Linear Stochastic Bandits over a Bit-Constrained Channel

Aritra Mitra AMITRA2@NCSU.EDU
Department of Electrical and Computer Engineering, North Carolina State University

Hamed Hassani HASSANI@SEAS.UPENN.EDU
Department of Electrical and Systems Engineering, University of Pennsylvania

George J. Pappas PAPPASG@SEAS.UPENN.EDU
Department of Electrical and Systems Engineering, University of Pennsylvania

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

One of the primary challenges in large-scale distributed learning stems from stringent communication constraints. While several recent works address this challenge for static optimization problems, sequential decision-making under uncertainty has remained much less explored in this regard. Motivated by this gap, we introduce a new linear stochastic bandit formulation over a bit-constrained channel. Specifically, in our setup, an agent interacting with an environment transmits encoded estimates of an unknown model parameter to a server over a communication channel of finite capacity. The goal of the server is to take actions based on these estimates to minimize cumulative regret. To this end, we develop a novel and general algorithmic framework that hinges on two main components: (i) an adaptive encoding mechanism that exploits statistical concentration bounds, and (ii) a decision-making principle based on confidence sets that account for encoding errors. As our main result, we prove that when the unknown model is d-dimensional, a channel capacity of $O(d)$ bits suffices to achieve order-optimal regret. We also establish that for the simpler unstructured multi-armed bandit problem, 1 bit channel capacity is sufficient for achieving optimal regret bounds.

Keywords: Linear Bandits, Distributed Learning, Communication Constraints

1. Introduction

In modern distributed computing paradigms such as federated learning (FL), a group of agents typically interact with a parameter server to train a common statistical model. A major bottleneck in such settings is the network communication cost of uploading (potentially high-dimensional) models and gradient vectors to the server. Motivated by this concern, several works draw on ideas from quantization theory (Seide et al., 2014; Alistarh et al., 2017), sparsification (Wen et al., 2017; Stich et al., 2018), and rate-distortion theory (Mitchell et al., 2022) to design communication-efficient algorithms that achieve a desired level of precision while exchanging as few bits as possible. Although this rich body of work contributes significantly to the study of static optimization problems under communication constraints, there remains a considerable gap in our understanding of similar questions when it comes to sequential decision-making under uncertainty (e.g., bandit problems and reinforcement learning). Our main goal in this paper is to bridge the above gap.

A common abstraction for analyzing optimization under limited communication is one where a worker agent transmits quantized gradients to a server over a finite bit-rate communication channel (Mayekar and Tyagi, 2020; Gandikota et al., 2021; Lin et al., 2021). Inspired by this model, for our

problem of interest, we introduce and study a new linear stochastic bandit formulation comprising of an agent connected to a decision-making entity (server) by a noiseless communication channel of finite capacity B; see Fig. 1. The agent interacts with an environment and observes noisy rewards that depend linearly on an unknown parameter vector $\theta_* \in \mathbb{R}^d$. It then encodes and transmits finite-precision estimates of θ_* to the server. Based on these estimates, the role of the server is to play a sequence of actions that maximizes the sum of rewards accrued over a time horizon T - a performance metric captured by cumulative regret.

Notably, the agent can only transmit encoded estimates of the parameter vector, but not the rewards. The reason for this is twofold. First, our setting is motivated by the popular federated learning (FL) framework (Konečnỳ et al., 2016) where due to privacy concerns, agents exchange their local models with the server instead of their raw observations. Indeed, our communication model is consistent with recent works on federated linear bandits (Huang et al., 2021) where agents are not allowed to exchange private rewards (observations), and instead only exchange model (parameter) estimates.[1] Second, beyond FL, our goal is to eventually consider challenging multi-agent reinforcement learning (RL) problems where the server may not be aware of the agents' actions. In such settings, consistent with collaborative filtering/estimation (Olfati-Saber, 2005; Speranzon et al., 2006) techniques, the agents would need to exchange and fuse parameter/model estimates (as in our work) via the server to benefit from collaboration. In this regard, we note that in existing works on multi-agent and federated RL (Doan et al., 2019; Qi et al., 2021; Jin et al., 2022; Khodadadian et al., 2022), agents exchange models (parameters), keeping their personal data (i.e., rewards, states, and actions) private. Our communication model thus aligns with these works as well.

The main technical challenge in our setup arises from the fact that the channel from the agent to the server introduces *additional uncertainty* into the decision-making process. Unless accounted for carefully, the instantaneous encoding errors resulting from such uncertainty can accumulate over time and lead to sub-optimal regret bounds. Given this challenge, the central question we investigate is: *Under what conditions on the channel capacity B can we achieve the order-optimal regret bound $\tilde{O}(d\sqrt{T})$?*[2] In this work, we rigorously answer the above question via the following contributions.

• **Algorithmic Contributions.** For the setting of interest, we develop a novel framework for statistical decision-making under communication constraints. Our approach hinges on two main components. The first is an adaptive quantization mechanism that encodes the change (*innovation*) in successive estimates of θ_* at the agent. The main intuition here is that with high probability, the gap between successive model estimates shrinks over time; as a result, the innovation signals are contained in balls of progressively smaller radii. Thus, roughly speaking, to achieve the same precision, it takes fewer bits to encode the innovation signals as compared to the model estimates (that can be of a much larger magnitude). A key feature of our encoding scheme is that the dynamic quantizer ranges are designed based on statistical concentration bounds specific to the stochastic process we study. As such, our encoding scheme is novel, and differs significantly from standard quantization approaches for optimization. The second key component of our framework is the decision-making policy at the server that comprises of two phases: (i) a pure exploration phase that facilitates estimation of θ_*, and (ii) an information-constrained exploration-exploitation phase. In the latter phase, actions are taken based on certain "inflated" confidence sets that are carefully constructed to account for the errors induced by compression. The construction of such sets is an important

1. We note, however, that providing formal privacy guarantees is not the main focus of our work. Instead, much like the initial papers on federated optimization (Konečnỳ et al., 2016), our focus is on *communication-efficiency*.
2. Under infinite channel capacity, i.e., when $B = \infty$, $\tilde{O}(d\sqrt{T})$ regret is optimal (Lattimore and Szepesvári, 2020).

algorithmic contribution of our work. We refer to our overall scheme as the Information-Constrained LinUCB algorithm (IC−LinUCB).

