Near-Optimal Representation Learning for Linear Bandits and Linear RL

Jiachen Hu [*1] **Xiaoyu Chen** [*1] **Chi Jin** [2] **Lihong Li** [3] **Liwei Wang** [1 4]

Abstract

This paper studies representation learning for multi-task linear bandits and multi-task episodic RL with linear value function approximation. We first consider the setting where we play M linear bandits with dimension d concurrently, and these bandits share a common k-dimensional linear representation so that $k \ll d$ and $k \ll M$. We propose a sample-efficient algorithm, MTLR-OFUL, which leverages the shared representation to achieve $\tilde{O}(M\sqrt{dkT} + d\sqrt{kMT})$ regret, with T being the number of total steps. Our regret significantly improves upon the baseline $\tilde{O}(Md\sqrt{T})$ achieved by solving each task independently. We further develop a lower bound that shows our regret is near-optimal when $d > M$. Furthermore, we extend the algorithm and analysis to multi-task episodic RL with linear value function approximation under low inherent Bellman error (Zanette et al., 2020a). To the best of our knowledge, this is the first theoretical result that characterize the benefits of multi-task representation learning for exploration in RL with function approximation.

1. Introduction

Multi-task representation learning is the problem of learning a common low-dimensional representation among multiple related tasks (Caruana, 1997). This problem has become increasingly important in many applications such as natural language processing (Ando & Zhang, 2005; Liu et al., 2019), computer vision (Li et al., 2014), drug discovery (Ramsundar et al., 2015), and reinforcement learning (Wilson et al., 2007; Teh et al., 2017; D'Eramo et al., 2019). In these cases, common information can be extracted from related tasks to improve data efficiency and accelerate learning.

While representation learning has achieved tremendous success in a variety of applications (Bengio et al., 2013), its theoretical understanding is still limited. A widely accepted assumption in the literature is the existence of a common representation shared by different tasks. For example, Maurer et al. (2016) proposed a general method to learn data representation in multi-task supervised learning and learning-to-learn setting. Du et al. (2020) studied few-shot learning via representation learning with assumptions on a common representation among source and target tasks. Tripuraneni et al. (2020) focused on the problem of multi-task linear regression with low-rank representation, and proposed algorithms with sharp statistical rates.

Inspired by the theoretical results in supervised learning, we take a step further to investigate provable benefits of representation learning for sequential decision making problems. First, we study the multi-task low-rank linear bandits problem, where M tasks of d-dimensional (infinite-arm) linear bandits are concurrently learned for T steps. The expected reward of arm $\boldsymbol{x}_i \in \mathbb{R}^d$ for task i is $\boldsymbol{\theta}_i^\top \boldsymbol{x}_i$, as determined by an unknown linear parameter $\boldsymbol{\theta}_i$. To take advantage of the multi-task representation learning framework, we assume that $\boldsymbol{\theta}_i$'s lie in an unknown k-dimensional subspace of $\mathbb{R}^d$, where k is much smaller compared to d and M (Yang et al., 2020). The dependence among tasks makes it possible to achieve a regret bound better than solving each task independently. Specifically, if the tasks are solved independently with standard algorithms such as OFUL (Abbasi-Yadkori et al., 2011), the total regret is $\tilde{O}(Md\sqrt{T})$.[1] By leveraging the common representation among tasks, we can achieve a better regret $\tilde{O}(M\sqrt{dkT} + d\sqrt{MkT})$. Our algorithm is also robust to the linear representation assumption when the model is misspecified. If the k-dimensional subspace approximates the rewards with error at most ζ, our algorithm can still achieve regret $\tilde{O}(M\sqrt{dkT} + d\sqrt{kMT} + MT\sqrt{d}\zeta)$. Moreover, we prove a regret lower bound indicating that the regret of our algorithm is not improvable except for logarithmic factors in the regime $d > M$.

Compared with multi-task linear bandits, multi-task reinforcement learning is a more popular research topic with a long line of works in both theoretical side and empirical side (Taylor & Stone, 2009; Parisotto et al., 2015; Liu

[*] Equal contribution [1] Key Laboratory of Machine Perception, MOE, School of EECS, Peking University [2] Department of Electrical and Computer Engineering, Princeton University [3] Amazon [4] Center for Data Science, Peking University. Correspondence to: Liwei Wang <wanglw@cis.pku.edu.cn>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

[1] $\tilde{O}$ hides the logarithmic factors.

et al., 2016; Teh et al., 2017; Hessel et al., 2019; D'Eramo et al., 2019; Arora et al., 2020). We extend our algorithm for linear bandits to the multi-task episodic reinforcement learning with linear value function approximation under low inherent Bellman error (Zanette et al., 2020a). Assuming a low-rank linear representation across all the tasks, we propose a sample-efficient algorithm with regret $\tilde{O}(HM\sqrt{dkT}+Hd\sqrt{kMT}+HMT\sqrt{d}\mathcal{I})$, where k is the dimension of the low-rank representation, d is the ambient dimension of state-action features, M is the number of tasks, H is the horizon, T is the number of episodes, and $\mathcal{I}$ denotes the inherent Bellman error. The regret significantly improves upon the baseline regret $\tilde{O}(HMd\sqrt{T}+HMT\sqrt{d}\mathcal{I})$ achieved by running ELEANOR algorithm (Zanette et al., 2020a) for each task independently. We also prove a regret lower bound $\Omega(Mk\sqrt{HT}+d\sqrt{HkMT}+HMT\sqrt{d}\mathcal{I})$. To the best of our knowledge, this is the first provably sample-efficient algorithm for exploration in multi-task low-rank linear RL.

2. Preliminaries

2.1. Multi-Task Linear Bandit

We study the problem of representation learning for linear bandits in which there are multiple tasks sharing common low-dimensional features. Let d be the ambient dimension and k be the representation dimension. We play M tasks concurrently for T steps each. Each task $i \in [M]$ is associated with an unknown vector $\boldsymbol{\theta}_i \in \mathbb{R}^d$. In each step $t \in [T]$, the player chooses one action $\boldsymbol{x}_{t,i} \in \mathcal{A}_{t,i}$ for each task $i \in [M]$, and receives a batch of rewards $\{y_{t,i}\}_{i=1}^M$ afterwards, where $\mathcal{A}_{t,i}$ is the feasible action set (can even be chosen adversarially) for task i at step t. The rewards received are determined by $y_{t,i} = \boldsymbol{\theta}_i^\top \boldsymbol{x}_{t,i} + \eta_{t,i}$, where the $\eta_{t,i}$ is the random noise.

We use total regret for M tasks in T steps to measure the performance of our algorithm, which is defined in the following way:

$$\mathrm{Reg}(T) \overset{\text{def}}{=} \sum_{t=1}^{T}\sum_{i=1}^{M} \left(\langle \boldsymbol{x}_{t,i}^\star, \boldsymbol{\theta}_i\rangle - \langle \boldsymbol{x}_{t,i}, \boldsymbol{\theta}_i\rangle\right),$$

where $\boldsymbol{x}_{t,i}^\star = \operatorname{argmax}_{\boldsymbol{x}\in\mathcal{A}_{t,i}} \langle \boldsymbol{x}, \boldsymbol{\theta}_i\rangle$.

The main assumption is the existence of a common linear feature extractor.

Assumption 1. *There exists a linear feature extractor $\boldsymbol{B} \in \mathbb{R}^{d\times k}$ and a set of k-dimensional coefficients $\{\boldsymbol{w}_i\}_{i=1}^M$ such that $\{\boldsymbol{\theta}_i\}_{i=1}^M$ satisfies $\boldsymbol{\theta}_i = \boldsymbol{B}\boldsymbol{w}_i$.*

Define filtration F_t to be the σ-field of random variables $\sigma(\{x_{\tau,i}\}_{\tau\le t+1, i\in[M]}, \{\eta_{\tau,i}\}_{\tau\le t, i\in[M]})$, then we have the following assumption.

