $\tau_1 = \tau_{\mathsf{warm}} \log^9 T$ with

$$\tau_{\mathsf{warm}} \ge \mathtt{poly}_{\mathsf{warm}}(\sigma, P_0^\vee, \Psi_B^\vee, \theta^\vee, x_b, d_\theta, d_{\mathsf{X}}, d_{\mathsf{U}}, \log(H)),$$

then the expected regret satisfies for $h = 1, \dots, H$

$$\boldsymbol{E}\left[\mathcal{R}_T^{(h)}\right] \le c_0 \log^9(T) + c_1 \sqrt{d_\theta d_{\mathsf{U}}}\sqrt{T}\log^2(T) + c_2 \frac{T^{3/4}}{H^{1/5}} \log^2(HT),$$

where $c_0 = \mathtt{poly}\Big(\sigma, d_{\mathsf{X}}, d_{\mathsf{U}}, d_\theta, x_b, K_b, \|Q\|, \theta^\vee, P_0^\vee, \Psi_B^\vee, \tau_{\mathsf{warm}}, x_b, d(\hat{\Phi}_0, \Phi_\star), \log H\Big)$, $c_1 = \mathtt{poly}(P_0^\vee, \Psi_B^\vee, \sigma)$, and $c_2 = \mathtt{poly}\Big(d_{\mathsf{X}}, d_{\mathsf{U}}, d_\theta, P_0^\vee, \Psi_B^\vee, \theta^\vee, \sigma, N\Big)$.

[5] The assumption is necessary for analysis; but it is likely only technical, as the algorithm experimentally converges from random initializations. The constants do not appear in the bounds.

Note that c_0 and c_2 depend on various system and algorithm quantities, however c_1 depends only upon quantities which nominally do not depend on system dimension. This is to emphasize the dimension dependence of the $\sqrt{T}$ term in the regret bound. Consider the above bound in the regime where T is small, e.g., on the order of the number of communicating agents. In this regime, the $T^{3/4}$ term becomes negligible, and the regret is dominated by the term that scales as $\sqrt{d_\theta d_{\mathsf{U}}}\sqrt{T}$. This should be contrasted with the minimax regret bound for single task adaptive control $\sqrt{d_{\mathsf{X}} d_{\mathsf{U}}^2 T}$ (Simchowitz and Foster 2020): if the system-specific parameter count d_θ is smaller than $d_{\mathsf{X}} d_{\mathsf{U}}$, then the dominant term in the low data regime is smaller than the minimax regret of the single-task setting. In the adaptive control setting under consideration, the low data regime is often the one of interest, as we want the controller to rapidly adapt to a changing environment. We note that the guarantees are not any time, as they require algorithm parameters to be chosen as a function of the time horizon T (as required by the choice of τ_1 and the assumption that N satisfies Assumption 4.2.) The algorithm depends on unknown quantities through σ_k^2, including ρ and $d(\Phi_0, \Phi^\star)$. These may instead be replaced with upper bounds, at the cost of larger c_0, c_1, c_2.

4.2 Easily Identifiable

In this setting, we assume that $\Phi^\star$ admits additional structure that makes the identification of $\theta_\star^{(h)}$ easy.

Assumption 4.3. *Let $\alpha \ge \frac{1}{3\|P_\star^\wedge\|^{3/2}}$. We assume that $\lambda_{\min}\left(\Phi_\star^\top \left(\begin{bmatrix} I \\ K \end{bmatrix}\begin{bmatrix} I \\ K \end{bmatrix}^\top \otimes I_{d_{\mathsf{X}}}\right)\Phi_\star\right) \ge \alpha^2$ for $K = K_0^{(h)}, K_\star^{(h)}$ for $h \in [H]$.*

The assumption is a persistence of excitation condition that captures a setting where playing either the initial controller K_0 or the optimal controller $K^\star$ provides persistence of excitation without any exploratory input if the representation were known. This can be seen by noting that the matrix $\Phi_\star^\top \left(\begin{bmatrix} I \\ K \end{bmatrix}\begin{bmatrix} I \\ K \end{bmatrix}^\top \otimes I_{d_{\mathsf{X}}}\right)\Phi_\star$ is a lower bound (in Loewner order) for the covariance matrix formed by taking the expectation of Λ/t in Algorithm 2 when $u_s = Kx_s$ and $\hat{\Phi} = \Phi_\star$.