• **Theoretical Contributions.** Our first main result (Theorem 2) reveals that with a channel capacity $B = O(d)$ bits, IC−LinUCB guarantees a regret bound of $\tilde{O}(d\sqrt{T})$. The main implication of this result is that one can achieve *minimax-optimal* regret guarantees with a bit-rate that is *independent of the horizon* T, and that depends only on the dimension d of the unknown model θ_*. This result is particularly appealing for infinite-horizon stochastic control problems. As far as we are aware, this is the first result of its kind for linear stochastic bandits, and complements similar results for stochastic optimization: Mayekar and Tyagi (2020) recently showed that with d-dimensional quantized gradients, a bit-rate of $\tilde{O}(d)$ bits is sufficient for achieving the optimal optimization convergence rate. On the technical front, we note that the proof of Theorem 2 is non-trivial, and relies on some key intermediate ideas that we outline in Section 3. We also ask: *When the action sets have additional structure, can we exploit such structure to achieve optimal performance with fewer than* $O(d)$ *bits?* To answer this question, we study a special case of the linear bandit problem where the actions are the standard basis vectors: the multi-armed bandit (MAB) problem with a finite number of arms (Auer et al., 2002). For this setting, we prove that with a bit-rate $B = 1$, one can achieve both gap-dependent (Theorem 5) and gap-independent (Theorem 6) regret bounds matching those of the celebrated upper-confidence bound (UCB) algorithm.

Overall, we envision that the insights from this work will pave the way for studying more complex multi-agent statistical decision-making problems under communication constraints.

Further Related Work. Our formulation is inspired by the classical work (Tatikonda and Mitter, 2004a) that studies the problem of stabilizing a linear time-invariant dynamical system over a bit-constrained channel. There, as in our setup, the estimation module (sensor) is separated from the decision-making module (controller) by the channel. Aside from the fact that we study a fundamentally different problem, our work departs from (Tatikonda and Mitter, 2004a) in that our setup is inherently stochastic, while the authors in (Tatikonda and Mitter, 2004a) consider a fully deterministic setting. Throughout the paper, to isolate the challenges unique to our problem, we consider a noiseless channel. We note here that several works at the intersection of control and information theory have studied coding schemes for various control tasks over noisy channels (Tatikonda and Mitter, 2004b; Matveev, 2008; Ostrovsky et al., 2009; Sukhavasi and Hassibi, 2016; Khina et al., 2019; Gatsis et al., 2020). We anticipate that ideas from these papers can be combined with the adaptive quantization mechanism that we develop. Crucially, our work departs from all the aforementioned papers in that the system model is assumed to be *known* in such papers. In contrast, a key feature of our setting is that the model θ_* is an unknown high-dimensional object; this *statistical uncertainty* is precisely what contributes to the learning component in our problem.

Our work is also naturally related to the seminal papers on linear stochastic bandits (Dani et al., 2008; Abbasi-Yadkori et al., 2011). In the context of multi-agent bandits (Landgren et al., 2016; Shahrampour et al., 2017; Sankararaman et al., 2019), a body of work focuses on achieving benefits of collaboration while minimizing the number of communication rounds (Wang et al., 2019; Dubey and Pentland, 2020; Chawla et al., 2020; Agarwal et al., 2021). The main goal of these papers is to achieve desirable performance while minimizing the *frequency* of communication. Our focus is orthogonal - that of studying the impact of finite-precision communication channels on the performance of bandit algorithms. As a result, our problem formulation, algorithmic techniques, and theoretical results differ from the above strand of literature.

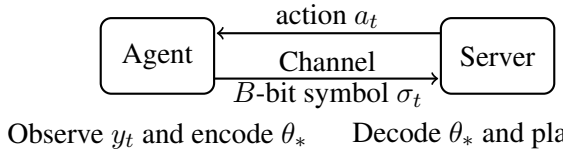

Figure 1: At each round t, the action a_t played by the server is sent to the agent without any loss of information. The agent then observes a reward y_t as per Eq. (1), encodes an estimate of the model θ_*, and transmits the encoded symbol σ_t back to the server under the B-bits per round channel constraint. The server performs decoding and plays the next action a_{t+1}.

Notation: We use $\mathcal{B}_d(0,1)$ and $\mathbb{S}^{d-1}$ to represent the d-dimensional Euclidean ball and the d-dimensional Euclidean sphere, respectively, of unit radius centered at the origin. We use x' to denote the transpose of a vector x.

