Solving Multi-Arm Bandit Using a Few Bits of Communication

Osama A. Hanna
UCLA
ohanna@ucla.edu

Lin F. Yang
UCLA
linyang@ucla.edu

Christina Fragouli
UCLA
christina.fragouli@ucla.edu

Abstract

The multi-armed bandit (MAB) problem is an active learning framework that aims to select the best among a set of actions by sequentially observing rewards. Recently, it has become popular for a number of applications over wireless networks, where communication constraints can form a bottleneck. Existing works usually fail to address this issue and can become infeasible in certain applications. In this paper we address the communication problem by optimizing the communication of rewards collected by distributed agents. By providing nearly matching upper and lower bounds, we tightly characterize the *number of bits* needed per reward for the learner to accurately learn without suffering additional regret. In particular, we establish a generic reward quantization algorithm, *QuBan*, that can be applied on top of any (no-regret) MAB algorithm to form a new communication-efficient counterpart, that requires only a few (as low as 3) bits to be sent per iteration while preserving the *same* regret bound. Our lower bound is established via constructing hard instances from a subgaussian distribution. Our theory is further corroborated by numerically experiments.

1 Introduction

Multi-armed bandit (MAB) is an active learning framework that finds applications in diverse domains, including recommendation systems, clinical trials, adaptive routing, and so on [Bouneffouf and Rish, 2019]. In a MAB problem, a learner interacts with an environment by pulling an arm from a set of arms, each of which, if played, gives a scalar reward, sampled from an unknown but fixed distribution. The goal of the learner is to find the arm with the highest mean reward using a minimum number of pulls. The performance of a learner is measured in terms of regret, that captures the expected difference between the observed rewards and rewards drawn from the best arm. Work on MAB algorithms and their applications spans several decades, cultivating a rich literature that considers a variety of models and algorithmic approaches [Lattimore and Szepesvári, 2020]. MAB algorithms include explore-then-commit [Robbins, 1952, Anscombe, 1963], ϵ-greedy [Auer et al., 2002], Thomson sampling [Thompson, 1933], and the upper confidence bound (UCB) [Lai, 1987, Auer et al., 2002], to name a few. Under some assumptions on the reward distribution, the explore-then-commit and ϵ-greedy algorithms achieve a regret bound $\propto O(\sqrt{n})$, where n is number of steps the learner plays, for the worst-case but known minimum reward gap[1], while Thomson sampling and UCB achieve a regret bound $\propto O(\sqrt{n \log(n)})$ without knowledge of the minimum means gap[2]. However, all these works assume that the rewards can be communicated to the learner at full precision which can be costly in communication constrained setups. In this paper we ask: *is it possible to perform efficient and effective bandit learning with only a few bits communicated per reward?*

Understanding how many bits of communication are really needed, is not only interesting from a theoretical viewpoint, but can also enable the MAB framework to support learning applications in settings that were challenging before. Consider, for instance, swarms of tiny robots (such as RoboBees and RoboFlies [Wood et al., 2013]), wearable (inside and outside the body) sensors, backscatterer and RFID networks, IoT and embedded systems; generally whenever low complexity

---

[1]The reward gap is defined to be the difference between the reward means of the best and second best arm.

[2]Variants of the UCB [Audibert et al., 2009, Degenne and Perchet, 2016] can achieve regret $\propto O(\sqrt{n})$, but can be worse than UCB in some regimes [Lattimore and Szepesvári, 2020].
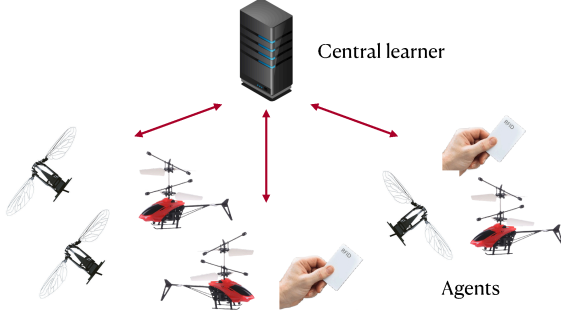

Figure 1: A central learner collects rewards from a set of agents. The agents can join and leave at any time and hence can be different and unaware of the historical rewards, i.e., memoryless.

sensors cooperate, the communication cost can become a performance bottleneck for any learning framework. MAB systems in areas such as mobile healthcare, social decision-making and spectrum allocation have already been implemented in a distributed manner, using limited bandwidth wireless links and simple sensors with low computational power [Anandkumar et al., 2011, Buccapatnam et al., 2013, 2014, Mary et al., 2015, Song et al., 2018, Ding et al., 2019]; reducing the number of bits directly translates to reduced power consumption and wireless interference for these systems.

In this paper we consider a common setup illustrated in Fig. 1, where a central learner can directly communicate with a set of agents. We assume that the agents may change from time to time (e.g., are mobile), but that each agent can pull whichever arm the learner requests it to, observe the reward, and communicate the reward to the learner. For example, the learner could be a "traffic policeman" for small drones that searches best current policies; or a base-station that helps low-capability sensors achieve spectrum sharing. For many existing systems, the learner may have already implemented a MAB algorithm to handle the learning task. Hence our goal is to design a communication protocol such that the rewards are communicated with only a few bits and yet the performance of the original MAB algorithm does not degrade.

Our main contribution is a set of upper and lower bounds on the required number of bits to achieve the same unquantized regret up to a small constant factor. In particular, our lower bound states that we need to send at least 2.2 bits per reward to maintain properties of the reward that enable to achieve a regret within a factor of 1.5 from the unquantized regret. This lower bound is established through proving necessary properties of the optimal quantizer, then constructing a hard instance from a gaussian distribution. Our upper bounds state that, on average, 3.4 bits are sufficient to maintain the required properties and achieve a regret

within a factor of 1.5 from the unquantized regret (and that we rarely need to use more than 4 bits).

The upper bounds are proved using a novel quantization scheme, that we term $QuBan$, tailored to compressing MAB rewards. $QuBan$ only cares to maintain what matters to the MAB algorithm operation, namely the ability to decide which is the best arm. At a high level, $QuBan$ maps rewards to quantization levels chosen to be dense around an estimate of the arm's mean values and sparse otherwise. $QuBan$ employs a stochastic correction term that enables to convey an unbiased estimate of the rewards with a small variance. $QuBan$ introduces a simple novel rounding trick to guarantee that the quantization error is conditionally independent on the history given the current pulled arm index. This maintains the Markov property which is crucial in the analysis of bandit algorithms and enables reusing the same analysis methods for unquantized rewards to bound the regret after quantization. Finally, $QuBan$ encodes the reward values that occur more frequently with shorter representations, in order to reduce the number of bits communicated. We provide empirical studies for a number of MAB algorithms, e.g., UCB and ϵ-greedy. Numerical results corroborate that $QuBan$, applied on top of MAB algorithms, using a few bits (as small as 3) can achieve the same regret as unquantized communication.

Related Work. To the best of our knowledge, the proposed model is novel and no scheme from the literature can be used to solve the problem of maintaining a regret bound that matches the unquantized regret bound, up to a small constant factor, while using a few bits of communication. In the following, we distinguish our work from the existing literature.

MAB algorithms. There is a long line of research in the literature about MAB algorithms. For instance, explore-then-commit [Robbins, 1952, Anscombe, 1963], ϵ-greedy [Auer et al., 2002], Thomson sampling [Thompson, 1933], the upper confidence bound [Lai, 1987, Auer et al., 2002] and its variant for contextual bandits [Dani et al., 2008, Li et al., 2010]. Under the assumption that the reward distributions are 1-subgaussian, these algorithms provide a worst case regret that is almost $O(\sqrt{n})$. The explore-then-commit regret is upper bounded by $C\sqrt{n}$ for bandits with 2 arms and known minimum means gap, while the regret of ϵ-greedy is upper bounded by $C'\sqrt{kn}$ for k-armed bandits with knowledge of the minimum means gap, where C, C' are constants that do not depend on k, n [Lattimore and Szepesvári, 2020]. Thomson sampling and UCB achieve a regret with upper bound $C\sqrt{kn \log(n)}$ for k-armed bandits, where C is a constant that does not depend on k, n [Russo and Van Roy, 2014, Agrawal and Goyal, 2012, 2013a, Katehakis and Robbins, 1995,

Agrawal, 1995, Auer et al., 2002]. Contextual Thomson sampling and LinUCB achieve a regret bounded by $Cd\sqrt{n}\log(n)$, where d is the dimension of an unknown system parameter, and C is a universal constant that does not depend on n, d [Agrawal and Goyal, 2013b, Abeille and Lazaric, 2017, Lattimore and Szepesvári, 2020, Dani et al., 2008]. These algorithms assume access to a full precision reward at each iteration. Our goal is not replacing existing MAB algorithms to deal with quantized rewards; instead, we are interested in a general quantization framework that can be applied on top of any existing (or future) MAB algorithm.