Under the above assumption, the weights θ are easily identifiable once the shared structure Φ is learned. As in the previous section, we require that the initial representation error is small enough to guarantee the closeness condition in Lemma 2.1 may be satisfied after a single epoch.[5]

Assumption 4.4. *Define*

$$\beta_2 \triangleq C_{\beta,2} \max_{h=1,\dots,H} \frac{\varepsilon^\wedge (P_0^\vee)^9 (\Psi_B^\vee)^8 (\theta^\vee)^2 (d_{\mathsf{X}}+d_{\mathsf{U}})}{d_\theta \min\{\alpha^2, \alpha^4\}}$$

$$\gamma_2 \triangleq \frac{1}{C_{\gamma,2}} \frac{\alpha^2}{x_b^2 (P_0^\vee)^5 \Psi_{B^\star}^2 \sqrt{\kappa\left(\sum_{h=1}^{d_\theta} \theta_\star^{(h)} \theta_\star^{(h),\top}\right)}}$$

for sufficiently large universal constants $C_{\beta,2}$ and $C_{\gamma,2}$. We assume that $d(\Phi_0, \Phi_\star) \le \min\left\{\frac{\varepsilon^\wedge}{2\beta_1}, \gamma_2\right\}$.

This allows us to state the following regret bound.

Theorem 4.2. *Consider applying Algorithm 1 with initial stabilizing controller $K_0^{(1)}, \dots, K_0^{(H)}$ for $T = \tau_1 2^{k_{\mathsf{fin}}}$ timesteps for some positive integers k_{fin}, and τ_1. Let $\tau_k = 2^k \tau_1$ for $k \in [k_{\mathsf{fin}}]$ and suppose the exploration sequence is*

$$\sigma_k^2 = \max\left\{\tau_k^{-1/2} H^{-1/2}, \rho^{(k-1)N} d(\Phi_0, \Phi_\star)\right\}, \quad (8)$$

for all $k \in [k_{\mathsf{fin}}]$, where ρ is the contraction rate of Theorem 3.1. Suppose the state bound x_b and the controller bound K_b satisfy Assumption 4.1, $\Phi_\star$ satisfies Assumption 4.3, and Φ_0 satsisfies Assumption 4.4. Additionally suppose that the parameter N is sufficiently large that $\rho^N \le \frac{1}{2}$ and that the weights satisfy Assumption 3.1. There exists a polynomial $\mathtt{poly}_{\mathsf{warm}}$ such that if $\tau_1 = \tau_{\mathsf{warm}} \log^4 T$ with

$$\tau_{\mathsf{warm}} \ge \mathtt{poly}_{\mathsf{warm}}\left(\sigma, P_0^\vee, \Psi_B^\vee, \theta^\vee, x_b, d_\theta, d_{\mathsf{X}}, d_{\mathsf{U}}, \log(H), \frac{1}{\alpha}\right),$$

then the expected regret satisfies for $h = 1, \dots, H$ satisfies

$$\boldsymbol{E}\left[\mathcal{R}_T^{(h)}\right] \le c_1 \log^4(T) + c_2 \frac{\sqrt{T}}{\sqrt{H}} \log^2(TH),$$

where $c_1 = \mathtt{poly}\Big(\sigma, d_\theta, d_{\mathsf{U}}, d_{\mathsf{X}}, \frac{1}{\alpha}, \Psi_B^\vee, P_0^\vee, x_b, K_b, \theta^\vee, \|Q\|, \tau_{\mathsf{warm}}, d(\hat{\Phi}_0, \Phi_\star), \log H\Big)$, *and* $c_2 = \mathtt{poly}\Big(\sigma, d_\theta, d_{\mathsf{U}}, d_{\mathsf{X}}, \frac{1}{\alpha}, \Psi_B^\vee, P_0^\vee, x_b, N\Big)$.

Consider once more the setting when the amount of data is on the order of the number of communicating agents. Here, the regret is dominated by a $\log T$ term. In particular, by sharing the "hard to learn" information, the communicating agents significantly simplify their respective adaptive control problems. Even in the regime of large T, the above regret bound improves upon what is possible in the single task setting as long as the number of agents is sufficiently large.