2. Model and Problem Formulation

We study a setting comprising of an agent and a decision-maker (server) separated by a noiseless communication channel of finite capacity; see Fig. 1. Based on all the information acquired by the server up to time-step $t-1$, it chooses an action $a_t \in \mathcal{A}_t$ at time t, where $\mathcal{A}_t \subset \mathbb{R}^d$ is the feasible decision set at time t. The agent then receives a reward according to the following model:

$$y_t = \langle \theta_*, a_t \rangle + \eta_t, \tag{1}$$

where $\{\eta_t\}$ is a sequence of i.i.d. 1-subgaussian noise random variables. Here, θ_* is an unknown parameter that belongs to a known compact set $\Theta \subset \mathbb{R}^d$; for each $\theta \in \Theta$, it holds that $\|\theta\|_2 \leq M$, where $M \geq 1$. Our performance measure of interest is the following cumulative regret metric R_T:

$$R_T = \mathbb{E}\left[\sum_{t=1}^{T} \max_{a \in \mathcal{A}_t} \langle \theta_*, a - a_t \rangle\right], \tag{2}$$

where T is the time horizon. Inspired by the setup in (Tatikonda and Mitter, 2004a), the role of the agent in our problem is to perform *sensing* (i.e., collecting rewards) and *estimation* (i.e., maintaining estimates of θ^*). The server, in turn, is responsible for *decision-making*, and seeks to play a sequence of actions such that R_T grows sub-linearly in T. When there is no loss of information from the agent to the server, it is well known that one can achieve $\tilde{O}\left(d\sqrt{T}\right)$ regret via the popular `LinUCB` algorithm (Abbasi-Yadkori et al., 2011). Our **goal** is to develop an algorithm that achieves the same performance subject to communication constraints that we describe next.

Communication constraints. To capture communication constraints, we assume that the channel from the agent to the server has a finite capacity of B bits. Thus, at each time-step, the channel can transmit without error one of 2^B symbols denoted by $\sigma \in \Sigma$, where $|\Sigma| = 2^B$. As explained and motivated in the introduction, we impose an additional information constraint that the agent can only transmit encoded estimates of the unknown model parameter θ_*, but not the rewards themselves. In Section 3, we will establish that with $B = O(d)$ bits, one can ensure that $R_T = \tilde{O}\left(d\sqrt{T}\right)$. Arriving

at this result is however quite non-trivial, and requires overcoming certain key technical challenges that we outline next.

Challenges. In the standard linear stochastic bandit formulation, the chief difficulty lies in taking decisions that incur low regret despite statistical uncertainty concerning the unknown parameter θ_*. In our setting, such uncertainty is accentuated by the loss of information incurred over the finite-capacity channel. Unless the server explicitly accounts for this additional source of error in its decision-making process, it can end up taking sub-optimal actions that generate low rewards. Moreover, since our problem is of an inherently sequential nature, the effect of "poor" actions coupled with channel-induced errors can pile up over time, resulting in the agent-server pair suffering linear regret. The above discussion highlights the challenge in decision-making. In terms of communication, our goal is to ensure that the *channel capacity B exhibits no dependence whatsoever on the horizon T*. This is particularly motivated by the need to accommodate general RL settings that are replete with infinite-horizon stochastic control problems, where $T \to \infty$. For such settings, any dependence of the capacity B on T would imply a prohibitively large communication cost. The above argument essentially rules out certain natural non-adaptive encoding strategies. To sum up, the design of a joint encoding-decoding and decision-making strategy that achieves order-optimal regret with a horizon-independent channel capacity is not at all obvious a priori.

We close this section by outlining some standard assumptions.

Assumption 1 *The following hold: (i)* $\max_{t\in[T]} \sup_{a,b\in\mathcal{A}_t} \langle \theta_*, a-b\rangle \leq 1$*; (ii)* $\|a\|_2 \leq L$, $\forall a \in \bigcup_{t=1}^{T} \mathcal{A}_t$*; and (iii) At each time-step* $t \in [T]$*, the decision set* $\mathcal{A}_t$ *contains the unit sphere* $\mathbb{S}^{d-1}$.

While assumptions (i) and (ii) are typical in the literature on linear stochastic bandits (Lattimore and Szepesvári, 2020), assumption (iii) is also quite standard and has been used in various different contexts (Amani et al., 2019; Yang et al., 2020). Without loss of generality, we assume that $L \geq 1$. Furthermore, we assume that the horizon is long-enough: $T \geq d^2$.

3. Information-Constrained Optimism in the Face of Uncertainty

In this section, we will develop our proposed algorithm (Algorithm 2) called Information Constrained `LinUCB` (`IC-LinUCB`) that comprises of two phases. Phase I is a pure exploration phase where the server picks i.i.d. actions from the uniform distribution over the unit sphere; such actions are feasible owing to Assumption 1-(iii). During this phase which lasts for $\bar{T}+1$ time-steps, the only transmission from the agent to the server takes place at time-step $\bar{T}+1$. The purpose of the pure exploration phase and the choice of the parameter $\bar{T}$ will be explained shortly. During each time-step of Phase II, the agent employs an *adaptive* encoding strategy (outlined in Algorithm 1) to transmit information about the unknown parameter θ_* to the server. Based on this information, the server takes decisions by constructing an "inflated" confidence set that accounts for encoding errors. We now describe in detail the two key ingredients of `IC-LinUCB`: (i) the adaptive encoding strategy at the agent, and (ii) the decision-making rule at the server.

• **Adaptive Encoding at Agent.** To describe the encoder, we will require the notion of an ϵ-net; the following definition is from (Vershynin, 2018).