Compression for ML and distributed optimization. There is a number of research results targeting reducing the communication cost of learning systems using compression. For instance, compression is applied on gradient updates [Seide et al., 2014, Alistarh et al., 2017, Mayekar and Tyagi, 2020]. Recent work has also looked at compression for classification tasks [Hanna et al., 2020]. However, compression schemes tailored to active learning, such as MAB problems, have not been explored. Our quantization scheme can be understood as a reward compression scheme that reduces the communication complexity for MABs. The main difference between the quantization for MABs and for distributed learning is that the later targets reducing the dependency of the number of bits and performance on the dimensionality of bounded training data, which can be in the order of tens of millions. In contrast, the rewards of MABs are scalars. The main challenge of our setting is to deal with a reward distribution that is either unbounded or the upper bound on the reward is much larger than the noise variance, which is typical in many MAB applications. This can be done by exploiting the fact that the rewards are more likely to be picked from the arm that appear to be best. Such a property is not applicable in the general distributed optimization setup and comes with new challenges as will discussed later.

Sample complexity. Compression is related to sample complexity [Even-Dar et al., 2002, Mannor and Tsitsiklis, 2004]: indeed, sending a small number of samples, reduces the overall communication load. However, the question we ask is different (and complementary): sample complexity asks how many (full precision) samples from each distribution do we need to draw; we are asking, how many bits of each sample do we really need to transmit, when we only care to decide the best arm and not to reconstruct the samples.

Distributed multi-agent MAB. Researchers have explored the distributed multi-agent MAB problem with a single [Anantharam et al., 1987] or multiple [Shahrampour et al., 2017] decision makers; in these settings, distributed agents pick arms under some constraints

(all agents pick the same arm [Shahrampour et al., 2017], at most one agent can pick the same arm at a time otherwise no reward is given [Anandkumar et al., 2011] and other constraints [Landgren et al., 2019]). The agents cooperate to aggregate their observed rewards so as to jointly make a more informed decision on the best arm. Most of the works do not take into account communication constraints, and rather focus on cooperation/coordination schemes. Our setup is different: we have a single learner (central server) and simple agents who do not learn (do not keep memory) but simply observe and transmit rewards, one at a time. Our scheme can be potentially applied to these settings to reduce communication cost.

Within this previous framework, work considers "batched" rewards, where agents keep their observed rewards in memory and communicate them infrequently, potentially summarizing their findings and thus reducing the communication load [Perchet et al., 2016, Esfandiari et al., 2019, Even-Dar et al., 2006]. Such schemes require agents to be present for the whole duration of learning, and address a different setup than ours. Practical cases where our setup fits better than batched rewards include: (i) agents with very limited memory/computational power, such as RFIDs; (ii) agents that can meaningfully implement an action and report the reward only once (eg., watching a video advertisement); (iii) mobile agents that do not remain connected for the duration of learning (eg., passing by drones directed to perform a manoeuvre) or (iv) agents simply not willing to invest the resources required to support a batched MAB learning process.

Independently and in parallel to ours, the work in [Vial et al., 2020] also considered MAB learning with reduced number of bits, restricted in their case to UCB policies. Their main result shows that for rewards supported on $[0, 1]$, one bit of communication is sufficient; our work recovers this result using a much simpler approach as a special case of Section 3. Additionally, our work applies on top of any MAB algorithm, and for unbounded rewards.

Paper Organization. Section 2 presents our model and notation; Section 3 looks at a special case; Section 4 describes *QuBan*; Section 5 presents our main theorems and Section 6 provides numerical evaluation.

2 Model and Notation

MAB Framework. We consider a multi-armed bandit (MAB) problem over a horizon of size n [Robbins, 1952]. At each iteration $t = 1, ..., n$, a learner chooses an arm (action) A_t from a set of arms $\mathcal{A}_t$ and receives a random reward r_t distributed according to an unknown reward distribution with mean μ_{A_t}. The

reward distributions are assumed to be σ^2-subgaussian [Boucheron et al., 2013]. The arm selected at time t depends on the previously selected arms and observed rewards $A_1, r_1, ..., A_{t-1}, r_{t-1}$. The learner is interested in minimizing the expected regret $R_n = \mathbb{E}[R'_n]$, where R'_n is the regret defined as

$$R'_n = \Sigma_{t=1}^n (\mu_t^* - r_t), \tag{1}$$

where $\mu_t^* = \max_{A \in \mathcal{A}_t} \mu_A$. The expected regret captures the difference between the expected total reward collected by the learner over n iterations and the reward if we selected the arm with the maximum mean (optimal arm).

Notation. When the set of arms $\mathcal{A}_t$ is finite and does not depend on t: we denote the number of arms by $k = |\mathcal{A}_t|$, the best arm mean by μ^*, and the gap between the best arm and the arm-i mean by $\Delta_i := \mu^* - \mu_i$. If X, Y are random variables, we refer to the expectation of X, variance of X, conditional expectation of X given Y, and conditional variance of X given Y as $\mathbb{E}[X]$, $\sigma^2(X)$, $\mathbb{E}[X|Y]$, and $\sigma^2(X|Y)$ respectively.

Popular MAB algorithms for the case where the set of actions is fixed over time, $\mathcal{A} = \mathcal{A}_t$, and $\mathcal{A}$ is finite include explore-then-commit [Robbins, 1952, Anscombe, 1963], ϵ-greedy [Auer et al., 2002], Thomson sampling [Thompson, 1933], and UCB [Lai, 1987, Auer et al., 2002]. In addition to this case we also consider an important class of bandit problems, contextual bandits [Langford and Zhang, 2007, Agrawal and Goyal, 2013b]. In this case, before picking an action, the learner observes a side information, the context. Specifically we consider the widely used stochastic linear bandits model [Abe and Long, 1999], where the contexts are modeled by changing the action set $\mathcal{A}_t$ across time. In this model, at iteration t, the learner chooses an action A_t from a given set $\mathcal{A}_t \subseteq \mathbb{R}^d$ and gets a reward

$$r_t = \langle \theta_*, A_t \rangle + \eta_t, \tag{2}$$

where $\theta_* \in \mathbb{R}^d$ is an unknown parameter, and η_t is a noise. Conditioned on $\mathcal{A}_1, A_1, r_1, ..., \mathcal{A}_t, A_t, r_t$, the noise η_{t+1} is assumed to be zero mean and σ^2-subgaussian. Popular algorithms for this case include LinUCB [Dani et al., 2008], explore-then-commit strategy [Rusmevichientong and Tsitsiklis, 2010], and contextual Thomson sampling [Agrawal and Goyal, 2013b].

System Setup. We are interested in a distributed setting, where a learner asks at each time a potentially different agent to play the arm A_t; the agent observes the reward r_t and conveys it to the learner over a communication constrained channel, as depicted in Fig. 1. In our setup, each agent needs to immediately communicate the observed reward (with no memory), using a quantization scheme to reduce the communication cost.

As learning progresses, the learner is allowed to refine the quantization scheme by broadcasting parameters to the agents they may need. We do not count these broadcast (downlink) transmissions in the communication cost since the learner has no restrictions in its power. We stress again that the agents cannot store information of the reward history since they may *join and leave* the system at any time. We thus opt to use a setting where the agents have no memory. This setting allows to support applications with simple agents (e.g. RFID applications and embedded systems).

Quantization. A quantizer consists of an encoder $\mathcal{E} : \mathbb{R} \to \mathcal{S}$ that maps $\mathbb{R}$ to a countable set $\mathcal{S}$, and a decoder $D : \mathcal{S} \to \mathbb{R}$. At each time t, the agent that observes the reward r_t transmits a finite length binary sequence representing $\mathcal{E}(r_t)$ to the learner which in turn decodes it using the decoder D to obtain the quantized reward $\hat{r}_t = D(\mathcal{E}(r_t))$. The range of a decoder is referred to as the set of quantization levels; the end-to-end operation of a quantizer maps the reward to a quantization level. We next describe a specific quantization module that we will use.