Assumption 2. *Following the standard regularity assumptions in linear bandits (Abbasi-Yadkori et al., 2011; Lattimore & Szepesvári, 2020), we assume*

- $\|\boldsymbol{\theta}_i\|_2 \le 1, \forall i \in [M]$
- $\|\boldsymbol{x}\|_2 \le 1, \forall \boldsymbol{x} \in \mathcal{A}_{t,i}, t \in [T], i \in [M]$
- *$\eta_{t,i}$ is conditionally zero-mean 1-sub-Gaussian random variable with regards to F_{t-1}.*

For notation convenience, we use $\boldsymbol{X}_{t,i} = [\boldsymbol{x}_{1,i}, \boldsymbol{x}_{2,i}, \cdots, \boldsymbol{x}_{t,i}]$ and $\boldsymbol{y}_{t,i} = [y_{1,i}, \cdots, y_{t,i}]^\top$ to denote the arms and the corresponding rewards collected for task $i \in [M]$ in the first t steps, and we also use $\boldsymbol{\eta}_{t,i} = [\eta_{1,i}, \eta_{2,i}, \cdots, \eta_{t,i}]^\top$ to denote the corresponding noise. We define $\boldsymbol{\Theta} \overset{\text{def}}{=} [\boldsymbol{\theta}_1, \boldsymbol{\theta_2}, \cdots, \boldsymbol{\theta_M}]$ and $\boldsymbol{W} \overset{\text{def}}{=} [\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_M]$. For any positive definite matrix $\boldsymbol{A} \in \mathbb{R}^{d\times d}$, the Mahalanobis norm with regards to $\boldsymbol{A}$ is denoted by $\|\boldsymbol{x}\|_{\boldsymbol{A}} = \sqrt{\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x}}$.

2.2. Multi-Task Linear RL

We also study how this low-rank structure benefits the exploration problem with approximate linear value functions in multi-task episodic reinforcement learning. For reference convenience, we abbreviate our setting as multi-task LSVI setting, which is a natural extension of LSVI condition in the single-task setting (Zanette et al., 2020a).

Consider an undiscounted episodic MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, p, r, H)$ with state space $\mathcal{S}$, action space $\mathcal{A}$, and fixed horizon H. At each step $h \in [H]$, the agent receives a random reward $R_h(s_h, a_h)$ with mean $r_h(s_h, a_h)$ based on the state s_h he is located and action a_h he takes. Then he transits to the next state s_{h+1} according to the transition kernel $p_h(\cdot \mid s_h, a_h)$. The action value function for each state-action pair at step h for some deterministic policy π is defined as $Q_h^\pi(s_h, a_h) \overset{\text{def}}{=} r_h(s_h, a_h) + \mathbb{E}\left[\sum_{t=h+1}^H R_t(s_t, \pi_t(s_t))\right]$, and the state value function is defined as $V_h^\pi(s_h) = Q_h^\pi(s_h, \pi_h(s_h))$

Note that there always exists an optimal deterministic policy (under some regularity conditions) π^* for which $V_h^{\pi^*}(s) = \max_\pi V_h^\pi(s)$ and $Q_h^{\pi^*}(s,a) = \max_\pi Q_h^\pi(s,a)$ for each $h \in [H]$. We denote $V_h^{\pi^*}$ and $Q_h^{\pi^*}$ by V_h^* and Q_h^* for short.

It's also convenient to define the Bellman optimality operator $\mathcal{T}_h$ as $\mathcal{T}_h(Q_{h+1})(s,a) \overset{\text{def}}{=} r_h(s,a) + \mathbb{E}_{s'\sim p_h(\cdot|s,a)} \max_{a'} Q_{h+1}(s', a')$.

In the framework of single-task approximate linear value functions (see Section 5 for more discussions), we assume a feature map $\boldsymbol{\phi} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ that maps each state-action pair to a d-dimensional vector. In case that $\mathcal{S}$ is too large or continuous (e.g. in robotics), this feature map helps to reduce the problem scale from $|\mathcal{S}| \times |\mathcal{A}|$ to d. The value functions are the linear combinations of those feature maps, so we can define the function space at step $h \in [H]$ to be $\mathcal{Q}'_h = \{Q_h(\boldsymbol{\theta}_h) \mid \boldsymbol{\theta}_h \in \Theta'_h\}$ and $\mathcal{V}'_h = \{V_h(\boldsymbol{\theta}_h) \mid \boldsymbol{\theta}_h \in \Theta'_h\}$, where $Q_h(\boldsymbol{\theta}_h)(s,a) \overset{\text{def}}{=} \boldsymbol{\phi}(s,a)^\top \boldsymbol{\theta}_h$, and $V_h(\theta_h)(s) \overset{\text{def}}{=} \max_a \boldsymbol{\phi}(s,a)^\top \boldsymbol{\theta}_h$.

In order to find the optimal value function using value iteration with $\mathcal{Q}_h$, we require that it is approximately close under $\mathcal{T}_h$, as measured by the inherent Bellman error (or IBE for short). The IBE (Zanette et al., 2020a) at step h is defined as

$$\mathcal{I}_h \overset{\text{def}}{=} \sup_{Q_{h+1}\in\mathcal{Q}_{h+1}} \inf_{Q_h\in\mathcal{Q}_h} \sup_{s\in\mathcal{S},a\in\mathcal{A}} \left|\left(Q_h - \mathcal{T}_h(Q_{h+1})\right)(s,a)\right|. \tag{1}$$

In multi-task reinforcement learning, we have M MDPs $\mathcal{M}^1, \mathcal{M}^2, ..., \mathcal{M}^M$ (we use superscript i to denote task i). Assume they share the same state space and action space, but have different rewards and transitions.

To take advantage of the multi-task LSVI setting and low-rank representation learning, we define a joint function space for all the tasks as $\Theta_h \overset{\text{def}}{=} \{(\boldsymbol{B}_h\boldsymbol{w}_h^1, \boldsymbol{B}_h\boldsymbol{w}_h^2, \cdots, \boldsymbol{B}_h\boldsymbol{w}_h^M) : \boldsymbol{B}_h \in \mathcal{O}^{d\times k}, \boldsymbol{w}_h^i \in \mathcal{B}^k, \boldsymbol{B}_h\boldsymbol{w}_h^i \in \Theta_h^{i\prime}\}$, where $\mathcal{O}^{d\times k}$ is the collection of all orthonormal matrices in $\mathbb{R}^{d\times k}$, and $\mathcal{B}^k$ is the unit ball in $\mathbb{R}^k$.

The induced function space is defined as

$$\begin{aligned} \mathcal{Q}_h \overset{\text{def}}{=} \{&\left(Q_h^1\left(\boldsymbol{\theta}_h^1\right), Q_h^2\left(\boldsymbol{\theta}_h^2\right), \cdots, Q_h^M\left(\boldsymbol{\theta}_h^M\right)\right) && (2)\\ &\mid \left(\boldsymbol{\theta}_h^1, \boldsymbol{\theta}_h^2, \cdots, \boldsymbol{\theta}_h^M\right) \in \Theta_h\} && (3)\\ \mathcal{V}_h \overset{\text{def}}{=} \{&\left(V_h^1\left(\boldsymbol{\theta}_h^1\right), V_h^2\left(\boldsymbol{\theta}_h^2\right), \cdots, V_h^M\left(\boldsymbol{\theta}_h^M\right)\right) && (4)\\ &\mid \left(\boldsymbol{\theta}_h^1, \boldsymbol{\theta}_h^2, \cdots, \boldsymbol{\theta}_h^M\right) \in \Theta_h\} && (5) \end{aligned}$$

The low-rank IBE at step h for multi-task LSVI setting is a generalization of IBE (Eqn 1) for the single-task setting, which is defined accordingly as

$$\begin{aligned} \mathcal{I}_h^{\text{mul}} \overset{\text{def}}{=} & \sup_{\{Q_{h+1}^i\}_{i=1}^M \in \mathcal{Q}_{h+1}} \inf_{\{Q_h^i\}_{i=1}^M \in \mathcal{Q}_h} && (6)\\ & \sup_{s\in\mathcal{S},a\in\mathcal{A},i\in[M]} \left|\left(Q_h^i - \mathcal{T}_h^i(Q_{h+1}^i)\right)(s,a)\right| && (7) \end{aligned}$$