Definition 1 *Consider a subset $\mathcal{K} \subset \mathbb{R}^d$ and let $\epsilon > 0$. A subset $\mathcal{N} \subseteq \mathcal{K}$ is called an ϵ-net of $\mathcal{K}$ if every point in $\mathcal{K}$ is within a distance of ϵ of some point of $\mathcal{N}$, i.e., $\forall x \in \mathcal{K}, \exists x_0 \in \mathcal{N} : \|x - x_0\|_2 \leq \epsilon$.*

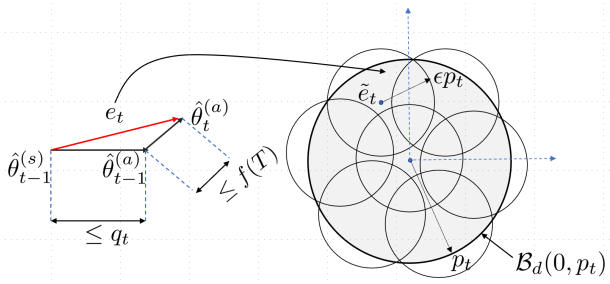

Figure 2: Illustration of the encoding technique in Algorithm 1. The agent computes the innovation signal e_t that belongs to $\mathcal{B}_d(0, p_t)$ with high probability. An ϵp_t-net of $\mathcal{B}_d(0, p_t)$ is constructed, and the center $\tilde{e}_t$ of the ball containing e_t is decoded by the server.

Next, consider the least-squares estimate $\hat{\theta}_t^{(a)}$ maintained by the agent: $\hat{\theta}_t^{(a)} = V_t^{-1} \sum_{s=1}^{t} a_s y_s$, where $V_t = \lambda I_d + \sum_{s=1}^{t} a_s a_s'$ is the covariance matrix at time-step t. Here, $\lambda > 0$ is a scalar regularization parameter. Let $\hat{\theta}_t^{(s)}$ be the estimate of θ_* maintained by the server; $\hat{\theta}_t^{(s)}$ is initialized from any arbitrary vector in Θ at time-step $\bar{T} + 1$. The choice of this initial vector is known to both the agent and the server.

Main Ideas. The key ideas guiding our encoding strategy are as follows. Once the agent has acquired sufficiently many observations, the gap $\hat{\theta}_t^{(a)} - \hat{\theta}_{t-1}^{(a)}$ between successive estimates will start shrinking due to the pure exploration phase. Thus, at this stage, if the gap $\hat{\theta}_{t-1}^{(a)} - \hat{\theta}_{t-1}^{(s)}$ is not too large, then the gap $e_t = \hat{\theta}_t^{(a)} - \hat{\theta}_{t-1}^{(s)}$ should not be too large either. In other words, eventually, a new observation y_t will not cause the agent's estimate of θ_* to deviate drastically from the estimate of θ_* held by the server. Intuitively, it thus makes sense to encode and transmit only the *new information* about θ_* contained in y_t, i.e., the "innovation" signal e_t (as opposed to encoding $\hat{\theta}_t^{(a)}$). However, given the stochastic nature of our setup, e_t is a random variable. Thus, encoding e_t poses the technical hurdle of characterizing the region containing e_t with high probability. To this end, in Lemma 3 of Section 4, we establish that with high probability, $\forall t \geq \bar{T} + 1$, $e_t \in \mathcal{B}_d(0, p_t)$, where p_t is the radius of the ball containing the innovation e_t. Our encoding strategy is adaptive since it requires dynamically updating the radius p_t (via Eq. (3)) based on statistical concentration bounds.

Algorithm 1 Adaptive Encoding at the Agent

1: **Input Parameters:** $\hat{\theta}_{\bar{T}}^{(s)}$ is any arbitrary vector in Θ; $q_{\bar{T}} = 10M$; and $f(T) = \frac{3}{5L}\sqrt{\frac{\beta_T}{T \log(dLT)}}$.
2: **for** $t \in \{\bar{T} + 1, \ldots, T\}$ **do**
3: Observe y_t; compute $\hat{\theta}_t^{(a)} = V_t^{-1} \sum_{s=1}^{t} a_s y_s$ and innovation $e_t = \hat{\theta}_t^{(a)} - \hat{\theta}_{t-1}^{(s)}$.
4: Construct an ϵp_t-net of $\mathcal{B}_d(0, p_t)$ to encode the innovation e_t, with p_t as given below:

$$q_t = \epsilon \left(q_{t-1} + f(T)\right); \quad p_t = q_t + f(T). \tag{3}$$

5: Determine the ball within $\mathcal{B}_d(0, p_t)$ that e_t falls into, and transmit the symbol $\sigma \in \Sigma$ corresponding to that ball. If $e_t \notin \mathcal{B}_d(0, p_t)$, transmit overflow symbol.
6: **end for**

Summary of Encoding Strategy. The overall encoding technique in Algorithm 1 can be summarized as follows. At each time-step $t \geq \bar{T}+1$, the agent observes y_t, computes $\hat{\theta}_t^{(a)}$, and then evaluates the innovation signal $e_t = \hat{\theta}_t^{(a)} - \hat{\theta}_{t-1}^{(s)}$. Given that $e_t \in \mathcal{B}_d(0, p_t)$ with high probability (as justified by Lemma 3), the region $\mathcal{B}_d(0, p_t)$ is covered by balls of radius ϵp_t, where $\epsilon \in (0,1)$ is a pre-decided constant, i.e., the agent constructs an ϵp_t-net of $\mathcal{B}_d(0, p_t)$.[3] The agent then determines the ball e_t falls into, and transmits the symbol $\sigma \in \Sigma$ corresponding to that ball.[4] If e_t falls outside $\mathcal{B}_d(0, p_t)$, the agent transmits a special symbol to indicate an overflow. We succinctly represent the entire operation described above by a dynamic encoder map $\mathcal{E}_t$ that takes as input e_t and generates as output the symbolic encoding $\sigma_t \in \Sigma$ that is transmitted to the server.