Stochastic Quantization (SQ). A stochastic quantizer that uses quantization levels in a set $\mathcal{L}$, which is a form of dithering [Gray and Stockham, 1993, Alistarh et al., 2017], consists of a randomized encoder $\mathcal{E}_{\mathcal{L}}$ and decoder $D_{\mathcal{L}}$ modules that can be described as following. The encoder $\mathcal{E}_{\mathcal{L}}$, that uses the set of quantization levels $\mathcal{L} = \{\ell_i\}_{i=1}^{2^B}$, takes as input a value x in $[\ell_1, \ell_{2^B}]$; it maps x to a level index described by B bits. The decoder, that uses the set of quantization levels $\mathcal{L} = \{\ell_i\}_{i=1}^{2^B}$, takes as input an index in $\{1, ..., 2^B\}$, and outputs the corresponding level value. Precisely,

$$i(x) = \max\{j | \ell_j \leq x \text{ and } j < 2^B\},$$

$$\mathcal{E}_{\mathcal{L}}(x) = \begin{cases} i(x) & \text{with probability } \frac{\ell_{i(x)+1} - x}{\ell_{i(x)+1} - \ell_{i(x)}} \\ i(x) + 1 & \text{with probability } \frac{x - \ell_{i(x)}}{\ell_{i(x)+1} - \ell_{i(x)}} \end{cases},$$

$$D_{\mathcal{L}}(j) = \ell_j, j \in \{1, ..., 2^B\}. \tag{3}$$

That is, if x is such that $\ell_i \leq x < \ell_{i+1}$, then the index i is transmitted with probability $\frac{\ell_{i+1} - x}{\ell_{i+1} - \ell_i}$ (and x is decoded to be ℓ_i) while the index $i+1$ is transmitted with probability $\frac{x - \ell_i}{\ell_{i+1} - \ell_i}$ (and x is decoded to be ℓ_{i+1}).

The analysis of bandit algorithms leverages the fact that conditioned on A_t, the communicated reward r_t is an unbiased estimate of the mean μ_{A_t}. It is not difficult to see that SQ preserves this property, namely conditioned on A_t, it conveys to the learner an unbiased estimate of μ_{A_t}.

Performance Metric $B_n, \bar{B}(n)$. Among the schemes that achieve a regret matching the unquantized regret, up to a fixed small constant factor, our performance

metrics are the instantaneous and average number of communication bits per reward B_n, and $\bar{B}(n)$ respectively. Let B_t be the number of bits used to transmit $\hat{r}_t$, and define the average number of bits after n iterations of the algorithm as $\bar{B}(n) = \frac{\sum_{t=1}^{n} B_t}{n}$. Our goal is to design quantization schemes that achieve expected regret matching the expected regret of unquantized communication (up to a small constant factor) while using a small number of bits B_n, and $\bar{B}(n)$.

3 A Case Where 1 Bit is Sufficient

In this section we consider the special case where the rewards are supported on $[0, 1]$. For simplicity, we also assume that all reward distributions have the same range and the same variance σ (but different means).

We explore schemes that use exactly one bit per reward; note that one bit is a trivial lower bound on the number of bits communicated, since each agent needs to respond to the learner for each observed reward. In particular, we use 1-bit Stochastic Quantization (SQ), as in (3). Assume that $r_t \in [0, 1]$ and the variance $\sigma^2(r_t|A_t) \approx 1/4$. The stochastic 1 bit quantizer takes r_t as input and interprets it as probability: outputs 1 with probability r_t and 0 with probability $1 - r_t$. Let $\hat{r}_t$ be the (binary) quantized reward, we then have that

$$\mathbb{E}[\hat{r}_t|A_t] = \mathbb{E}[\mathbb{E}[\hat{r}_t|r_t, A_t]|A_t] = \mathbb{E}[r_t|A_t] = \mu_{A_t}. \quad (4)$$

Since $\hat{r}_t \in [0, 1]$, its variance is upper bounded by $\frac{1}{4}$. Recall that for bandit algorithms the expected regret scales linearly with the variance. For example, the UCB algorithm (c.f., [Lattimore and Szepesvári, 2020]) with unquantized rewards, achieves $R_n \le C\sqrt{nk \log(n)}$ for a constant C that does not depend on k, n. It is not difficult to see that similar to [Auer et al., 2002], UCB with 1-bit SQ achieves a regret bound $R_n \le C\sqrt{nk \log(n)}$. Simulation results verify that, for $r_t \in [0, 1]$, 1-bit SQ performs very close to unquantized rewards (proofs and simulation results are in App. I).

To motivate our general quantization scheme, we consider a case where 1-bit SQ results in a potentially large performance loss. Assume that the variance, σ, is much smaller than the range of r_t: $r_t \in [-\lambda, \lambda]$ and $\sigma = 1$, where $\lambda \gg 1$ is a parameter known to the learner. The 1-bit SQ maps r_t to either λ or $-\lambda$; it is not difficult to see that we still have $\mathbb{E}[\hat{r}_t|A_t] = \mu_{A_t}$, but $R_n \le C\lambda\sqrt{kn \log(n)}$, where C is a constant that does not depend on n, k [Auer et al., 2002][3]. Thus the expected regret bound grows linearly with λ, which can be arbitrarily large. In contrast, without quantization

---

[3] We note that this bound cannot be improved using techniques in [Auer et al., 2002], since it is possible that $\sigma^2(\hat{r}_t|A_t) = \lambda^2$ (e.g., if $r_t = 0$ almost surely).
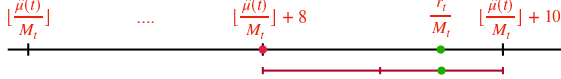
UCB still achieves $C'\sqrt{kn \log(n)}$, where C' is another constant of the same order of C. Simulation results (in App. I) verify that the convergence to the unquantized case can be slow.

We take away the following observations:
• If the range λ is of the same order as the variance σ, 1-bit SQ is sufficient to preserve the regret bound up to a small constant factor.
• If the range λ is much larger than σ, 1-bit SQ leads to a regret penalty proportional to $\frac{\lambda}{\sigma}$; thus we may want to only perform stochastic quantization within intervals of size similar to σ.

In this section we assumed that the rewards r_t are bounded almost surely. This is not true in general; we would like an algorithm that uses a small average number of bits even when the reward distributions are unbounded; $QuBan$, described next, achieves this.

4 $QuBan$: A General Quantizer for Bandit Rewards

In this section, we propose $QuBan$, an adaptive quantization scheme that can be applied on top of any MAB algorithm. Our scheme maintains attractive properties (such that the Markov property, unbiasedness, and bounded variance) for the quantized rewards that enable to retain the same regret bound as unquantized communication for the vast majority of MAB algorithms, while using a few bits for communication (simulation results show convergence to ~ 3 bits per iteration for n that is sufficiently large, see App. H). We use ideas that include: (i) centering the quantization scheme around a value that is believed to be close to the picked arm mean in the majority of iterations; (iii) maintaining a quantization error that is conditionally independent on previously observed rewards given the arm selection, which is achieved by choosing the quantization center to be an integer value (illustrated in more detail in the proof of Theorem 2); (iv) assigning shorter codes to the values near the quantization center and otherwise longer codes to maintain a finite expected number of bits even if the reward distribution has infinite support; and (iv) using stochastic quantization to convey an unbiased estimate of the reward.

$QuBan$ builds on the following observations. Recall that at time t the learner selects an action A_t and needs to convey the observed reward r_t. As we expect r_t to be close to the mean μ_{A_t}, we would like to use quantization levels that are dense around μ_{A_t} and sparse in other areas. Since μ_{A_t} is unknown, we estimate it using some function of the observed rewards that we term $\hat{\mu}(t)$; we can think of $\hat{\mu}(t)$ as specifying a "point" on the real line around which we want to provide denser quantization.

Figure 2: Illustration of $QuBan$. In the shown example, r_t is mapped to a value of the red dot (conveyed with the index $I_t = 4$), and stochastically to one of the two nearest quantization levels depicted on the red line.

4.1 Choices for $\hat{\mu}(t)$

In this work, we analyze the following three choices for $\hat{\mu}(t)$, the first two applying to MAB with a finite fixed set of arms, while the third to linear bandits.