5 Numerical Validation

We present numerical results to validate our bounds. In particular, we compare multi-task representation learning approach for the adaptive LQR design (Algorithm 1) over the setting where a single system attempts to learn its dynamics by using its local simulation data and computes a CE controller on top of the estimated model. To this end, our experimental setup considers H dynamical systems, described by (2), where the system matrices $(A_\star^{(h)}, B_\star^{(h)})$ are obtained by linearizing (around the origin) and discretizing (with Euler's approach) multiple cartpole dynamics with equations:

$$(M^{(h)} + m^{(h)})\ddot{x} + m^{(h)}\ell^{(h)}(\ddot{\theta}\cos(\theta) - \dot{\theta}^2 \sin(\theta)) = u,$$
$$m^{(h)}(\ddot{x}\cos(\theta) + \ell^{(h)}\ddot{\theta} - g\sin(\theta)) = 0, \quad (9)$$

for all $h \in [H]$, where $c_{\mathrm{p}}^{(h)} = (M^{(h)}, m^{(h)}, l^{(h)})$ denote the tuple of cartpole parameters. Such parameters represent the cart mass, pole mass, and pole length, respectively. We set the gravity $g = 1$ and perform the discretization of (9) with step-size 0.25. Following (Lee, Rantzer, and Matni 2024), we generate H $(A_\star^{(h)}, B_\star^{(h)})$, by first considering a set of *nominal* cartpole parameters: $c_{\mathrm{p}}^{(1)} = (0.4, 1.0, 1.0)$, $c_{\mathrm{p}}^{(2)} = (1.6, 1.3, 0.3)$, $c_{\mathrm{p}}^{(3)} = (1.3, 0.7, 0.65)$, $c_{\mathrm{p}}^{(4)} = (0.2, 0.055, 1.36)$, and $c_{\mathrm{p}}^{(5)} = (0.2, 0.47, 1.825)$.

We then perturb such parameters with a random scalar within the interval $(0, 0.1)$ to generate different cartpole parameters $c_p^{(h)}$. With the system matrices $(A_\star^{(h)}, B_\star^{(h)})$ in hands, for all $h \in [H]$, we generate the disturbance signal as $w_t^{(h)} \sim \mathcal{N}(0, 0.01 I_{d_{\mathsf{X}}})$ and set the step-size and number of iterations of Algorithm 3 as $\eta = 0.25$, and $N = 1000$. It is worth noting that step 2 of Algorithm 3 is considered for the simplicity of the theoretical analysis only, in our experiments we exploit the entire dataset for all `DFW` iterations.

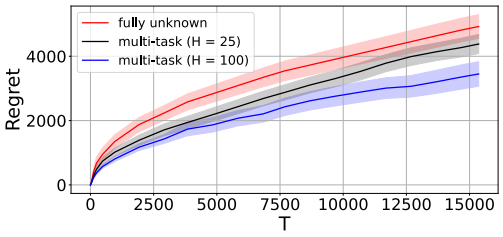

Figure 1: Regret of Algorithm 1 with varying number of tasks H. We consider $k_{\mathsf{fin}} = 10$ epochs with initial epoch length $\tau_1 = 30$, an exploratory sequence scaling as $\sigma_k^2 \propto \frac{1}{\sqrt{2^k}}$, state and controller bounds $x_b = 25$, and $K_b = 15$, and random Φ_0 with $d(\Phi_0, \Phi_\star) \approx 0.99$.

Figure 1 shows the expected regret of Algorithm 1 as a function of timesteps T for varying tasks H, with respect to a nominal task $h = 1$. This figure illustrates the easily identifiable setting where Assumption 4.3 is satisfied. The "fully-unknown" curve represents a single system estimating its dynamics and computing its controller using only its own data. As predicted in our bounds (Theorem 4.2), learning the representation in a multi-task setting reduces expected regret compared to the fully-unknown case. In the single-task setting, regret is $\mathcal{O}(\sqrt{T})$, while in the easily identifiable multi-task setting, it is $\mathcal{O}\left(\frac{\sqrt{T}}{\sqrt{H}}\right)$. Thus, as H increases, the regret decreases, as seen by comparing $H = 25$ to $H = 100$, both of which improve upon the fully-unknown setting.