Decoding at Server. For correct decoding, we assume that the server is aware of the encoding operation at the agent. Note that the sequences $\{p_t\}$ and $\{q_t\}$ defined in Eq. (3) are deterministic, and can be computed by the server at its end. Thus, at any time-step $t \geq \bar{T}+1$, the server is aware of the region $\mathcal{B}_d(0, p_t)$ being encoded. Upon receiving σ_t, the server can thus correctly determine the center $\tilde{e}_t$ of the ball containing e_t. We represent the above decoding operation at time t by the decoder map $\mathcal{D}_t$ that takes as input σ_t and outputs $\tilde{e}_t$. Having decoded the innovation signal, the server computes an estimate $\hat{\theta}_t^{(s)}$ of θ_* as per line 11 of Algorithm 2. The agent computes $\hat{\theta}_t^{(s)}$ on its end as well in order to evaluate the innovation signal at time $t+1$. Our encoding-decoding technique is illustrated in Figure 2.

Till now, we have only described how to transit finite-precision information about θ_* from the agent to the server. However, the key question that remains unanswered is the following: *How should the server take decisions that yield low cumulative regret while accounting for the additional uncertainty introduced by the channel?* We now turn to answering this question.

• **Decision-Making at the Server.** When there is no loss of information over the channel, i.e., when $\hat{\theta}_t^{(s)} = \hat{\theta}_t^{(a)}$, the `LinUCB` algorithm relies on the principle of *optimism in the face of uncertainty*. Specifically, at each time-step, an ellipsoidal confidence set is constructed that contains θ_* with high-probability. The learner then acts optimistically by playing an action that yields the highest reward over all possible values of θ in the confidence set. Our approach builds on the same high-level principle, but relies *crucially on the construction of a new "inflated" ellipsoidal confidence set*:

$$\mathcal{C}_t^{(s)} = \{\theta \in \mathbb{R}^d : \|\theta - \hat{\theta}_{t-1}^{(s)}\|_{V_{t-1}} \leq \sqrt{\beta_T} + \left(\sqrt{\lambda + (t-1)L^2}\right) q_t\}, \quad \text{where} \tag{4}$$

$$\sqrt{\beta_T} = \sqrt{\lambda} M + \sqrt{2 \log\left(\frac{1}{\delta}\right) + d \log\left(\frac{d\lambda + TL^2}{d\lambda}\right)}, \tag{5}$$

q_t is as in Eq. (3), and $\delta \in (0,1)$ is a constant to be picked later. Notably, when $\hat{\theta}_{t-1}^{(s)} = \hat{\theta}_{t-1}^{(a)}$, and $q_t = 0$, $\mathcal{C}_t^{(s)}$ reduces to the confidence set in `LinUCB`. The inflation in the radius of the confidence set (relative to `LinUCB`) carefully accounts for the compression errors resulting from the finite capacity of the channel. Our main technical contribution here is to establish that $\forall t \geq \bar{T}+2$, $\theta_* \in \mathcal{C}_t^{(s)}$ with high probability; see Lemma 4 in Section 4. This result, in turn, justifies the optimistic decision-making rule of `IC-LinUCB` in line 12 of Algo. 2. During the pure exploration

3. For a discussion on constructing such coverings, see (Dumer et al., 2004; Verger-Gaugry, 2005) and the references therein.
4. In case e_t lands on the boundary of more than one ball, it is assigned the label/symbol of any one of those balls based on a fixed priority rule.

Algorithm 2 Information Constrained `LinUCB` (`IC-LinUCB`)

Input Parameters: $\bar{T} = \lceil 10L^2 d\sqrt{T}\log(dLT)\rceil$.
Phase I: *Pure Exploration*
for $t \in \{1, \ldots, \bar{T}+1\}$ **do**
  Server plays $a_t \sim \text{Unif}(\mathbb{S}^{d-1})$.
  Agent receives reward y_t as per (1) and computes estimate $\hat{\theta}_t^{(a)} = V_t^{-1}\sum_{s=1}^{t} a_s y_s$.
end for
Agent encodes $e_{\bar{T}+1} = \hat{\theta}^{(a)}_{\bar{T}+1} - \hat{\theta}^{(s)}_{\bar{T}}$ as per Algo. 1, and transmits $\sigma_{\bar{T}+1} = \mathcal{E}_{\bar{T}+1}\left(e_{\bar{T}+1}\right)$.

Phase II: *Information-Constrained Exploration-Exploitation*
for $t \in \{\bar{T}+2, \ldots, T\}$ **do**
  Server decodes $\tilde{e}_{t-1} = \mathcal{D}_{t-1}(\sigma_{t-1})$, and generates $\hat{\theta}^{(s)}_{t-1} = \hat{\theta}^{(s)}_{t-2} + \tilde{e}_{t-1}$.
  Server constructs the confidence set $\mathcal{C}_t^{(s)}$ described in Eq. (4), and plays the action $a_t = \text{argmax}_{a\in\mathcal{A}_t} \max_{\theta\in\mathcal{C}_t^{(s)}} \langle \theta, a\rangle$.
  Agent receives reward y_t as per (1) and computes estimate $\hat{\theta}_t^{(a)} = V_t^{-1}\sum_{s=1}^{t} a_s y_s$.
  Agent encodes the innovation $e_t = \hat{\theta}_t^{(a)} - \hat{\theta}_{t-1}^{(s)}$ as per Algo. 1, and transmits $\sigma_t = \mathcal{E}_t(e_t)$.
end for

phase, the server simply samples actions independently from the uniform distribution over $\mathbb{S}^{d-1}$, i.e., $a_t \sim \text{Unif}(\mathbb{S}^{d-1}), \forall t \in [\bar{T}+1]$.[5] At every $t \in [T]$, the action a_t decided upon by the server is passed down to the agent without any loss of information. This completes the description of `IC-LinUCB`.