Let $\mathcal{I} \overset{\text{def}}{=} \sup_h \mathcal{I}_h^{\text{mul}}$ be the maximum inherent Bellman error with regards to the joint function space $\mathcal{Q}_h$ over $h \in [H]$. When $\mathcal{I} = 0$, this multi-task RL problem can be regarded as a natural extension of Assumption 1 in linear bandits to episodic RL. This is because there exists $\{\bar{\boldsymbol{\theta}}_h^{i*}\}_{i=1}^M \in \Theta_h$ such that $Q_h^{i*} = Q_h^i(\bar{\boldsymbol{\theta}}_h^{i*})$ for all $i \in [M]$ and $h \in [H]$ in the case $\mathcal{I} = 0$. According to the definition of Θ_h we know that $\{\bar{\boldsymbol{\theta}}_h^{i*}\}_{i=1}^M$ also admit a low-rank property as Assumption 1 indicates. When $\mathcal{I} > 0$, it is an extension of misspecified multi-task linear bandits (discussed in Section 4.3) to episodic RL.

Define the filtration $\mathcal{F}_{h,t}$ to be the σ-field induced by all the random variables up to step h in episode t (not include the rewards at step h in episode t), then we have the following assumptions.

Assumption 3. *Following the parameter scale in (Zanette et al., 2020a), we assume*

- $\|\boldsymbol{\phi}(s,a)\|_2 \leq 1, \forall(s,a) \in \mathcal{S} \times \mathcal{A}, h \in [H]$
- $0 \leq Q_h^\pi(s,a) \leq 1, \forall(s,a) \in \mathcal{S} \times \mathcal{A}, h \in [H], \forall\pi.$
- *There exists constant D that for any $h \in [H]$ and any $\left\{\boldsymbol{\theta}_h^i\right\}_{i=1}^M \in \Theta_h$, it holds that $\|\boldsymbol{\theta}_h^i\|_2 \leq D, \forall i \in [M]$.*
- *For any fixed $\left\{Q_{h+1}^i\right\}_{i=1}^M \in \mathcal{Q}_{h+1}$, the random noise $z_h^i(s,a) \overset{\text{def}}{=} R_h^i(s,a) + \max_a Q_{h+1}^i(s',a) - \mathcal{T}_h^i\left(Q_{h+1}^i\right)(s,a)$ is bounded in $[-1,1]$ a.s., and is independent conditioned on $\mathcal{F}_{h,t}$ for any $s \in \mathcal{S}, a \in \mathcal{A}, h \in [H], i \in [M]$, where the randomness is from reward R and $s' \sim p_h(\cdot \mid s,a)$.*

The first condition is a standard regularization condition for linear features. The second condition is on the scale of the problem. This scale of the exploration problem that the value function is bounded in $[0,1]$ has also been studied in both tabular and linear setting (Zhang et al., 2020; Wang et al., 2020; Zanette et al., 2020a). The last two conditions are compatible with the scale of the problem. It's sufficient to assume the constant norm of $\boldsymbol{\theta}_h^i$ since the optimal value function is of the same scale. The last condition is standard in linear bandits (Abbasi-Yadkori et al., 2011; Lattimore & Szepesvári, 2020) and RL (Zanette et al., 2020a), and is automatically satisfied if $D = 1$.

The total regret of M tasks in T episodes is defined as

$$\operatorname{Reg}(T) \overset{\text{def}}{=} \sum_{t=1}^T \sum_{i=1}^M \left(V_1^{i*} - V_1^{\pi_t^i}\right)\left(s_{1t}^i\right) \tag{8}$$

where π_t^i is the policy used for task i in episode t, and s_{ht}^i denotes the state encountered at step h in episode t for task i. We assume $M \geq 5, T \geq 5$ throughout this paper.

3. Related Work

Multi-task Supervised Learning The idea of multi-task representation learning at least dates back to Caruana (1997); Thrun & Pratt (1998); Baxter (2000). Empirically, representation learning has shown its great power in various domains. We refer readers to Bengio et al. (2013) for a detailed review about empirical results. From the theoretical perspective, Baxter (2000) performed the first theoretical analysis and gave sample complexity bounds using covering number. Maurer et al. (2016) considered the setting where all tasks are sampled from a certain distribution, and analyzed the benefits of representation learning for reducing the sample complexity of the target task. Following their results, Du et al. (2020) and Tripuraneni et al. (2020) replaced the i.i.d assumption with a deterministic assumption on the data distribution and task diversity, and proposed efficient algorithms that can fully utilize all source data with better sample complexity. These results mainly focus on the statistical rate for multi-task supervised learning, and cannot tackle the exploration problem in bandits and RL.

Multi-task Bandit Learning For multi-task linear bandits, the most related work is a recent paper by Yang et al. (2020). For linear bandits with infinite-action set, they firstly proposed an explore-then-exploit algorithm with regret $\tilde{O}(Mk\sqrt{T}+d^{1.5}k\sqrt{MT})$, which outperforms the naive approach with $\tilde{O}(Md\sqrt{T})$ regret in the regime $M=\Omega(dk^2)$. Though their results are insightful, they required the action set for all tasks and all steps to be the same well-conditioned d-dimensional ellipsoids which cover all directions with constant radius. Besides, they assumed that the task parameters are diverse enough with $\boldsymbol{W}\boldsymbol{W}^\top$ well-conditioned, and the norm of $\boldsymbol{w}_i$ is lower bounded by a constant. These assumptions make the application of the theory rather restrictive to only a subset of linear bandit instances with benign structures. In contrast, our theory is more general since we do not assume the same and well-conditioned action set for different tasks and time steps, nor assume the benign properties of $\boldsymbol{w}_i$'s.

Multi-task RL For multi-task reinforcement learning, there is a long line of works from the empirical perspective (Taylor & Stone, 2009; Parisotto et al., 2015; Liu et al., 2016; Teh et al., 2017; Hessel et al., 2019). From the theoretical perspective, Brunskill & Li (2013) analyzed the sample complexity of multi-task RL in the tabular setting. D'Eramo et al. (2019) showed that representation learning can improve the rate of approximate value iteration algorithm. Arora et al. (2020) proved that representation learning can reduce the sample complexity of imitation learning.

Bandits with Low Rank Structure Low-rank representations have also been explored in single-task settings. Jun et al. (2019) studied bilinear bandits with low rank representation. The mean reward in their setting is defined as the bilinear multiplication $\boldsymbol{x}^\top\boldsymbol{\Theta}\boldsymbol{y}$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are two actions selected at each step, and $\boldsymbol{\Theta}$ is an unknown low rank parameter matrix. Their setting is further generalized by Lu et al. (2020). Furthermore, sparse linear bandits can be regarded as a simplified setting, where $\boldsymbol{B}$ is a binary matrix indicating the subset of relevant features in context $\boldsymbol{x}$ (Abbasi-Yadkori et al., 2012; Carpentier & Munos, 2012; Lattimore et al., 2015; Hao et al., 2020).

Exploration in Bandits and RL Our regret analysis is also related to exploration in single-task linear bandits and linear RL. Linear bandits have been extensively studied in recent years (Auer, 2002; Dani et al., 2008; Rusmevichientong & Tsitsiklis, 2010; Abbasi-Yadkori et al., 2011; Chu et al., 2011; Li et al., 2019a;b). Our algorithm is most relevant to the seminal work of Abbasi-Yadkori et al. (2011), who applied self-normalized techniques to obtain near-optimal regret upper bounds. For single-task linear RL, recent years have witnessed a tremendous of works under different function approximation settings, including linear MDPs (Yang & Wang, 2019; Jin et al., 2020), linear mixture MDPs (Ayoub et al., 2020; Zhou et al., 2020a), linear RL with low inherent Bellman error (Zanette et al., 2020a;b), and MDPs with low Bellman-rank (Jiang et al., 2017). Our multi-task setting is a natural extension of linear RL with low inherent Bellman error setting, which covers linear MDP setting as a special case (Zanette et al., 2020a).