• **Average arm point (Avg-arm-pt):** $\hat{\mu}(t) = \hat{\mu}_{A_t}(t-1)$. We use $\hat{\mu}_{A_t}(t-1)$, the average of the samples picked from arm A_t up to time $t-1$, as an estimate of μ_{A_t}.

• **Average point (Avg-pt):** $\hat{\mu}(t) = \frac{1}{t-1}\sum_{j=1}^{t-1} \hat{r}_j$ (the average over all observed rewards). Here we can think of $\frac{1}{t-1}\sum_{j=1}^{t-1} \hat{r}_j$ as an estimate of the mean of the best arm. Indeed, the average reward of a well behaved algorithm will converge to the best mean reward.

These two choices of $\hat{\mu}(t)$ give us flexibility to fit different regimes of MAB systems. In particular, we expect the avg-arm-pt to be a better choice for a small number of arms k and MAB algorithms that achieve good estimates of μ_{A_t} (explore all arms sufficiently so that $\hat{\mu}_{A_t}(t-1)$ approaches μ_{A_t}). However, as our analysis also shows (see Section 5), if k is large, acquiring good estimates for all arms may be costly and not what good algorithms necessarily pursue; instead, the avg-pt has a simpler implementation, as it only requires to keep track of a single number, and still enables to distinguish well in the neighborhood of the best arm, which is essentially what we mostly want.

• **Contextual bandit choice:** $\hat{\mu}(t) = \langle \theta_t, A_t \rangle$. Consider the widely used stochastic linear bandits model in Section 2. We observe that linear bandit algorithms, such as contextual Thomson sampling and LinUCB, choose a parameter θ_t believed to be close to the unknown parameter θ_*, and pick an action based on θ_t. For example, LinUCB [Dani et al., 2008] chooses a confidence set $\mathcal{C}_t$ with center θ_t believed to contain θ_* and picks an action $A_t = \arg\max_{a \in \mathcal{A}_t} \max_{\theta \in \mathcal{C}_t} \langle \theta, a \rangle$. Accordingly, we propose to use $\hat{\mu}(t) = \langle \theta_t, A_t \rangle$. We note that our intuition for the avg-pt choice does not work for contextual bandits as it relies on that $\max_{a \in \mathcal{A}_t} \langle \theta_*, a \rangle$ is the same for all t, which might not hold in general. Likewise, the avg-arm-pt choice will not work as the set of actions $\mathcal{A}_t$ can be infinite or change with time.

We underline that the estimator $\hat{\mu}(t)$ is only maintained at the learner's side and is broadcasted to the agents. As discussed before, this downlink communication is

---

Algorithm 1 Learner operation with input MAB algorithm Λ

1: Initialize: $\hat{\mu}(1) = 0$
2: **for** $t = 1, ..., n$ **do**
3: Choose an action A_t based on the bandit
4:   algorithm Λ and ask the next agent to play it
5: Send M_t[6], $\hat{\mu}(t)$ to an agent
6: Receive the encoded reward $(b_t, I_t, \mathcal{E}_{\mathcal{L}_t}(e_t))$ (see
7:   Algorithm 2)
8: **Decode $\hat{r}_t$:**
9: **if** length$(b_t) \le 4$ **then**
10:    $\hat{r}_t$ can be decoded using a lookup table
11: **else**
12:    Decode the sign, s_t, of r_t from b_t
13:    Set ℓ_t to be the I_t-th element in the set
14:    $\{0, 2^0, ...\}$
15:    Set $\mathcal{L}_t = \{\ell_t, \ell_t + 1, ..., \max\{2\ell_t, \ell_t + 1\}\}$
16:    Let $e_t^{(q)} = D_{\mathcal{L}_t}(\mathcal{E}_{\mathcal{L}_t}(e_t))$
17:    $\hat{r}_t = (s_t(e_t^{(q)} + \ell_t + 3.5) + 0.5 + \lfloor \hat{\mu}(t)/M_t \rfloor)M_t$
18: **Calculate $\hat{\mu}(t+1)$** (using one of the discussed
19:   choices)
20: **Update the parameters required by Λ**

---

not counted as communication cost.

4.2 $QuBan$ Components

At iteration t, $QuBan$ centers its quantization around the value $\hat{\mu}(t)$. It then quantizes the normalized reward $\bar{r}_t = r_t/M_t - \lfloor \hat{\mu}(t)/M_t \rfloor$ to one of the two values $\lfloor \bar{r}_t \rfloor, \lceil \bar{r}_t \rceil$, where $M_t = \epsilon\sigma X_t$[4], ϵ is a parameter to control the regret vs number of bits trade-off as will be illustrated later in this section, and $\{X_t\}_{i=1}^{n}$ are independent samples from a $\frac{1}{4}$-subgaussian distribution satisfying $|X_t| \ge 1$ almost surely, e.g., we can use $X_t = 1$ almost surely[5]. This introduces an error in estimating $\bar{r}_t$ that is bounded by 1, which results in error of at most M_t in estimating $r_t = M_t(\bar{r}_t + \lfloor \hat{\mu}(t)/M_t \rfloor)$. This quantization is done in a randomized way to convey an unbiased estimate of r_t.

Rounding of $\hat{\mu}(t)/M_t$: the reason for choosing the quantization to be centered around $\lfloor \hat{\mu}(t)/M_t \rfloor$ instead of $\hat{\mu}(t)/M_t$ is to guarantee that the distance between r_t and the two closest quantization levels is independent of $\hat{\mu}(t)$[7] (which is dependent on $\hat{r}_1, ..., \hat{r}_{t-1}$). As we discuss in the following section, this preserves the Markov

---

[4]The case where σ is unkown is discussed in App. A.

[5]For our proofs we set $X_t = 1$ for simplicity; more sophisticated choices can further improve the upper bounds as discussed in App. A.

[6]If X_t is chosesn to be 1, then sending M_t is not required.

[7]As will be shown in App. B, centering the quantization around any integer value implies that the two closest quantization levels to $\frac{r_t}{M_t}$ are $\lfloor \frac{r_t}{M_t} \rfloor, \lceil \frac{r_t}{M_t} \rceil$.

Osama A. Hanna, Lin F. Yang, Christina Fragouli

Algorithm 2 Distributed Agent Operation

1: **Inputs:** r_t, $\hat{\mu}(t)$ and M_t
2: Set $L = \{\lfloor \bar{r}_t \rfloor, \lceil \bar{r} \rceil\}$, $\hat{\bar{r}}_t = D_L(\mathcal{E}_L(\bar{r}_t))$
3: Set b_t with three bits to distinguish between the 8 cases: $\hat{\bar{r}}_t < -2, \hat{\bar{r}}_t > 3, \hat{\bar{r}}_t = i, i \in \{-1, 0, 1, 2\}$.
4: **if** $|\hat{\bar{r}}_t| > |a|$ and $\hat{\bar{r}}_t a > 0$, $a \in \{-2, 3\}$ **then**
5:     Augment b_t with an extra one bit to indicate if $|\hat{\bar{r}}_t| = |a| + 1$ or $|\hat{\bar{r}}_t| > |a| + 1$.
6:     **if** $|\hat{\bar{r}}_t| > |a| + 1$ **then**
7:         Let $L' = \{0, 2^0, ...\}$
8:         Set $\ell_t = \max\{j \in L | j \leq |\bar{r}_t| - |a|\}$
9:         Encode ℓ_t by $I_t - 1$ zeros followed by a one
10:         (unary coding), where I_t is the index of ℓ_t
11:         in the set L'.
12:         Let $e_t = |\bar{r}_t| - |a| - \ell_t$
13:         Set $\mathcal{L}_t = \{\ell_t, \ell_t + 1, ..., \max\{2\ell_t, \ell_t + 1\}\}$
14:         Encode e_t using SQ to get $\mathcal{E}_{\mathcal{L}_t}(e_t)$
15: Transmit $(b_t, I_t, \mathcal{E}_{\mathcal{L}_t}(e_t))$

---

property (given A_t, the quantized reward $\hat{r}_t$ is conditionally independent on the history $A_1, \hat{r}_1, ..., A_{t-1}, \hat{r}_{t-1}$), a property that is exploited in the analysis of bandit algorithms to guarantee that $|\sum_{t=1}^n \hat{r}_t - \mu_{A_t}/n|$ approaches zero in some probabilistic sense as n increases.