4. Main Result and Analysis

Our main result concerning the performance of the `IC-LinUCB` algorithm is as follows.

Theorem 2 *(**Regret of `IC-LinUCB`**) Suppose Assumption 1 holds, and let the channel capacity satisfy $B \geq 6d$. Then, with $\epsilon = 1/2$ and $\delta = 1/T$, `IC-LinUCB` guarantees:*

$$R_T = O\left(L^2 d\sqrt{T}\log(dLT)\right) = \tilde{O}\left(d\sqrt{T}\right). \tag{6}$$

Discussion. We note that for the `IC-LinUCB` algorithm, the dependence of the regret on d and T exactly matches that of `LinUCB`. Thus, our work is the first to establish that with a horizon-independent channel capacity of $O(d)$ bits, one can achieve the same performance as when the channel has infinite capacity. Interestingly, Mayekar and Tyagi (2020) recently showed that for stochastic optimization with d-dimensional quantized gradients, a bit-rate of $\Omega(d)$ is necessary for achieving the optimal convergence rate of $O(1/\sqrt{T})$, where T is the number of iterations. Verifying whether a similar lower bound holds for our setting as well is left for future work.

Analysis. Due to space constraints, a detailed proof of Theorem 2 is omitted here, but can be found in (Mitra et al., 2022). Nonetheless, in what follows, we outline the key technical steps in the analysis of `IC-LinUCB`.

5. A random variable Z is uniformly distributed on $\mathbb{S}^{d-1}$ if, for every Borel subset $\mathcal{K} \subset \mathbb{S}^{d-1}$, the probability $\mathbb{P}(Z \in \mathcal{K})$ equals the ratio of the $(d-1)$-dimensional areas of $\mathcal{K}$ and $\mathbb{S}^{d-1}$.

There are three main steps in the proof of Theorem 2.

• **Step 1.** We construct an appropriate clean event $\mathcal{G}$ of measure at least $1 - 5/T$, and argue that on this event, the gap between successive model estimates at the agent is eventually small: in (Mitra et al., 2022, Lemma 8), we establish that $\|\hat{\theta}_{t+1}^{(a)} - \hat{\theta}_t^{(a)}\|_2 \leq f(T), \forall t \geq \bar{T}$, where $f(T)$ is as defined in the input parameters of Algorithm 1. The proof of this result, in turn, relies on the fact that with high probability, $\lambda_{\min}(V_t) \geq 5L^2\sqrt{T}\log(dLT), \forall t \geq \bar{T}$. The above claim is established in (Mitra et al., 2022, Lemma 6) by appealing to the Matrix Bernstein inequality.

• **Step 2.** The next key result justifies the encoding strategy in Algorithm 1.

Lemma 3 ***(Encoding Region)*** *With probability at least $1 - 5/T$, we have: $e_t \in \mathcal{B}_d(0, p_t), \forall t \in \{\bar{T}+1, \ldots, T\}$, where e_t is the innovation in line 3 of Algorithm 1, and p_t is as defined in Eq. (3).*

Lemma 3 tells us that with high probability, the innovation random variable e_t always falls within the desired encoding region, i.e., there is never any overflow on the event $\mathcal{G}$.

• **Step 3.** It remains to justify the choice of the confidence set $\mathcal{C}_t^{(s)}$ in Eq. (4). This is achieved in the following lemma.

Lemma 4 ***(Confidence Region at Server)*** *With probability at least $1 - 5/T$, the following is true: $\theta_* \in \mathcal{C}_t^{(s)}, \forall t \in \{\bar{T}+2, \ldots, T\}$, where $\mathcal{C}_t^{(s)}$ is the confidence set defined in Eq. (4). Moreover, $\forall t \geq \bar{T} + \tilde{T}$, we have $\left(\sqrt{\lambda + (t-1)L^2}\right) q_t \leq 4\sqrt{\beta_T / \log(dLT)}$, where $\tilde{T} = O\left(\log(dLT)\right)$.*

The above result implies that the inflated confidence set (that accounts for encoding errors) eventually contains the true parameter θ_* with high probability. At the same time, the quantization error q_t decays fast enough to ensure that the radius of the confidence set is eventually $O(\sqrt{\beta_T})$ - exactly as in the `LinUCB` algorithm. In other words, our approach ensures that the impact of the quantization error on decision-making vanishes over time.

5. One Bit Capacity is Sufficient for the Multi-armed Bandit Problem

In Section 3, we saw that for a d-dimensional model, $O(d)$ bits suffice to achieve order-optimal regret. In this section, we investigate whether one can achieve similar order-optimal regret bounds with fewer bits when the set of feasible actions has additional structure. We will show that this is indeed the case for a particular setting of interest when $\mathcal{A}_t = \{e_1, \ldots, e_d\}, \forall t \in [T]$, where $(e_i)_i$ are the standard orthonormal unit vectors. This setting represents the popular unstructured multi-armed bandit problem with a finite number of arms. Our main insight is the following: playing action/arm i only reveals information about the i-th component of θ_*, denoted by θ_i, and hence, when the i-th action is played, it makes sense for the agent to encode and transmit the innovation related to only θ_i. In other words, the above intuition suggests that encoding a scalar innovation signal (as opposed to a d-dimensional innovation vector) should suffice for the specific setting under consideration. In what follows, we formalize this reasoning.