4. Main Results for Linear Bandits

In this section, we present our main results for multi-task linear bandits.

4.1. Construction of the Confidence Sets

A natural and successful method to design efficient algorithms for sequential decision making problem is the *optimism in the face of uncertainty principle*. When applied to single-task linear bandits, the basic idea is to maintain a confidence set $\mathcal{C}_t$ for the parameter $\boldsymbol{\theta}$ based on history observations for each step $t\in[T]$. The algorithm chooses an optimistic estimation $\tilde{\boldsymbol{\theta}}_t=\operatorname{argmax}_{\boldsymbol{\theta}\in\mathcal{C}_t}\left(\max_{\boldsymbol{x}\in\mathcal{A}_t}\langle\boldsymbol{x},\boldsymbol{\theta}\rangle\right)$ and then selects action $\boldsymbol{x}_t=\operatorname{argmax}_{\boldsymbol{x}_t\in\mathcal{A}_t}\langle\boldsymbol{x},\tilde{\boldsymbol{\theta}}_t\rangle$, which maximizes the reward according to the estimation $\tilde{\boldsymbol{\theta}}_t$.

For multi-task linear bandits, the main difference is that we need to tackle M highly correlated tasks concurrently. To obtain tighter confidence bound, we maintain the confidence set $\mathcal{C}_t$ for $\boldsymbol{B}$ and $\{\boldsymbol{w}_i\}_{i=1}^M$, then choose the optimistic estimation $\tilde{\boldsymbol{\Theta}}_t$ for all tasks concurrently. To be more specific, the algorithm chooses an optimistic estimate $\tilde{\boldsymbol{\Theta}}_t=\operatorname{argmax}_{\boldsymbol{\Theta}\in\mathcal{C}_t}(\max_{\{x_i\in\mathcal{A}_{t,i}\}_{i=1}^M}\sum_{i=1}^M\langle\boldsymbol{x}_i,\boldsymbol{\theta}_i\rangle)$, and then selects action $\boldsymbol{x}_{t,i}=\operatorname{argmax}_{x_i\in\mathcal{A}_{t,i}}\langle\boldsymbol{x}_i,\tilde{\boldsymbol{\theta}}_{t,i}\rangle$ for each task $i\in[M]$.

The main technical contribution is the construction of a tighter confidence set $\mathcal{C}_t$ for the estimation of $\boldsymbol{\Theta}$. At each step $t\in[T]$, we solve the following least-square problem based on the samples collected so far and obtain the minimizer $\hat{\boldsymbol{B}}_t$ and $\hat{\boldsymbol{W}}_t$:

$$\underset{\boldsymbol{B}\in\mathbb{R}^{d\times k},\boldsymbol{w}_{1..M}\in\mathbb{R}^{k\times M}}{\arg\min} \sum_{i=1}^{M} \left\|\boldsymbol{y}_{t-1,i} - \boldsymbol{X}_{t-1,i}^\top \boldsymbol{B}\boldsymbol{w}_i\right\|_2^2 \tag{9}$$

$$\text{s.t.} \quad \left\|\boldsymbol{B}\boldsymbol{w}_i\right\|_2 \leq 1, \forall i \in [M]. \tag{10}$$

The confidence set $\mathcal{C}_t$ is constructed as follows:

$$\mathcal{C}_t \overset{\text{def}}{=} \Bigg\{ \boldsymbol{\Theta} = \boldsymbol{B}\boldsymbol{W} : \sum_{i=1}^{M} \left\|\hat{\boldsymbol{B}}_t\hat{\boldsymbol{w}}_{t,i} - \boldsymbol{B}\boldsymbol{w}_i\right\|^2_{\tilde{\boldsymbol{V}}_{t-1,i}(\lambda)} \leq L, \\ \boldsymbol{B} \in \mathbb{R}^{d\times k}, \boldsymbol{w}_i \in \mathbb{R}^k, \left\|\boldsymbol{B}\boldsymbol{w}_i\right\|_2 \leq 1, \forall i \in [M] \Bigg\}, \tag{11}$$

where $L = \tilde{O}(Mk + kd)$ (see Appendix A.1 for the exact value) and $\tilde{\boldsymbol{V}}_{t-1,i}(\lambda) = \boldsymbol{X}_{t-1,i}\boldsymbol{X}_{t-1,i}^\top + \lambda \boldsymbol{I}_d$. λ is a hyperparameter used to ensure that $\tilde{\boldsymbol{V}}_{t-1,i}(\lambda)$ is always invertable, which can be set to 1. We can guarantee that $\boldsymbol{\Theta} \in \mathcal{C}_t$ for all $t \in [T]$ with high probability by the following lemma.

Lemma 1. *With probability at least $1 - \delta$, for any step $t \in [T]$, suppose $\hat{\boldsymbol{\Theta}}_t = \hat{\boldsymbol{B}}_t\hat{\boldsymbol{W}}_t$ is the optimal solution of the least-square regression (Eqn 9), the true parameter $\boldsymbol{\Theta} = \boldsymbol{B}\boldsymbol{W}$ is always contained in the confidence set $\mathcal{C}_t$, i.e.*

$$\sum_{i=1}^{M} \left\|\hat{\boldsymbol{B}}_t\hat{\boldsymbol{w}}_{t,i} - \boldsymbol{B}\boldsymbol{w}_i\right\|^2_{\tilde{\boldsymbol{V}}_{t-1,i}(\lambda)} \leq L, \tag{12}$$

where $\tilde{\boldsymbol{V}}_{t-1,i}(\lambda) = \boldsymbol{X}_{t-1,i}\boldsymbol{X}_{t-1,i}^\top + \lambda \boldsymbol{I}_d$.

If we solve each tasks independently with standard single-task algorithms such as OFUL (Abbasi-Yadkori et al., 2011), it is not hard to realize that we can only obtain a confidence set with $\sum_{i=1}^{M} \|\hat{\boldsymbol{B}}_t\hat{\boldsymbol{w}}_{t,i} - \boldsymbol{B}\boldsymbol{w}_i\|^2_{\tilde{\boldsymbol{V}}_{t-1,i}(\lambda)} \leq L_1 = \tilde{O}(Md)$. Our confidence bound is much sharper compared with this naive bound, which explains the improvement in our final regret. Compared with Yang et al. (2020), we are not able to estimate $\boldsymbol{B}$ and $\boldsymbol{W}$ directly like their methods due to the more relaxed bandit setting. In our setting, the empirical design matrix $\tilde{\boldsymbol{V}}_{t-1,i}(\lambda)$ can be quite ill-conditioned if the action set at each step is chosen adversarially. Thus, we have to establish a tighter confidence set to improve the regret bound.