6 Conclusion

We proposed an algorithm for simultaneous adaptive control of multiple linear dynamical systems sharing a representation. Leveraging recent results for representation learning with non-iid data, we provide non-asymptotic regret bounds in two settings: one where system-specific parameters are easily identified from the shared representation, and one where they are not. In the easily identifiable setting, the regret scales as $\sqrt{T}/\sqrt{H}$, while in the difficult-to-identify setting, it scales as $T^{3/4}/H^{1/5}$. Future work could explore improving the $T^{3/4}/H^{1/5}$ regret bound to $\sqrt{T}/\sqrt{H}$ in the difficult setting and extend the analysis to nonlinear systems, as done in the single-task setting (Boffi, Tu, and Slotine 2021).

Acknowledgements

BL, TZ, and NM gratefully acknowledge support from NSF Award SLES 2331880, NSF CAREER award ECCS 2045834, and NSF EECS 2231349. LT is funded by the Center for AI and Responsible Financial Innovation (CAIRFI) Fellowship and by the Columbia Presidential Fellowship. JA is partially funded by NSF grants ECCS 2144634 and 2231350 and the Columbia Data Science Institute.

References

Abbasi-Yadkori, Y.; and Szepesvári, C. 2011. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, 1–26. JMLR Workshop and Conference Proceedings.

Baxter, J. 2000. A model of inductive bias learning. *Journal of artificial intelligence research*, 12: 149–198.

Boffi, N. M.; Tu, S.; and Slotine, J.-J. E. 2021. Regret bounds for adaptive nonlinear control. In *Learning for Dynamics and Control*, 471–483. PMLR.

Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Chen, X.; Choromanski, K.; Ding, T.; Driess, D.; Dubey, A.; Finn, C.; et al. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*.

Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Dabis, J.; Finn, C.; Gopalakrishnan, K.; Hausman, K.; Herzog, A.; Hsu, J.; et al. 2022. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*.

Cassel, A.; Cohen, A.; and Koren, T. 2020. Logarithmic regret for learning linear quadratic regulators efficiently. In *International Conference on Machine Learning*, 1328–1337. PMLR.

Cohen, A.; Koren, T.; and Mansour, Y. 2019. Learning Linear-Quadratic Regulators Efficiently with only $\sqrt{T}$ Regret. In *International Conference on Machine Learning*, 1300–1309. PMLR.

Collins, L.; Hassani, H.; Mokhtari, A.; and Shakkottai, S. 2021. Exploiting shared representations for personalized federated learning. In *International Conference on Machine Learning*, 2089–2099. PMLR.

Dean, S.; Mania, H.; Matni, N.; Recht, B.; and Tu, S. 2018. Regret bounds for robust adaptive control of the linear quadratic regulator. *Advances in Neural Information Processing Systems*, 31.

Du, S. S.; Hu, W.; Kakade, S. M.; Lee, J. D.; and Lei, Q. 2020. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, 1126–1135. PMLR.

Ghai, U.; Chen, X.; Hazan, E.; and Megretski, A. 2022. Robust online control with model misspecification. In *Learning for Dynamics and Control Conference*, 1163–1175. PMLR.

Goyal, A.; Xu, J.; Guo, Y.; Blukis, V.; Chao, Y.-W.; and Fox, D. 2023. RVT: Robotic View Transformer for 3D Object Manipulation. *arXiv preprint arXiv:2306.14896*.

Gregory, P. 1959. *Proceedings of the Self Adaptive Flight Control Systems Symposium*, volume 59. Wright Air Development Center, Air Research and Development Command, United ....

Guo, T.; Al Makdah, A. A.; Krishnan, V.; and Pasqualetti, F. 2023. Imitation and transfer learning for LQG control. *IEEE Control Systems Letters*.

Hazan, E.; Kakade, S.; and Singh, K. 2020. The nonstochastic control problem. In *Algorithmic Learning Theory*, 408–421. PMLR.

Horn, R. A.; and Johnson, C. R. 2012. *Matrix analysis*. Cambridge university press.

Jedra, Y.; and Proutiere, A. 2022. Minimal expected regret in linear quadratic control. In *International Conference on Artificial Intelligence and Statistics*, 10234–10321. PMLR.

Kumar, A.; Singh, A.; Ebert, F.; Nakamoto, M.; Yang, Y.; Finn, C.; and Levine, S. 2022. Pre-training for robots: Offline rl enables learning new tasks from a handful of trials. *arXiv preprint arXiv:2210.05178*.