To get started, let us note that the optimal action a_* is the unit vector corresponding to the largest component of θ_*. Without loss of generality, let this component be θ_1, i.e., $\theta_1 = \max_{i \in [d]} \theta_i$. We thus have $a_* = e_1$. Let us denote by $\hat{\theta}_{i,k}^{(a)}$ (resp., $\hat{\theta}_{i,k}^{(s)}$) the estimate of θ_i at the agent (resp., at the server) after arm i has been played k times. We now develop an information-constrained variant of the celebrated upper confidence bound algorithm that we call `IC-UCB`.

Description of `IC-UCB`. Let $\gamma = 1/2^B$ where B is the channel capacity, and define:

$$p_{k+1} = \gamma p_k + 2f_k; \quad q_k = \gamma p_k; \quad f_k = 2\sqrt{\log T/k}, \tag{7}$$

where $p_1 = m + f_1$, and $m \geq 1$ is such that $\max_{i\in[d]} |\theta_i| \leq m$. Suppose the action at time t is $a_t = e_i$. The agent first updates its estimate of θ_i:

$$\hat{\theta}^{(a)}_{i,n_i(t)} = \frac{1}{n_i(t)} \sum_{k=1}^{n_i(t)} y_{i,k}, \tag{8}$$

where $y_{i,k}$ is the agent's observation when the i-th arm is played the k-th time, and $n_i(t)$ is the number of times arm i is played up to (and including) time-step t. It then computes the *scalar* innovation $e_{i,n_i(t)} = \hat{\theta}^{(a)}_{i,n_i(t)} - \hat{\theta}^{(s)}_{i,n_i(t)-1}$, where $\hat{\theta}^{(s)}_{i,0} = 0, \forall i \in [d]$. If $e_{i,n_i(t)}$ falls in the interval $\mathcal{Z}_{i,t} = [-p_{n_i(t)}, p_{n_i(t)}]$, then $\mathcal{Z}_{i,t}$ is partitioned uniformly into 2^B bins, and the symbol $\sigma_t \in \Sigma$ encoding the bin containing $e_{i,n_i(t)}$ is transmitted to the server. The server then decodes the center $\tilde{e}_{i,n_i(t)}$ of that bin, and computes $\hat{\theta}^{(s)}_{i,n_i(t)} = \hat{\theta}^{(s)}_{i,n_i(t)-1} + \tilde{e}_{i,n_i(t)}$. If $e_{i,n_i(t)} \notin \mathcal{Z}_{i,t}$, then there is no transmission from the agent to the server. As for decision-making, each arm is first played once by the server. Subsequently, the action chosen by the server at time-step $t+1$ is the one that maximizes the following index:

$$\texttt{IC-UCB}_i(t) = \hat{\theta}^{(s)}_{i,n_i(t)} + q_{n_i(t)} + f_{n_i(t)}, \tag{9}$$

where $q_{n_i(t)}$ and $f_{n_i(t)}$ are generated as per Eq. (7). Let $\Delta_i = \theta_1 - \theta_i$ denote the sub-optimality gap of arm i. To present our results in a clean way, we will focus on the particularly important case where the sub-optimality gaps are small: $\Delta_i \in (0, 1], \forall i \in [d] \setminus \{1\}$. Our results can be easily generalized to arbitrary sub-optimality gaps. For the setting considered in this section, it is easy to verify that the regret R_T in Eq. (2) simplifies to $R_T = \sum_{i=1}^d \Delta_i \mathbb{E}[n_i(T)]$.

The main results of this section are as follows.

Theorem 5 ***(Regret of `IC-UCB`)*** *Suppose the channel capacity is at least 1 bit, i.e., $B \geq 1$. The `IC-UCB` algorithm then guarantees: $R_T \leq 5\sum_{i=1}^d \Delta_i + \sum_{i=1}^d O\left(\log(mT)/\Delta_i\right)$.*

We can also establish the following gap-independent bound.

Theorem 6 ***(Gap-independent bound)*** *Suppose $B \geq 1$. The `IC-UCB` algorithm then guarantees: $R_T \leq 5\sum_{i=1}^d \Delta_i + O\left(\sqrt{dT\log(mT)}\right)$.*

Discussion: Our bounds above match those for `UCB`, revealing that for the MAB problem, one can achieve order-optimal bounds with a bit-rate of *just* 1 *bit*. The main takeaway here is that when the action sets have more structure, one can achieve optimal performance with fewer than $O(d)$ bits. As future work, it would be interesting to see if one can draw similar conclusions for other types of common action sets.

6. Conclusion

We introduced and studied a new linear stochastic bandit problem subject to communication channel constraints. We developed a general algorithmic framework comprising of an adaptive compression mechanism, and a decision-making rule that explicitly accounts for encoding errors. We then showed how this framework leads to order-optimal regret bounds for (i) the linear bandit setting, and (ii) the MAB problem, with horizon-independent bit-rates. Ongoing work involves deriving lower bounds for our setup, and also generalizing our algorithms and results to MDPs.

Acknowledgments

This work was supported by NSF Award 1837253, NSF CAREER award CIF 1943064, and the Air Force Office of Scientific Research Young Investigator Program (AFOSR-YIP) under award FA9550-20-1-0111.

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24:2312–2320, 2011.

Mridul Agarwal, Vaneet Aggarwal, and Kamyar Azizzadenesheli. Multi-agent multi-armed bandits with limited communication. *arXiv preprint arXiv:2102.08462*, 2021.

Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. *Advances in Neural Information Processing Systems*, 30:1709–1720, 2017.

Sanae Amani, Mahnoosh Alizadeh, and Christos Thrampoulidis. Linear stochastic bandits under safety constraints. *arXiv preprint arXiv:1908.05814*, 2019.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002.