We only sketch the main idea of the proof for Lemma 1 and defer the detailed explanation to Appendix A.1. Considering the non-trivial case where $d > 2k$, our main observation is that both $\boldsymbol{B}\boldsymbol{W}$ and $\hat{\boldsymbol{B}}_t\hat{\boldsymbol{W}}_t$ are low-rank matrix with rank upper bounded by k, which indicates that $\operatorname{rank}\left(\hat{\boldsymbol{B}}_t\hat{\boldsymbol{W}}_t - \boldsymbol{B}\boldsymbol{W}\right) \leq 2k$. Therefore, we can write $\hat{\boldsymbol{B}}_t\hat{\boldsymbol{W}}_t - \boldsymbol{B}\boldsymbol{W} = \boldsymbol{U}_t\boldsymbol{R}_t = [\boldsymbol{U}_t\boldsymbol{r}_{t,1}, \boldsymbol{U}_t\boldsymbol{r}_{t,2}, \cdots, \boldsymbol{U}_t\boldsymbol{r}_{t,M}]$, where $\boldsymbol{U}_t \in \mathbb{R}^{d\times 2k}$ is an orthonormal matrix and $\boldsymbol{R}_t \in \mathbb{R}^{2k\times M}$. Thus we have

$$\boldsymbol{X}_{t-1,i}^\top\left(\hat{\boldsymbol{B}}_t\hat{\boldsymbol{w}}_{t,i} - \boldsymbol{B}\boldsymbol{w}_i\right) = \left(\boldsymbol{U}_t^\top\boldsymbol{X}_{t-1,i}\right)^\top \boldsymbol{R}_t.$$

This observation indicates that we can project the history actions $\boldsymbol{X}_{t-1,i}$ to a $2k$-dimensional space with $\boldsymbol{U}_t$, and take $\boldsymbol{U}_t^\top\boldsymbol{X}_{t-1,i}$ as the $2k$-dimensional actions we have selected in the first $t-1$ steps. Following this idea, we connect the approximation error $\sum_{i=1}^{M}\left\|\hat{\boldsymbol{B}}_t\hat{\boldsymbol{w}}_{t,i} - \boldsymbol{B}\boldsymbol{w}_i\right\|^2_{\tilde{\boldsymbol{V}}_{t-1,i}(\lambda)}$ to the term $\sum_{i=1}^{M}\left\|\boldsymbol{\eta}_{t-1,i}^\top\left(\boldsymbol{U}_t^\top\boldsymbol{X}_{t-1,i}\right)^\top\right\|^2_{\boldsymbol{V}^{-1}_{t-1,i}(\lambda)}$, where $\boldsymbol{V}_{t-1,i}(\lambda) \overset{\text{def}}{=} \left(\boldsymbol{U}_t^\top\boldsymbol{X}_{t-1,i}\right)\left(\boldsymbol{U}_t^\top\boldsymbol{X}_{t-1,i}\right)^\top + \lambda\boldsymbol{I}$. We bound this term for the fixed $\boldsymbol{U}_t$ with the technique of self-normalized bound for vector-valued martingales (Abbasi-Yadkori et al., 2011), and then apply the ϵ-net trick to cover all possible $\boldsymbol{U}_t$. This leads to an upper bound for $\sum_{i=1}^{M}\left\|\boldsymbol{\eta}_{t-1,i}^\top\boldsymbol{X}_{t-1,i}^\top\boldsymbol{U}_t\right\|^2_{\boldsymbol{V}^{-1}_{t-1,i}(\lambda)}$, and consequently helps to obtain the upper bound in Lemma 1.

4.2. Algorithm and Regret

Algorithm 1 Multi-Task Low-Rank OFUL

```
for step $t = 1, 2, \cdots, T$ do
    Calculate the confidence interval $\mathcal{C}_t$ by Eqn 11
    $\tilde{\boldsymbol{\Theta}}_t, \boldsymbol{x}_{t,i} = \operatorname{argmax}_{\boldsymbol{\Theta}\in\mathcal{C}_t, \boldsymbol{x}_i\in\mathcal{A}_{t,i}} \sum_{i=1}^{M} \langle \boldsymbol{x}_i, \boldsymbol{\theta}_i \rangle$
    for task $i = 1, 2, \cdots, M$ do
        Play $\boldsymbol{x}_{t,i}$ for task $i$, and obtain the reward $y_{t,i}$
```

We describe our Multi-Task Low-Rank OFUL algorithm in Algorithm 1. The following theorem states a bound on the regret of the algorithm.

Theorem 1. *Under Assumption 1 and Assumption 2, with probability at least $1 - \delta$, the regret of Algorithm 1 is bounded by*

$$\operatorname{Reg}(T) = \tilde{O}\left(M\sqrt{dkT} + d\sqrt{kMT}\right) \tag{13}$$

We defer the proof of Theorem 1 to Appendix A.2. The first term in the regret has linear dependence on M. This term characterizes the regret caused by learning the parameters $\boldsymbol{w}_i$ for each task. The second term has square root dependence on the number of total samples MT, which indicates the cost to learn the common representation with samples from M tasks. By dividing the total regret by the number of tasks M, we know that the average regret for each task is $\tilde{O}(\sqrt{dkT} + d\sqrt{kT/M})$. Note that if we solve M tasks with algorithms such as OFUL (Abbasi-Yadkori et al., 2011) independently, the regret per task can be $\tilde{O}(d\sqrt{T})$. Our bound saves a factor of $\sqrt{d/k}$ compared with the naive method by leveraging the common representation features. We also show that when $d > M$ our regret bound is near optimal (see Theorem 3).

4.3. Misspecified Multi-Task Linear Bandits

For multi-task linear bandits problem, it is relatively unrealistic to assume a common feature extractor that can fit the reward functions of M tasks exactly. A more natural situation is that the underlying reward functions are not exactly linear, but have some misspecifications. There are also relevant discussions on single-task linear bandits in recent works (Lattimore et al., 2020; Zanette et al., 2020a). We first present a definition for the approximately linear bandits learning in multi-task setting.

Assumption 4. *There exists a linear feature extractor $\boldsymbol{B} \in \mathbb{R}^{d\times k}$ and a set of linear coefficients $\{\boldsymbol{w}_i\}_{i=1}^M$ such that the expectated reward $\mathbb{E}[y_i|\boldsymbol{x}_i]$ for any action $\boldsymbol{x}_i \in \mathbb{R}^d$ satisfies $|\mathbb{E}[y_i|\boldsymbol{x}_i] - \langle \boldsymbol{x}_i, \boldsymbol{B}\boldsymbol{w}_i\rangle| \le \zeta$.*

In general, an algorithm designed for a linear model could break down entirely if the underlying model is not linear. However, we find that our algorithm is in fact robust to small model misspecification if we set $L = \tilde{O}(Mk+kd+MT\zeta^2)$ (see Appendix A.4 for the exact value). The following regret bound holds under Assumption 4 if we slightly modify the hyperparameter L in the definition of confidence region $\mathcal{C}_t$.

Theorem 2. *Under Assumption 1, 2 and 4, with probability at least $1-\delta$, the regret of Algorithm 1 is bounded by*

$$\mathrm{Reg}(T) = \tilde{O}\left(M\sqrt{dkT} + d\sqrt{kMT} + MT\sqrt{d}\zeta\right) \quad (14)$$

Theorem 2 is proved in Appendix A.4. Compared with Theorem 1, there is an additional term $\tilde{O}(MT\sqrt{d}\zeta)$ in the regret of Theorem 2. This additional term is inevitably linear in MT due to the intrinsic bias introduced by linear function approximation. Note that our algorithm can still enjoy good theoretical guarantees when ζ is sufficiently small.

4.4. Lower Bound

In this subsection, we propose the regret lower bound for multi-task linear bandit problem under Assumption 4.

Theorem 3. *For any $k, M, d, T \in \mathbb{Z}^+$ with $k \le d \le T$ and $k \le M$, and any learning algorithm $\mathcal{A}$, there exist a multi-task linear bandit instance that satisfies Assumption 4, such that the regret of Algorithm $\mathcal{A}$ is lower bounded by*

$$\mathrm{Reg}(T) \ge \Omega\left(Mk\sqrt{T} + d\sqrt{kMT} + MT\sqrt{d}\zeta\right).$$

We defer the proof of Theorem 3 to Appendix A.5. By setting $\zeta = 0$, Theorem 3 can be converted to the lower bound for multi-task linear bandit problem under Assumption 1, which is $\Omega(Mk\sqrt{T}+d\sqrt{kMT})$. These lower bounds match the upper bounds in Theorem 1 and Theorem 2 in the regime where $d > M$ respectively. There is still a gap of $\sqrt{d/k}$ in the first part of the regret. For the upper bounds, the main difficulty to obtain $\tilde{O}(Mk\sqrt{T})$ regret in the first part comes from the estimation of $\boldsymbol{B}$. Since the action sets are not fixed and can be ill-conditioned, we cannot follow the explore-then-exploit framework and estimate $\boldsymbol{B}$ at the beginning. Besides, explore-then-exploit algorithms always suffer $\tilde{O}(T^{2/3})$ regret in the general linear bandits setting without further assumptions. Without estimating $\boldsymbol{B}$ beforehand with enough accuracy, the exploration in original d-dimensional space can be difficult since we cannot identify actions that have the similar k-dimensional representations before pulling them. We conjecture that our upper bound is tight and leave the gap as future work.