The precise learner and agent operations used for *QuBan* are presented in pseudo-code in Algorithms 1 and 2 (see Fig. 2 for an example), respectively, and discussed in detail in App. A, C. The learner at each iteration broadcasts $\hat{\mu}(t)$ and asks one of the agents available at time t to play an action A_t. Initially, since we have no knowledge about μ_i, the learner assumes that $\hat{\mu}(0) = 0$. The agent that plays the action uses the observed r_t together with $\hat{\mu}(t)$ it has received to transmit three values we term (b_t, I_t, e_t), to the learner, as described in Algorithm 2 using $O(\log(|\bar{r}_t|))$ bits.

5 Main Results

Our main results provide an upper and lower bound on the number of bits required to achieve the same unquantized regret up to a small constant factor. In particular, we show that 2.2 bits per reward are required to achieve a sub-linear regret and a quantization error, $\hat{r}_t - r_t$, that is $(\frac{\sigma}{2})^2$ subgaussian[8]. These conditions imply a regret within a factor of 1.5 from the unquantized regret. We also show that, on the average, 3.4 bits are sufficient to maintain a $(\frac{\sigma}{2})^2$-subgaussian quantization error, and achieve a regret within a factor of 1.5 from the unquantized regret. Before stating the results, we state our assumptions.

Assumption 1. *We assume that all the codes are prefix free [Cover, 1999][9] and that we are given:*
(i) a MAB instance with σ^2-subgaussian[10] rewards where the Markov property holds: conditioned on the action at time t, the current reward is conditionally independent on the history (past actions and rewards).
(ii) a MAB algorithm Λ such that for any instance with σ^2-subgaussian rewards, and time horizon n, the algorithm's expected regret (with unquantized rewards) is upper-bounded by R_n^U.

The following proposition gives an upper bound on the regret after quantization showing that for $\epsilon = 1$, the regret is within a factor of 1.5 from the regret of the unquantized case. The proof is provided in App. B.

Proposition 1. *Suppose Assumption 1 holds. Then, when we apply QuBan, the following hold:*
1. Conditioned on A_t, the quantized reward $\hat{r}_t$ is $((1 + \frac{\epsilon}{2})\sigma)^2$-subgaussian, conditionally independent on the history $A_1, \hat{r}_1, ..., A_{t-1}, \hat{r}_{t-1}$ (Markov property), and satisfies $\mathbb{E}[\hat{r}_t|A_t] = \mu_{A_t}$, $|\hat{r}_t - r_t| \leq M_t$ almost surely $(t = 1, ..., n)$.
2. The expected regret R_n is bounded as $R_n \leq (1 + \frac{\epsilon}{2})R_n^U$, where ϵ is a parameter to control the regret vs number of bits trade-off.

In the following we provide upper bound on the expected average number of bits. We also provides a high-probability upper bound on the instantaneous number of bits. For simplicity we only consider the case where $\epsilon = 1$ and discuss the other case in App. C. The proof is given in App. C.

Theorem 1. *Suppose Assumption 1 holds. Let $\epsilon = 1$. There is a universal constant C such that:*
1. For QuBan with $\hat{\mu}(t) = \hat{\mu}_{A_t}(t - 1)$ (avg-arm-pt), the average number of bits communicated satisfies that $\mathbb{E}[\bar{B}(n)] \leq 3.4 + (C/n)\sum_{i=1}^k \log(1 + |\mu_i|/\sigma) + C/\sqrt{n}$.
2. For QuBan with $\hat{\mu}(t) = \frac{1}{t-1}\sum_{j=1}^{t-1} \hat{r}_j$ (avg-pt), the average number of bits communicated satisfies $\mathbb{E}[\bar{B}(n)] \leq 3.4 + \frac{C}{n}\left(1 + \log(1 + \frac{|\mu^*|}{\sigma}) + \frac{R_n}{\sigma} + \sum_{t=1}^{n-1} \frac{R_t}{(\sigma t)}\right) + C/\sqrt{n}$.
3. For QuBan with $\hat{\mu}(t) = \langle \theta_t, A_t \rangle$ (stochastic linear bandit), the average number of bits communicated satisfies that $\mathbb{E}[\bar{B}](n) \leq 3.4 + C\mathbb{E}[\sum_{t=1}^n |\langle \theta_t - \theta_*, A_t \rangle|]/(\sigma n)$.

In App. C we also provide almost surely bounds on the asymptotic average number of bits, namely, $\lim_{n \to \infty}(1/n)\sum_{t=1}^n B_t \leq 3.4$ almost surely.

---

[8]The subgaussian condition is required for the standard analysis techniques of many algorithms, which is why we want to satisfy this property for the quantized rewards.

[9]A similar analysis can be carried out for non-singular codes [Cover, 1999].

[10]This is a standard assumption used for simplicity but is not required for our main results.

In the following we provide a high probability bound on the number of bits that *QuBan* uses in each iteration. We analyze the performance for avg-arm-pt only; the other choices for $\hat{\mu}(t)$ can be handled similarly.

Theorem 2. *For a MAB instance with σ^2-subgaussian rewards, QuBan with $\epsilon = 1, \hat{\mu}(t) = \hat{\mu}_{A_t}(t-1)$ (avg-arm-pt), satisfies that for t with $T_t(A_t) > 0$, where $T_t(i)$ is the number of pulls for arm i prior to iteration t, with probability at least $1 - \frac{1}{n}$ it holds that $\forall t \leq n$:*

$$B_t \leq 4 + \lceil \log(4\log(n)) \rceil + \lceil \log\log(4\log(n)) \rceil. \quad (5)$$

The proof is provided in App. D.

Remark 1. *Using the previous theorem we can modify QuBan to have that* (5) *is satisfied almost surely, by sending a random 1 bit when* (5) *is not satisfied. This will only add at most $n \sum_{i=1}^{k} \Delta_i$ regret with probability at most $\frac{1}{n}$. Hence, the expected regret is increased by at most a factor of 2.*

5.1 Lower Bound

In this subsection we provide a lower bound showing that an average number of 2.2 bits per iteration are required to maintain a sublinear regret and a $(\frac{\sigma}{2})^2$-subgaussian quantization error, $\hat{r}_t - r_t$. We also show that the instantaneous number of bits cannot be almost surely bounded by a constant.

Theorem 3. *For any memoryless algorithm that only uses quantized rewards, prefix-free encoding and satisfies that for any MAB instance with subgaussian rewards:*
i. R_n is sublinear in n,
ii. Conditioned on r_t, $\hat{r}_t - r_t$ is $(\frac{\sigma}{2})^2$-subgaussian $(t = 1, \ldots, n)$,
we have that there exist σ^2-subgaussian reward distributions for which:
1. $(\forall b \in \mathbb{N}) \ (\exists t, \delta > 0)$ such that $\mathbb{P}[B_t > b] > \delta$.
2. $(\forall t > 0) \ (\exists n > t)$ such that $\mathbb{E}[\bar{B}(n)] \geq 2.2$ bits.

Our lower bound is established by proving necessary properties for the set of quantization schemes that satisfy i, ii which include that $\mathbb{E}[\hat{r}_t|r_t] = r_t$ and that the distance between the quantization levels cannot be too large. We then show that 2.2 bits are needed to satisfy the proved properties for a Gaussian distribution. The proof is given in App. E.

5.2 Application to UCB, ϵ-greedy, and LinUCB

We here leverage Theorem 1 to derive bounds for three widely used MAB algorithms. We highlight that although the regret bounds hide constant factors, these constants are within 1.5 of the unquantized constants according to Theorem 1. The proofs are in App. F for Corollaries 2 and 3 and in App. G for Corollary 4.

Corollary 2. *Assume we use QuBan with avg-pt on top of **UCB** [Auer et al., 2002] with σ^2-subgaussian reward distributions and worst case gap Δ_i. Then there is a constant C that does not depend on n and k such that $R_n \leq C\sigma\sqrt{nk\log(n)}$, $\mathbb{E}[\bar{B}(n)] \leq 3.4 + C\sqrt{k\log(n)/n}$.*

Corollary 3. *Assume we use QuBan with avg-pt on top of ϵ-**greedy** [Auer et al., 2002] with σ^2-subgaussian reward distributions and constant gaps $\Delta_i \ \forall i$. Let $\epsilon_t = \min\{1, Ck/(t\Delta_{min}^2)\}$, where $\Delta_{min} = \min_i\{\Delta_i|\Delta_i > 0\}$ and $C > 0$ is a sufficiently large universal constant. Then there exists a constant C' that does not depend on n and k such that $R_n \leq C'\sigma k\log(1 + n/k)$, $\mathbb{E}[\bar{B}(n)] \leq 3.4 + C'(k\log^2(n)/n + 1/\sqrt{n})$.*

To simplify the expressions, we include the dependency on μ^* and Δ_i in the constant C for Corollary 2 and respectively C' for Corollary 3.