Ronshee Chawla, Abishek Sankararaman, Ayalvadi Ganesh, and Sanjay Shakkottai. The gossiping insert-eliminate algorithm for multi-agent bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 3471–3481. PMLR, 2020.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. 2008.

Thinh Doan, Siva Maguluri, and Justin Romberg. Finite-time analysis of distributed td (0) with linear function approximation on multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 1626–1635. PMLR, 2019.

Abhimanyu Dubey and Alex Pentland. Differentially-private federated linear bandits. *arXiv preprint arXiv:2010.11425*, 2020.

Ilya Dumer, Mark S Pinsker, and Vyacheslav V Prelov. On coverings of ellipsoids in euclidean spaces. *IEEE transactions on information theory*, 50(10):2348–2356, 2004.

Venkata Gandikota, Daniel Kane, Raj Kumar Maity, and Arya Mazumdar. vqsgd: Vector quantized stochastic gradient descent. In *International Conference on Artificial Intelligence and Statistics*, pages 2197–2205. PMLR, 2021.

Konstantinos Gatsis, Hamed Hassani, and George J Pappas. Latency-reliability tradeoffs for state estimation. *IEEE Transactions on Automatic Control*, 66(3):1009–1023, 2020.

Ruiquan Huang, Weiqiang Wu, Jing Yang, and Cong Shen. Federated linear contextual bandits. *Advances in Neural Information Processing Systems*, 34, 2021.

Hao Jin, Yang Peng, Wenhao Yang, Shusen Wang, and Zhihua Zhang. Federated reinforcement learning with environment heterogeneity. In *International Conference on Artificial Intelligence and Statistics*, pages 18–37. PMLR, 2022.

Anatoly Khina, Elias Riedel Gårding, Gustav M Pettersson, Victoria Kostina, and Babak Hassibi. Control over gaussian channels with and without source–channel separation. *IEEE Transactions on Automatic Control*, 64(9):3690–3705, 2019.

Sajad Khodadadian, Pranay Sharma, Gauri Joshi, and Siva Theja Maguluri. Federated reinforcement learning: Linear speedup under markovian sampling. In *International Conference on Machine Learning*, pages 10997–11057. PMLR, 2022.

Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 167–172. IEEE, 2016.

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

Chung-Yi Lin, Victoria Kostina, and Babak Hassibi. Differentially quantized gradient descent. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 1200–1205. IEEE, 2021.

Alexey S Matveev. State estimation via limited capacity noisy communication channels. *Mathematics of Control, Signals, and Systems*, 20(1):1–35, 2008.

Prathamesh Mayekar and Himanshu Tyagi. Ratq: A universal fixed-length quantizer for stochastic optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1399–1409. PMLR, 2020.

Nicole Mitchell, Johannes Ballé, Zachary Charles, and Jakub Konečnỳ. Optimizing the communication-accuracy trade-off in federated learning with rate-distortion theory. *arXiv preprint arXiv:2201.02664*, 2022.

Aritra Mitra, Hamed Hassani, and George J Pappas. Linear stochastic bandits over a bit-constrained channel. *arXiv preprint arXiv:2203.01198*, 2022.

Reza Olfati-Saber. Distributed kalman filter with embedded consensus filters. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 8179–8184. IEEE, 2005.

Rafail Ostrovsky, Yuval Rabani, and Leonard J Schulman. Error-correcting codes for automatic control. *IEEE Transactions on Information Theory*, 55(7):2931–2941, 2009.

Jiaju Qi, Qihao Zhou, Lei Lei, and Kan Zheng. Federated reinforcement learning: techniques, applications, and open challenges. *arXiv preprint arXiv:2108.11887*, 2021.

Abishek Sankararaman, Ayalvadi Ganesh, and Sanjay Shakkottai. Social learning in multi agent multi armed bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3 (3):1–35, 2019.

Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

Shahin Shahrampour, Alexander Rakhlin, and Ali Jadbabaie. Multi-armed bandits in multi-agent networks. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2786–2790. IEEE, 2017.

Alberto Speranzon, Carlo Fischione, and Karl Henrik Johansson. Distributed and collaborative estimation over wireless sensor networks. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 1025–1030. IEEE, 2006.

Sebastian U Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified sgd with memory. In *Advances in Neural Information Processing Systems*, pages 4447–4458, 2018.

Ravi Teja Sukhavasi and Babak Hassibi. Linear time-invariant anytime codes for control over noisy channels. *IEEE Transactions on Automatic Control*, 61(12):3826–3841, 2016.

Sekhar Tatikonda and Sanjoy Mitter. Control under communication constraints. *IEEE Transactions on automatic control*, 49(7):1056–1068, 2004a.

Sekhar Tatikonda and Sanjoy Mitter. Control over noisy channels. *IEEE transactions on Automatic Control*, 49(7):1196–1201, 2004b.

Jean-Louis Verger-Gaugry. Covering a ball with smaller equal balls in $\mathbb{R}^n$. *Discrete & Computational Geometry*, 33(1):143–155, 2005.

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

Yuanhao Wang, Jiachen Hu, Xiaoyu Chen, and Liwei Wang. Distributed bandit learning: Near-optimal regret with efficient communication. *arXiv preprint arXiv:1904.06309*, 2019.

Wei Wen, Cong Xu, Feng Yan, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Terngrad: Ternary gradients to reduce communication in distributed deep learning. In *Advances in neural information processing systems*, pages 1509–1519, 2017.

Jiaqi Yang, Wei Hu, Jason D Lee, and Simon Shaolei Du. Impact of representation learning in linear bandits. In *International Conference on Learning Representations*, 2020.