5. Main Results for Linear RL

We now show the main results for the multi-task episodic reinforcement learning under the assumption of low inherent Bellman error (i.e. the multi-task LSVI setting).

5.1. Multi-task LSVI Framework

In the exploration problems in RL where linear value function approximation is employed (Yang & Wang, 2019; Jin et al., 2020; Yang & Wang, 2020), LSVI-based algorithms are usually very effective when the linear value function space are *close* under Bellman operator. For example, it is shown that a LSVI-based algorithm with additional bonus can solve the exploration challenge effectively in low-rank MDP (Jin et al., 2020), where the function space $\mathcal{Q}_h, \mathcal{Q}_{h+1}$ are exactly close under Bellman operator (i.e. any function Q_{h+1} in $\mathcal{Q}_{h+1}$ composed with Bellman operator $\mathcal{T}_h Q_{h+1}$ belongs to $\mathcal{Q}_h$). For the release of such strong assumptions, the inherent Bellman error for a MDP (Definition 1) was proposed to measure how close is the function space under Bellman operator (Zanette et al., 2020a). We extend the definition of IBE to the multi-task LSVI setting (Definition 6), and show that our refined confidence set for the least square estimator can be applied to the low-rank multi-task LSVI setting, and gives an optimism-based algorithm with sharper regret bound compared to naively do exploration in each task independently.

5.2. Algorithm

The MTLR-LSVI (Algorithm 2) follows the LSVI-based (Jin et al., 2020; Zanette et al., 2020a) algorithms to build our (optimistic) estimator for the optimal value functions. To understand how this works for multi-task LSVI setting, we first take a glance at how LSVI-based algorithms work in single-task LSVI setting.

In traditional value iteration algorithms, we perform an approximate Bellman backup in episode t for each step $h \in [H]$ on the estimator $Q_{h+1,t-1}$ constructed at the end of episode $t-1$, and find the best approximator for

$\mathcal{T}_h\left(Q_{h+1,t-1}\right)$ in function space $\mathcal{Q}_h$. An effective and widely-used approximator is the least-square solution of the empirical Bellman backup on $Q_{h+1,t-1}$.

In the multi-task framework, suppose we have obtained the estimator $Q_{h+1}\left(\boldsymbol{\theta}_{h+1}^i\right)$ for each $i \in [M]$. To apply such least-square value iteration to our low-rank multi-task LSVI setting, we use the solution to the following constrained optimization problem

$$\sum_{i=1}^{M}\sum_{j=1}^{t-1}\left(\left(\boldsymbol{\phi}_{hj}^i\right)^\top \boldsymbol{\theta}_h^i - R_{hj}^i - V_{h+1}^i\left(\boldsymbol{\theta}_{h+1}^i\right)\left(s_{h+1,j}^i\right)\right)^2 \tag{15}$$

$$\text{s.t.} \quad \boldsymbol{\theta}_h^1, \boldsymbol{\theta}_h^2, ..., \boldsymbol{\theta}_h^M \text{ lies in a } k\text{-dimensional subspace} \tag{16}$$

to approximate the Bellman update in the t-th episode, where $\boldsymbol{\phi}_{hj}^i = \boldsymbol{\phi}_h(s_{hj}^i, a_{hj}^i)$ is the feature observed at step h in episode j for task i, and similarly $R_{hj}^i = R_h(s_{hj}^i, a_{hj}^i)$.

To guarantee the optimistic property of our estimator, we follow the global optimization procedure of Zanette et al. (2020a) which solves the following optimization problem in the t-th episode:

Definition 1 (Global Optimization Procedure)**.**

$$\max_{\bar{\boldsymbol{\xi}}_h^i, \hat{\boldsymbol{\theta}}_h^i, \bar{\boldsymbol{\theta}}_h^i} \sum_{i=1}^{M} \max_{a^i}\left(\boldsymbol{\phi}(s_1^i, a^i)\right)^\top \bar{\boldsymbol{\theta}}_1^i \tag{17}$$

$$\begin{aligned} s.t. \quad \left(\hat{\boldsymbol{\theta}}_h^1, ..., \hat{\boldsymbol{\theta}}_h^M\right) &= \hat{\boldsymbol{B}}_h \begin{bmatrix}\hat{\boldsymbol{w}}_h^1 & \hat{\boldsymbol{w}}_h^2 & \cdots & \hat{\boldsymbol{w}}_h^M\end{bmatrix} \\ &= \operatorname*{argmin}_{\left\|\boldsymbol{B}_h\boldsymbol{w}_h^i\right\|_2 \le D} \sum_{i=1}^{M}\sum_{j=1}^{t-1} L_j(\boldsymbol{B}_h, \boldsymbol{w}_h^i) \end{aligned} \tag{18}$$

$$\bar{\boldsymbol{\theta}}_h^i = \hat{\boldsymbol{\theta}}_h^i + \bar{\boldsymbol{\xi}}_h^i; \quad \sum_{i=1}^{M}\left\|\bar{\boldsymbol{\xi}}_h^i\right\|_{\tilde{\boldsymbol{V}}_{ht}^i(\lambda)}^2 \le \alpha_{ht} \tag{19}$$

$$\left(\bar{\boldsymbol{\theta}}_h^1, \bar{\boldsymbol{\theta}}_h^2, \cdots, \bar{\boldsymbol{\theta}}_h^M\right) \in \Theta_h \tag{20}$$

where the empirical least-square loss $L_j(\boldsymbol{B}_h, \boldsymbol{w}_h^i) \overset{\text{def}}{=} ((\boldsymbol{\phi}_{hj}^i)^\top \boldsymbol{B}_h \boldsymbol{w}_h^i - R_{hj}^i - V_{h+1}^i(\bar{\boldsymbol{\theta}}_{h+1}^i)(s_{h+1,j}^i))^2$, and $\tilde{\boldsymbol{V}}_{ht}^i(\lambda) \overset{\text{def}}{=} \sum_{j=1}^{t-1}(\boldsymbol{\phi}_{hj}^i)(\boldsymbol{\phi}_{hj}^i)^\top + \lambda\boldsymbol{I}$ is the regularized empirical linear design matrix for task i in episode t.

We have three types of variables in this global optimization problem, $\bar{\boldsymbol{\xi}}_h^i, \hat{\boldsymbol{\theta}}_h^i$, and $\bar{\boldsymbol{\theta}}_h^i$. Here $\bar{\boldsymbol{\theta}}_h^i$ denotes the estimator for Q_h^{i*}. We solve for the low-rank least-square solution of the approximate value iteration and denote the solution by $\hat{\boldsymbol{\theta}}_h^i$. Instead of adding the bonus term directly on $Q_h^i(\hat{\boldsymbol{\theta}}_h^i)$ to obtain an optimistic estimate of Q_h^{i*} as in the tabular setting (Azar et al., 2017; Jin et al., 2018) and linear MDP setting (Jin et al., 2020), we use global variables $\bar{\boldsymbol{\xi}}_h^i$ to quantify the confidence bonus. This is because we cannot preserve the linear property of our estimator if we add the bonus directly, resulting in an exponential propagation of error. However, by using $\bar{\boldsymbol{\xi}}_h^i$ we can construct a linear estimator $Q_h^i\left(\bar{\boldsymbol{\theta}}_h^i\right)$ and obtain much smaller regret. A drawback of this global optimization technique is that we can only obtain an optimistic estimator at step 1, since values in different states and steps are possibly negatively correlated.