Corollary 4. *Assume we use QuBan on top of **Lin-UCB** [Dani et al., 2008], then there is a constant C that does not depend on n and d such that $R_n \leq Cd\sqrt{n}\log(n)$, $\mathbb{E}[\bar{B}(n)] \leq 3.4 + C\frac{d\log(n)}{\sqrt{n}}$.*

6 Numerical Evaluation

We here present representatives of our numerical results; additional plots are included in App. H.

Quantization Schemes. We compare *QuBan* against the baseline schemes described next.
Unquantized. Rewards are conveyed using the standard 32 bits representation.
r-bit SQ. We implement r-bit stochastic quantization, by using the quantizer described in Section 2, with 2^r levels uniformly dividing a range $[-\lambda, \lambda]$.
QuBan. We implement *QuBan* with $\epsilon = X_t = 1$.

MAB Algorithms. We use quantization on top of:
(i) the UCB implementation in [Lattimore and Szepesvári, 2020, chapter 8]. The UCB exploration constant is chosen to be σ_q, an estimate of the standard deviation of the quantized reward distribution.
(ii) the ϵ-greedy algorithm in [Lattimore and Szepesvári, 2020, chapter 6], where ϵ_t is set to be $\epsilon_t = \min\{1, \frac{C\sigma_q k}{t\Delta_{min}^2}\}$.
(iii) the LinUCB algorithm for stochastic linear bandits in [Lattimore and Szepesvári, 2020, chapter 19].

Synthetic Dataset:

MAB Setup. We simulate three cases. In each case we average over 10 runs of each experiment.
• **Setup 1: (Figs 3(a)).** We use $k = 100, \lambda = 100, C = 10$, the arms' means are picked from a Gaussian distribution with mean 0 and standard deviation 10 and the reward distributions are conditionally Gaussian
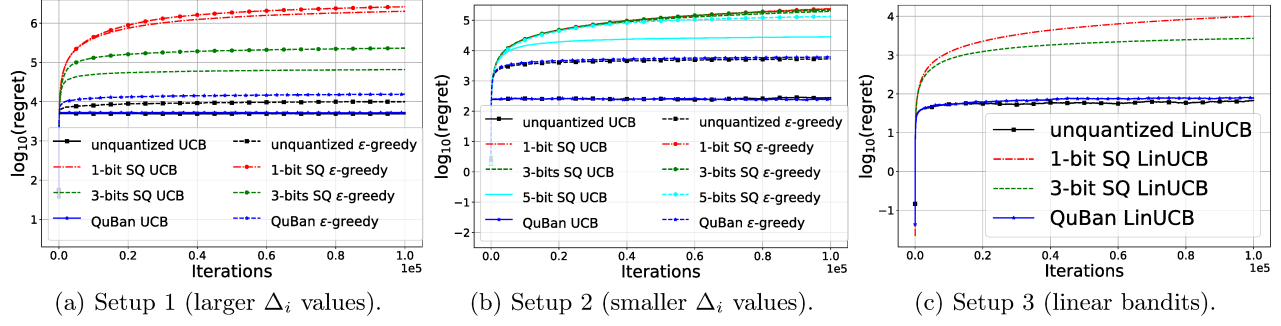
(a) Setup 1 (larger Δ_i values). (b) Setup 2 (smaller Δ_i values). (c) Setup 3 (linear bandits).

Figure 3: Regret versus number of iterations.

given the actions A_t with variance 0.1. The parameter σ_q is 0.1 for $QuBan$ and $200/2^r - 1$ for the r-bit SQ.

• **Setup 2: (Figs 3(b))** This differs from the previous only in that the means are picked from a Gaussian distribution with mean 95 and standard deviation 1 (leading to smaller Δ_i).

• **Setup 3: (Figs 3(c)).** This is our contextual bandit setup with parameters included in App. H.

Results. Fig. 3 plots the regret R'_n in (1) vs. the number of iterations. In App. H, we also plot the number of bits required to achieve a certain average regret. We find that:

• $QuBan$ in all three setups offers minimal or no regret increase compared to the unquantized rewards regret and achieves savings of tens of thousands of bits as compared to unquantized communication.

• Both $QuBan$ avg-pt and avg-arm-pt achieve the same regret (they are not distinguishable in Fig. 3 and thus we use a common legend), yet avg-arm-pt uses a smaller number of bits when the means of the arms tend to be well separated (Fig. 5(a) in App. H) while avg-pt uses a smaller number of bits when they tend to be closer together (Fig. 5(b) in App. H).

• 1-bit SQ significantly diverges in most cases; 3-bit and 5-bit SQ show better performance yet still not matching $QuBan$ with a performance gap that increases when the arms means are closer (Δ_i smaller), and hence, more difficult to distinguish.

• $QuBan$ in all three setups achieves $\mathbb{E}[\bar{B}(n)] \approx 3$ (plots are provided in App. H).

Cryptocurrency Returns Dataset:

MAB Setup. In this part we compare the performance of our scheme against 3-bit SQ using multiple cryptocurrencies prices from binance.com in October 2021, where the reward is the investment return. The action represents which cryptocurrency, among {Bitcoin, Ethereum, Dogecoin, and Litecoin}, to buy then sell on the next day. The return for each currency is samples uniformly at random from the daily returns in the month of October 2021.

Results. In Fig. 4 we plot the regret (daily return - optimal average return) versus the number of iterations
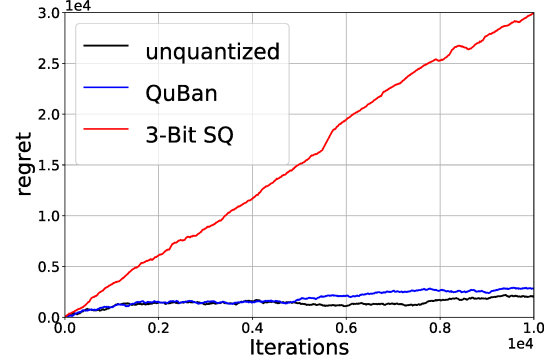

Figure 4: Regret versus number of iterations for the cryptocurrency prices dataset.

for the UCB algorithm using our quantization scheme and 3-bit SQ. We observe that the performance of $QuBan$ almost matches the unquantized performance (using ≈ 3 bits) while the regret of 3-bit SQ is linear for the used number of iterations.

7 Conclusion and Future Work

In this paper we provide a generic framework, $QuBan$, to quantize rewards for MAB problems. This framework can be used on top of nearly all the existing and future MAB algorithms, making them attractive for distributed learning applications where communication can become a bottleneck. We have demonstrated that, both in theory and by numerical experiments, $QuBan$ can provide very significant savings in terms of communication and barely affects the learning performance. We identify several future research directions: (1) How to exploit memory? In the setup we consider, the remote agents are changing over time, and thus they are essentially memoryless, i.e., a new agent does not know the history information of previous agents. (2) How to deal with heavy tailed noise? (3) How to convey contexts in the contextual bandit setting if these are not implicitly conveyed? Resolving such questions can offer additional benefits for communication-sensitive bandit learning setups.

References

Naoki Abe and Philip M Long. Associative reinforcement learning using linear probabilistic concepts. In *ICML*, pages 3–11. Citeseer, 1999.

Marc Abeille and Alessandro Lazaric. Linear thompson sampling revisited. In *Artificial Intelligence and Statistics*, pages 176–184. PMLR, 2017.

Rajeev Agrawal. Sample mean based index policies with o (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, pages 1054–1078, 1995.

Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*, pages 39–1. JMLR Workshop and Conference Proceedings, 2012.

Shipra Agrawal and Navin Goyal. Further optimal regret bounds for thompson sampling. In *Artificial intelligence and statistics*, pages 99–107. PMLR, 2013a.

Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135. PMLR, 2013b.

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

Animashree Anandkumar, Nithin Michael, Ao Kevin Tang, and Ananthram Swami. Distributed algorithms for learning and cognitive medium access with logarithmic regret. *IEEE Journal on Selected Areas in Communications*, 29(4):731–745, 2011.