Lee, B.; Rantzer, A.; and Matni, N. 2024. Nonasymptotic regret analysis of adaptive linear quadratic control with model misspecification. In *6th Annual Learning for Dynamics & Control Conference*, 980–992. PMLR.

Ma, X.; Zhu, J.; Lin, Z.; Chen, S.; and Qin, Y. 2022. A state-of-the-art survey on solving non-IID data in Federated Learning. *Future Generation Computer Systems*, 135: 244–258.

Mania, H.; Tu, S.; and Recht, B. 2019. Certainty equivalence is efficient for linear quadratic control. *Advances in Neural Information Processing Systems*, 32.

Modi, A.; Faradonbeh, M. K. S.; Tewari, A.; and Michailidis, G. 2021. Joint learning of linear time-invariant dynamical systems. *arXiv preprint arXiv:2112.10955*.

Petersen, K. B.; Pedersen, M. S.; et al. 2008. The matrix cookbook. *Technical University of Denmark*, 7(15): 510.

Simchowitz, M.; and Foster, D. 2020. Naive exploration is optimal for online lqr. In *International Conference on Machine Learning*, 8937–8948. PMLR.

Simchowitz, M.; Singh, K.; and Hazan, E. 2020. Improper learning for non-stochastic control. In *Conference on Learning Theory*, 3320–3436. PMLR.

Stewart, G. W.; and Sun, J.-g. 1990. *Matrix perturbation theory*. Academic press.

Tan, Y.; Long, G.; Liu, L.; Zhou, T.; Lu, Q.; Jiang, J.; and Zhang, C. 2022. Fedproto: Federated prototype learning across heterogeneous clients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8432–8440.

Thekumparampil, K. K.; Jain, P.; Netrapalli, P.; and Oh, S. 2021. Sample Efficient Linear Meta-Learning by Alternating Minimization. arXiv:2105.08306.

Toso, L. F.; Zhan, D.; Anderson, J.; and Wang, H. 2024. Meta-Learning Linear Quadratic Regulators: A Policy Gradient MAML Approach for the Model-free LQR. *arXiv preprint arXiv:2401.14534*.

Tripuraneni, N.; Jin, C.; and Jordan, M. 2021. Provable meta-learning of linear representations. In *International Conference on Machine Learning*, 10434–10443. PMLR.

Tripuraneni, N.; Jordan, M.; and Jin, C. 2020. On the theory of transfer learning: The importance of task diversity. *Advances in Neural Information Processing Systems*, 33: 7852–7862.

Vaswani, N. 2024. Efficient Federated Low Rank Matrix Recovery via Alternating GD and Minimization: A Simple Proof. *IEEE Transactions on Information Theory*.

Vershynin, R. 2018. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press.

Wang, H.; Toso, L. F.; Mitra, A.; and Anderson, J. 2023a. Model-free learning with heterogeneous dynamical systems: A federated lqr approach. *arXiv preprint arXiv:2308.11743*.

Wang, L.; Zhang, K.; Zhou, A.; Simchowitz, M.; and Tedrake, R. 2023b. Fleet Policy Learning via Weight Merging and An Application to Robotic Tool-Use. *arXiv preprint arXiv:2310.01362*.

Zhang, T. T.; Kang, K.; Lee, B. D.; Tomlin, C.; Levine, S.; Tu, S.; and Matni, N. 2023a. Multi-task imitation learning for linear dynamical systems. In *Learning for Dynamics and Control Conference*, 586–599. PMLR.

Zhang, T. T.; Toso, L. F.; Anderson, J.; and Matni, N. 2023b. Meta-Learning Operators to Optimality from Multi-Task Non-IID Data. *arXiv preprint arXiv:2308.04428*.

Zhang, T. T.; Toso, L. F.; Anderson, J.; and Matni, N. 2024. Sample-Efficient Linear Representation Learning from Non-IID Non-Isotropic Data. In *The Twelfth International Conference on Learning Representations*.

Ziemann, I.; Tsiamis, A.; Lee, B.; Jedra, Y.; Matni, N.; and Pappas, G. J. 2023. A Tutorial on the Non-Asymptotic Theory of System Identification. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, 8921–8939. IEEE.