Algorithm 2 Multi-Task Low-Rank LSVI

```
Input: low-rank parameter $k$, failure probability $\delta$, regularization $\lambda = 1$, inherent Bellman error $\mathcal{I}$
Initialize $\tilde{\boldsymbol{V}}_{h1} = \lambda\boldsymbol{I}$ for $h \in [H]$
for episode $t = 1, 2, \cdots$ do
    Compute $\alpha_{ht}$ for $h \in [H]$. (see Lemma 9)
    Solve the global optimization problem 1
    Compute $\pi_{ht}^i(s) = \text{argmax}_a\, \boldsymbol{\phi}(s,a)^\top \bar{\boldsymbol{\theta}}_{ht}^i$
    Execute $\pi_{ht}^i$ for task $i$ at step $h = 1, 2, ..., H$
    Collect $\left\{s_{ht}^i, a_{ht}^i, r\left(s_{ht}^i, a_{ht}^i\right)\right\}$ for episode $t$.
```

5.3. Regret Bound

Theorem 4. *Under Assumption 3, with probability* $1-\delta$ *the regret after* T *episodes is bounded by*

$$\text{Reg}(T) = \tilde{O}\left(HM\sqrt{dkT} + Hd\sqrt{kMT} + HMT\sqrt{d}\mathcal{I}\right) \tag{21}$$

Compared to naively executing single-task linear RL algorithms (e.g. the ELEANOR algorithm) on each task without information-sharing, which incurs regret $\tilde{O}(HMd\sqrt{T} + HMT\sqrt{d}\mathcal{I})$, our regret bound is smaller by a factor of approximately $\sqrt{d/k}$ where $k \ll d$ and $k \ll M$.

We give a brief explanation on how we improve the regret bound and defer the full analysis to appendix B. We start with the decomposition of the regret. Let $\bar{Q}_{ht}^i(\bar{V}_{ht}^i)$ be the solution of the problem in definition 1 in episode t, then

$$\text{Reg}(T) = \sum_{t=1}^{T}\sum_{i=1}^{M}\left(V_1^{i*} - \bar{V}_{1t}^i + \bar{V}_{1t}^i - V_1^{\pi_t^i}\right)\left(s_{1t}^i\right) \tag{22}$$

$$\le HMT\mathcal{I} \qquad \text{(by Lemma 12)} \tag{23}$$

$$+\sum_{t=1}^{T}\sum_{h=1}^{H}\sum_{i=1}^{M}\left(\left|\bar{Q}_{ht}^i(s,a) - \mathcal{T}_h^i\bar{Q}_{h+1,t}^i(s,a)\right| + \zeta_{ht}^i\right). \tag{24}$$

In (23) we use the optimistic property of $\bar{V}_{1t}^i$. In (24), ζ_{ht}^i is a martingale difference (defined in section B.5) with regards to $\mathcal{F}_{h,t}$, and the dominate term (the first term) is the Bellman error of $\bar{Q}_{ht}^i$.

For any $\{Q_{h+1}^i\}_{i=1}^M \in \mathcal{Q}_{h+1}$, we can find a group of vectors $\{\dot{\boldsymbol{\theta}}_h^i(Q_{h+1}^i)\}_{i=1}^M \in \Theta_h$ that satisfy $\Delta_h^i\left(Q_{h+1}^i\right)(s,a) \overset{\text{def}}{=}$

$\mathcal{T}_h^i\left(Q_{h+1}^i\right)(s,a) - \boldsymbol{\phi}(s,a)^\top \dot{\boldsymbol{\theta}}_h^i\left(Q_{h+1}^i\right)$ and the approximation error $\left\|\Delta_h^i\left(Q_{h+1}^i\right)\right\|_\infty \le \mathcal{I}$ for each $i \in [M]$. By definition, $\dot{\boldsymbol{\theta}}_h^i\left(Q_{h+1}^i\right)$ is actually the best approximator of $\mathcal{T}_h^i\left(Q_{h+1}^i\right)$ in the function class $\mathcal{Q}_h$. Since our algorithm is based on least-square value iteration, a key step is to bound the error of estimating $\dot{\boldsymbol{\theta}}_h^i(\bar{Q}_{h+1,t}^i)$ ($\dot{\boldsymbol{\theta}}_h^i$ for short). In the global optimization procedure, we use $\hat{\boldsymbol{\theta}}_h^i$ to approximate the empirical Bellman backup. In Lemma 9 we show

$$\sum_{i=1}^{M}\left\|\hat{\boldsymbol{\theta}}_h^i - \dot{\boldsymbol{\theta}}_h^i\right\|_{\tilde{\mathbf{V}}_{ht}^i(\lambda)}^2 = \tilde{O}\left(Mk + kd + MT\mathcal{I}^2\right) \tag{25}$$

This is the key step leading to improved regret bound. If we solve each task independently without information sharing, we can only bound the least square error in (25) as $\tilde{O}(Md + MT\mathcal{I}^2)$. Our bound is much more sharper since $k \ll d$ and $k \ll M$.

Using the least square error in (25), we can show that the dominate term in (24) is bounded by (see Lemma 10 and section B.5)

$$\sum_{i=1}^{M}\left|\bar{Q}_{ht}^i(s,a) - \mathcal{T}_h^i\bar{Q}_{h+1,t}^i(s,a)\right| \le M\mathcal{I} + \tag{26}$$

$$\tilde{O}\left(\sqrt{Mk + kd + MT\mathcal{I}^2}\right) \cdot \sqrt{\sum_{i=1}^{M}\left\|\boldsymbol{\phi}(s_{ht}^i, a_{ht}^i)\right\|_{\tilde{\mathbf{V}}_{ht}^i(\lambda)^{-1}}^2} \tag{27}$$

Abbasi-Yadkori et al. (2011, Lemma 11) states that $\sum_{t=1}^{T}\left\|\boldsymbol{\phi}(s_{ht}^i, a_{ht}^i)\right\|_{\tilde{\mathbf{V}}_{ht}^i(\lambda)^{-1}}^2 = \tilde{O}(d)$ for any h and i, so we can finally bound the regret as

$$\begin{aligned}\operatorname{Reg}(T) &= \tilde{O}\left(HMT\mathcal{I} + H\sqrt{Mk + kd + MT\mathcal{I}^2} \cdot \sqrt{MTd}\right)\\ &= \tilde{O}\left(HM\sqrt{dkT} + Hd\sqrt{kMT} + HMT\sqrt{d}\mathcal{I}\right)\end{aligned}$$

where the first equality is by Cauchy-Schwarz.

5.4. Lower Bound

This subsection presents the lower bound for multi-task reinforcement learning with low inherent Bellman error. Our lower bound is derived from the lower bound in the single-task setting. As a byproduct, we also derive a lower bound for misspecified linear RL in the single-task setting. We defer the proof of Theorem 5 to Appendix C.

Theorem 5. *For our construction in appendix C, the expected regret of any algorithm where $d, k, H \ge 10, |\mathcal{A}| \ge 3, M \ge k, T = \Omega(d^2 H), \mathcal{I} \le 1/4H$ is*

$$\Omega\left(Mk\sqrt{HT} + d\sqrt{HkMT} + HMT\sqrt{d}\mathcal{I}\right)$$

Careful readers may find that there is a gap of $\sqrt{H}$ in the first two terms between the upper bound and the lower bound. This gap is because the confidence set used in the algorithm is intrinsically "Hoeffding-type". Using a "Bernstein-type" confidence set can potentially improve the upper bound by a factor of $\sqrt{H}$. This "Bernstein" technique has been well exploited in many previous results for single-task RL (Azar et al., 2017; Jin et al., 2018; Zhou et al., 2020a). Since our focus is mainly on the benefits of multi-task representation learning, we don't apply this technique for the clarity of the analysis. If we ignore this gap in the dependence on H, our upper bound matches this lower bound in the regime where $d \ge M$.

6. Conclusion

In this paper, we study provably sample-efficient representation learning for multi-task linear bandits and linear RL. For linear bandits, we propose an algorithm called MTLR-OFUL, which obtains near-optimal regret in the regime where $d \ge M$. We then extend our algorithms to multi-task RL setting, and propose a sample-efficient algorithm, MTLR-LSVI.