Venkatachalam Anantharam, Pravin Varaiya, and Jean Walrand. Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-part i: Iid rewards. *IEEE Transactions on Automatic Control*, 32(11):968–976, 1987.

FJ Anscombe. Sequential medical trials. *Journal of the American Statistical Association*, 58(302):365–383, 1963.

Jean-Yves Audibert, Sébastien Bubeck, et al. Minimax policies for adversarial and stochastic bandits. In *COLT*, volume 7, pages 1–122, 2009.

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

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence.* Oxford university press, 2013.

Djallel Bouneffouf and Irina Rish. A survey on practical applications of multi-armed and contextual bandits. *arXiv preprint arXiv:1904.10040*, 2019.

Swapna Buccapatnam, Atilla Eryilmaz, and Ness B Shroff. Multi-armed bandits in the presence of side observations in social networks. In *52nd IEEE Conference on Decision and Control*, pages 7309–7314. IEEE, 2013.

Swapna Buccapatnam, Atilla Eryilmaz, and Ness B Shroff. Stochastic bandits with side observations on networks. In *The 2014 ACM international conference on Measurement and modeling of computer systems*, pages 289–300, 2014.

Thomas M Cover. *Elements of information theory.* John Wiley & Sons, 1999.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. *21st Annual Conference on Learning Theory*, pages 355–366, 2008.

Rémy Degenne and Vianney Perchet. Anytime optimal algorithms in stochastic multi-armed bandits. In *International Conference on Machine Learning*, pages 1587–1595. PMLR, 2016.

Kaize Ding, Jundong Li, and Huan Liu. Interactive anomaly detection on attributed networks. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 357–365, 2019.

Hossein Esfandiari, Amin Karbasi, Abbas Mehrabian, and Vahab Mirrokni. Regret bounds for batched bandits. *arXiv preprint arXiv:1910.04959*, 2019.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *International Conference on Computational Learning Theory*, pages 255–270. Springer, 2002.

Eyal Even-Dar, Shie Mannor, Yishay Mansour, and Sridhar Mahadevan. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(6), 2006.

Robert M Gray and Thomas G Stockham. Dithered quantizers. *IEEE Transactions on Information Theory*, 39(3):805–812, 1993.

Osama A Hanna, Yahya H Ezzeldin, Tara Sadjad-pour, Christina Fragouli, and Suhas Diggavi. On distributed quantization for classification. *IEEE Journal on Selected Areas in Information Theory*, 1 (1):237–249, 2020.

Michael N Katehakis and Herbert Robbins. Sequential choice from several populations. *Proceedings of the National Academy of Sciences of the United States of America*, 92(19):8584, 1995.

Tze Leung Lai. Adaptive treatment allocation and the multi-armed bandit problem. *The Annals of Statistics*, pages 1091–1114, 1987.

Peter Landgren et al. *Distributed Multi-agent Multi-armed Bandits*. PhD thesis, Princeton University, 2019.

John Langford and Tong Zhang. Epoch-greedy algorithm for multi-armed bandits with side information. *Advances in Neural Information Processing Systems (NIPS 2007)*, 20:1, 2007.

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

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

Shie Mannor and John N Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5(Jun): 623–648, 2004.

Jérémie Mary, Romaric Gaudel, and Philippe Preux. Bandits and recommender systems. In *International Workshop on Machine Learning, Optimization and Big Data*, pages 325–336. Springer, 2015.

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.

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, and Erik Snowberg. Batched bandit problems. *The Annals of Statistics*, 44(2):660–681, 2016.

Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

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

Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

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.

Linqi Song, Christina Fragouli, and Devavrat Shah. Recommender systems over wireless: Challenges and opportunities. In *2018 IEEE Information Theory Workshop (ITW)*, pages 1–5. IEEE, 2018.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

Daniel Vial, Sanjay Shakkottai, and R Srikant. One-bit feedback is sufficient for upper confidence bound policies. *arXiv preprint arXiv:2012.02876*, 2020.

Robert Wood, Radhika Nagpal, and Gu-Yeon Wei. Flight of the robobees. *Scientific American*, 308(3): 60–65, 2013.

A   Discussion On the Quantization Scheme

In this appendix we discuss some aspects of *QuBan*.

Description of the algorithm:

At iteration t, *QuBan* centers its quantization around the value $\hat{\mu}(t)$. It then quantizes the normalized reward $\bar{r}_t = r_t/M_t - \lfloor \hat{\mu}(t)/M_t \rfloor$ to one of the two values $\lfloor \bar{r}_t \rfloor, \lceil \bar{r}_t \rceil$, where $M_t = \epsilon \sigma X_t$, ϵ is a parameter to control the regret vs number of bits trade-off as will be illustrated later in this section, and $\{X_t\}_{i=1}^n$ are independent samples from a $\frac{1}{4}$-subgaussian distribution satisfying $|X_t| \geq 1$ almost surely, e.g., we can use $X_t = 1$ almost surely. If X_t is allowed to take larger values, it will make the quantization coarser with some probability, as will be described next, resulting in less number of bits. For example, if X_t has Gaussian tail, the average number of bits can be improved to ≈ 2.2 bits. The quantization introduces an error in estimating $\bar{r}_t$ that is bounded by 1, which results in error of at most M_t in estimating $r_t = M_t(\bar{r}_t + \lfloor \hat{\mu}(t)/M_t \rfloor)$. This quantization is done in a randomized way to convey an unbiased estimate of r_t. More precisely, the agent that pulls the arm at time t does the following operations. If $-3 \leq \bar{r}_t \leq 4$, the agent quantizes $\bar{r}_t$ to $\lceil \bar{r}_t \rceil$ with probability $\bar{r}_t - \lfloor (\rfloor \bar{r}_t)$, and to $\lfloor \bar{r}_t \rfloor$ with probability $\lceil \bar{r}_t \rceil - \bar{r}_t$. It then transmits 3 bits to distinguish between the 8 cases: **(a)** $-2 \leq \bar{r}_t \leq 3$ and $\hat{\bar{r}}_t = i)$, $i \in \{-2, .., 3\}$; **(b)** (a) does not hold and $\bar{r}_t > 0$; **(c)** (a) does not hold and $\bar{r}_t < 0$. If (b) holds, the agent transmits one bit to distinguish between the two cases: $\hat{\bar{r}}_t = 4, \bar{r}_t > 4$. If (c) holds, the agent transmits one bit to distinguish between the two cases: $\hat{\bar{r}}_t = -3, \bar{r}_t < -3$. If $\bar{r}_t > 4$ or $\bar{r}_t < -3$ the agent quantizes $\bar{r}' = |\bar{r}_t| - |a|$ as follows, where $a = 4$ if $\bar{r}_t > 4$, $a = -3$ if $\bar{r}_t < -3$. The agent conveys the greatest power of 2 below $\bar{r}'_t$, call it 2^{I_t} (an extra bit is transmitted to separate the case where $|\bar{r}_t| \leq 1$, and 2^{I_t} is assumed to be 0 in that case). Then, it quantizes $|\bar{r}_t| - 2^{I_t}$ using SQ with levels that are 1 distance apart in the interval $[0, 2^{I_t}]$[11] (see Fig. 2 for an example) (if $|\bar{r}_t| \leq 1$, SQ is used in the interval $[0, 1]$). The agent then transmits I_t with unary coding by transmitting I_t zeros followed by 1 one[12]. This uses $O(\log(\bar{r}_t))$ bits. The SQ output is transmitted using $O(\log(\bar{r}_t))$ bits. An estimated value of r_t is obtained from the quantized $\bar{r}_t$ by a proper shift and scaling. We recall that $\frac{\hat{\mu}(t)}{\sigma}$ is believed to be close to $\frac{r_t}{\sigma}$ in the majority of iterations resulting in small values for $\log(\bar{r}_t)$. Algorithms 1 and 2 describe the operation of *QuBan*.