There are two directions for future investigation. First, our algorithms are statistically sample-efficient, but a computationally efficient implementation is still unknown, although we conjecture our MTLR-OFUL algorithm is computationally efficient. How to design both computationally and statistically efficient algorithms in our multi-task setting is an interesting problem for future research. Second, there remains a gap of $\sqrt{d/k}$ between regret upper and lower bounds (in the first term). We conjecture that our lower bound is not minimax optimal and hope to address this problem in the future work.

Acknowledgements

This work was supported by National Key R&D Program of China (2018YFB1402600), Key-Area Research and Development Program of Guangdong Province (No. 2019B121204008), BJNSF (L172037) and Beijing Academy of Artificial Intelligence.

References

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pp. 2312–2320, 2011.

Abbasi-Yadkori, Y., Pal, D., and Szepesvari, C. Online-to-confidence-set conversions and application to sparse stochastic bandits. In *Artificial Intelligence and Statistics*, pp. 1–9. PMLR, 2012.

Ando, R. K. and Zhang, T. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(Nov):1817–1853, 2005.

Arora, S., Du, S. S., Kakade, S., Luo, Y., and Saunshi, N. Provable representation learning for imitation learning via bi-level optimization. *arXiv preprint arXiv:2002.10544*, 2020.

Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

Ayoub, A., Jia, Z., Szepesvari, C., Wang, M., and Yang, L. F. Model-based reinforcement learning with value-targeted regression. *arXiv preprint arXiv:2006.01107*, 2020.

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pp. 263–272. PMLR, 2017.

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

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8): 1798–1828, 2013.

Brunskill, E. and Li, L. Sample complexity of multi-task reinforcement learning. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence (UAI-13)*, pp. 122–131, 2013.

Carpentier, A. and Munos, R. Bandit theory meets compressed sensing for high dimensional stochastic linear bandit. In *Artificial Intelligence and Statistics*, pp. 190–198. PMLR, 2012.

Caruana, R. Multitask learning. *Machine learning*, 28(1): 41–75, 1997.

Chu, W., Li, L., Reyzin, L., and Schapire, R. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 208–214, 2011.

Dani, V., Hayes, T. P., and Kakade, S. M. Stochastic linear optimization under bandit feedback. 2008.

D'Eramo, C., Tateo, D., Bonarini, A., Restelli, M., and Peters, J. Sharing knowledge in multi-task deep reinforcement learning. In *International Conference on Learning Representations*, 2019.

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

Hao, B., Lattimore, T., and Wang, M. High-dimensional sparse linear bandits. *arXiv preprint arXiv:2011.04020*, 2020.

Hessel, M., Soyer, H., Espeholt, L., Czarnecki, W., Schmitt, S., and van Hasselt, H. Multi-task deep reinforcement learning with popart. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3796–3803, 2019.

Jiang, N., Krishnamurthy, A., Agarwal, A., Langford, J., and Schapire, R. E. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pp. 1704–1713. PMLR, 2017.

Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. Is q-learning provably efficient? *arXiv preprint arXiv:1807.03765*, 2018.

Jin, C., Yang, Z., Wang, Z., and Jordan, M. I. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pp. 2137–2143. PMLR, 2020.

Jun, K.-S., Willett, R., Wright, S., and Nowak, R. Bilinear bandits with low-rank structure. In *International Conference on Machine Learning*, pp. 3163–3172. PMLR, 2019.

Lattimore, T. and Szepesvári, C. *Bandit algorithms*. Cambridge University Press, 2020.

Lattimore, T., Crammer, K., and Szepesvári, C. Linear multi-resource allocation with semi-bandit feedback. In *NIPS*, pp. 964–972, 2015.

Lattimore, T., Szepesvari, C., and Weisz, G. Learning with good feature representations in bandits and in rl with a generative model. In *International Conference on Machine Learning*, pp. 5662–5670. PMLR, 2020.

Li, J., Zhang, H., Zhang, L., Huang, X., and Zhang, L. Joint collaborative representation with multitask learning for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 52(9):5923–5936, 2014.

Li, Y., Wang, Y., and Zhou, Y. Nearly minimax-optimal regret for linearly parameterized bandits. *arXiv preprint arXiv:1904.00242*, 2019a.

Li, Y., Wang, Y., and Zhou, Y. Tight regret bounds for infinite-armed linear contextual bandits. *arXiv preprint arXiv:1905.01435*, 2019b.

Liu, L. T., Dogan, U., and Hofmann, K. Decoding multitask dqn in the world of minecraft. In *The 13th European Workshop on Reinforcement Learning (EWRL) 2016*, 2016.

Liu, X., He, P., Chen, W., and Gao, J. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*, 2019.

Lu, Y., Meisami, A., and Tewari, A. Low-rank generalized linear bandit problems. *arXiv preprint arXiv:2006.02948*, 2020.

Maurer, A., Pontil, M., and Romera-Paredes, B. The benefit of multitask representation learning. *The Journal of Machine Learning Research*, 17(1):2853–2884, 2016.

Parisotto, E., Ba, J. L., and Salakhutdinov, R. Actor-mimic: Deep multitask and transfer reinforcement learning. *arXiv preprint arXiv:1511.06342*, 2015.

Ramsundar, B., Kearnes, S., Riley, P., Webster, D., Konerding, D., and Pande, V. Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072*, 2015.

Rusmevichientong, P. and Tsitsiklis, J. N. Linearly parameterized bandits. *Mathematics of Operations Research*, 35 (2):395–411, 2010.

Taylor, M. E. and Stone, P. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(7), 2009.

Teh, Y., Bapst, V., Czarnecki, W. M., Quan, J., Kirkpatrick, J., Hadsell, R., Heess, N., and Pascanu, R. Distral: Robust multitask reinforcement learning. In *Advances in Neural Information Processing Systems*, pp. 4496–4506, 2017.

Thrun, S. and Pratt, L. Learning to learn: Introduction and overview. In *Learning to learn*, pp. 3–17. Springer, 1998.

Tripuraneni, N., Jin, C., and Jordan, M. I. Provable meta-learning of linear representations. *arXiv preprint arXiv:2002.11684*, 2020.

Wang, R., Du, S. S., Yang, L. F., and Kakade, S. M. Is long horizon reinforcement learning more difficult than short horizon reinforcement learning? *arXiv preprint arXiv:2005.00527*, 2020.

Wilson, A., Fern, A., Ray, S., and Tadepalli, P. Multi-task reinforcement learning: a hierarchical bayesian approach. In *Proceedings of the 24th international conference on Machine learning*, pp. 1015–1022, 2007.

Yang, J., Hu, W., Lee, J. D., and Du, S. S. Provable benefits of representation learning in linear bandits, 2020.

Yang, L. and Wang, M. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International Conference on Machine Learning*, pp. 10746–10756. PMLR, 2020.

Yang, L. F. and Wang, M. Sample-optimal parametric q-learning using linearly additive features. *arXiv preprint arXiv:1902.04779*, 2019.

Zanette, A., Lazaric, A., Kochenderfer, M., and Brunskill, E. Learning near optimal policies with low inherent bellman error. *arXiv preprint arXiv:2003.00153*, 2020a.

Zanette, A., Lazaric, A., Kochenderfer, M. J., and Brunskill, E. Provably efficient reward-agnostic navigation with linear value iteration. *arXiv preprint arXiv:2008.07737*, 2020b.

Zhang, Z., Ji, X., and Du, S. S. Is reinforcement learning more difficult than bandits? a near-optimal algorithm escaping the curse of horizon. *arXiv preprint arXiv:2009.13503*, 2020.

Zhou, D., Gu, Q., and Szepesvari, C. Nearly minimax optimal reinforcement learning for linear mixture markov decision processes. *arXiv preprint arXiv:2012.08507*, 2020a.

Zhou, D., He, J., and Gu, Q. Provably efficient reinforcement learning for discounted mdps with feature mapping. *arXiv preprint arXiv:2006.13165*, 2020b.