Sending the least power of 2 below $\bar{r}_t$:

For simplicity we consider the case where $\bar{r}_t \geq 0$. We note that since it is possible for the decoded reward to take any value in the set $\{\lfloor \frac{\hat{\mu}}{\sigma} \rfloor, \lfloor \frac{\hat{\mu}}{\sigma} \rfloor + 1, \lfloor \frac{\hat{\mu}}{\sigma} \rfloor + 2...\}$ (to guarantee the uniform upper bound on $|\hat{r}_t - r_t|$), every value in that set needs to be encoded. A *good* encoding strategy assigns shorter codes to the levels that are close to $\lfloor \frac{\hat{\mu}}{\sigma} \rfloor$ as they are expected to occur more often. Hence, the best we can hope for is to encode r_t using $O(\log(\frac{r_t}{\sigma} - \lfloor \frac{\hat{\mu}}{\sigma} \rfloor))$ bits as it is quantized to either $\lfloor \frac{r_t}{\sigma} \rfloor$ or $\lceil \frac{r_t}{\sigma} \rceil$ and the quantization level at $\lfloor \frac{r_t}{\sigma} \rfloor$ is encoded using the largest number of bits among the levels in the set $\{\lfloor \frac{\hat{\mu}}{\sigma} \rfloor, \lfloor \frac{\hat{\mu}}{\sigma} \rfloor + 1, \lfloor \frac{\hat{\mu}}{\sigma} \rfloor + 2..., \lfloor \frac{r_t}{\sigma} \rfloor\}$. As can be seen in Appendix C, sending the greatest power of 2 below $\bar{r}_t$ then quantizing the difference using SQ gives that r_t is encoded using $O(\log(\frac{r_t}{\sigma} - \lfloor \frac{\hat{\mu}}{\sigma} \rfloor))$ bits. This is achieved since I_t is $O(\log(\frac{r_t}{\sigma} - \lfloor \frac{\hat{\mu}}{\sigma} \rfloor))$ and the SQ uses $2^{I_t} + 1$ quantization levels.

Alternatives to unary coding:

An alternative way to decode I_t is recursively applying our scheme by using unary coding to transmit the largest $I_t^{(2)}$ with $2^{I_t^{(2)}} \leq I_t$ and then encode the difference $I_t - 2^{I_t^{(2)}}$ using $\log(1 + 2^{I_t^{(2)}})$ bits noting that $I_t - 2^{I_t^{(2)}} \leq 2^{I_t^{(2)}}$. This results in using $O(\log(\log(\frac{r_t}{\sigma} - \lfloor \frac{\hat{\mu}}{\sigma} \rfloor)))$ bits to encode I_t. We keep the unary coding for I_t for simplicity and since it does not dominate the average number of bits.

Preserving regret bounds:

The main reasons *QuBan* preserves existing regret bounds is that it does not destroy the Markov property (as we prove in Appendix C) and it provides that $|\hat{r}_t - r_t|$ is uniformly upper bound. The later property implies that if given A_t, r_t is conditionally integrable, sub-exponential, sub-gaussian, or almost surely bounded, then given A_t, $\hat{r}_t$ is conditionally integrable, sub-exponential, sub-gaussian, or almost surely bounded respectively. A widely used assumption is that given A_t, r_t is conditionally sub-gaussian.

Unknown σ:

Throughout the paper, we assume a known upper bound on the noise variance. However, it is not difficult to see that a variance estimate within a constant factor would suffice. Running QuBan with an estimate σ' that is possibly different from the true σ results in a degradation in the regret by a factor of $\max\{1, \frac{\sigma'}{\sigma}\}$ and increase in

---

[11]Note that $0 \leq |\bar{r}_t| - 2^{I_t} \leq 2^{I_t}$.

[12]Alternative (to unary) coding techniques that can result in a smaller number of bits are discussed later in this appendix.

the communication by $2\log(\frac{\sigma}{\sigma'})$ bits. An optimistic estimate of the noise $\sigma' < \sigma$ results in finer quantization, hence, no degradation in the regret at the cost of increasing the communication by $2\log(\frac{\sigma}{\sigma'})$ bits.

B   Proof of Proposition 1

Proof. We start by proving that $\hat{r}_t$ is an unbiased estimate of μ_{A_t}. If $-3 \le r_t \le -4$, we have that $\hat{r}_t$ takes the value $\lceil r_t \rceil$ with probability $r_t - \lfloor r_t \rfloor$, and the value $\lfloor r_t \rfloor$ with probability $\lceil r_t \rceil - r_t$. Hence, $\mathbb{E}[\hat{r}_t|r_t] = r_t$. For all the other cases we have that

$$
\begin{aligned}
\mathbb{E}[\hat{r}_t|r_t] &= \mathbb{E}[M_t(s_t e_t^{(q)} + \lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor + s_t \ell_t)|r_t] \\
&= \mathbb{E}[M_t \mathbb{E}[s_t e_t^{(q)} + \lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor + s_t \ell_t|r_t, \hat{\mu}(t), M_t]|r_t] \\
&\overset{(i)}{=} \mathbb{E}[M_t(\frac{r_t}{M_t} - (\lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor + s\ell_t) + \lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor + s\ell_t)|r_t] \\
&= r_t,
\end{aligned}
\tag{6}
$$

where (i) follows from the fact that the stochastic quantization (SQ) that we use gives an unbiased estimate of the input. Hence, in all cases we have that

$$
\mathbb{E}[\hat{r}_t|A_t] = \mathbb{E}[\mathbb{E}[\hat{r}_t|r_t, A_t]|A_t] = \mathbb{E}[\mathbb{E}[\hat{r}_t|r_t]|A_t] = \mathbb{E}[r_t|A_t] = \mu_{A_t}
\tag{7}
$$

The bound on $|r_t - \hat{r}_t|$ follows from the fact that the distance between the quantization levels for which we use the randomized quantization is 1, hence, in all cases we have that

$$
1 \ge |s_t e_t^{(q)} - (\frac{r_t}{M_t} - \lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor - s_t \ell_t)| = \frac{|\hat{r}_t - r_t|}{M_t}.
\tag{8}
$$

We note that this implies

$$
\begin{aligned}
\mathbb{E}[|\hat{r}_t - \mu_{A_t}|^2|A_t] &= \mathbb{E}[|\hat{r}_t - r_t + r_t - \mu_{A_t}|^2|A_t] \\
&= \mathbb{E}[|\hat{r}_t - r_t|^2|A_t] + \mathbb{E}[|r_t - \mu_{A_t}|^2|A_t] + 2\mathbb{E}[(r_t - \mu_{A_t})(\hat{r}_t - r_t)|A_t] \\
&\le (1 + \epsilon^2)\sigma^2 + 2\mathbb{E}[(r_t - \mu_{A_t})\mathbb{E}[(\hat{r}_t - r_t)|A_t, r_t]|A_t] \\
&= (1 + \epsilon^2)\sigma^2.
\end{aligned}
\tag{9}
$$

To see that conditioned on A_t, $\hat{r}_t$ is conditionally independent on the history $A_1, \hat{r}_1, ..., A_{t-1}, \hat{r}_{t-1}$, we notice that since we replace $\frac{\hat{\mu}(t)}{M_t}$ by an integer, $\lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor$ and since the distance between the quantization levels is 1, we have that the two nearest quantization levels to $\frac{r_t}{M_t}$ are at $\lfloor \frac{r_t}{M_t} \rfloor, \lceil \frac{r_t}{M_t} \rceil$. Hence, conditioned on M_t, $\hat{r}_t$ takes the value $M_t \lceil \frac{r_t}{M_t} \rceil$ with probability $\frac{r_t}{M_t} - \lfloor \frac{r_t}{M_t} \rfloor$, and the value $M_t \lfloor \frac{r_t}{M_t} \rfloor$ with probability $\lceil \frac{r_t}{M_t} \rceil - \frac{r_t}{M_t}$. This shows that despite the fact that the encoding of $\hat{r}_t$ is a function of $r_1, ..., r_t$, the value of $\hat{r}_t$ is a function of r_t only, since M_t is generated independently of the history. As a result, given A_t, $\hat{r}_t$ is conditionally independent on the history $A_1, \hat{r}_1, ..., A_{t-1}, \hat{r}_{t-1}$.

The fact that $\hat{r}_t$ is subgaussian can be proven by Cauchy-Schwarz

$$
\begin{aligned}
\mathbb{E}[e^{\lambda(\hat{r}_t - \mu_{A_t})}|A_t] &= \mathbb{E}[e^{\lambda(\hat{r}_t - r_t + r_t - \mu_{A_t})}|A_t] \\
&\le \mathbb{E}[e^{p\lambda(\hat{r}_t - r_t)}|A_t]^{\frac{1}{p}} \mathbb{E}[e^{(1-p)\lambda(r_t - \mu_{A_t})}|A_t]^{\frac{1}{1-p}} \\
&\le e^{\lambda^2 \frac{\sigma^2(1+\frac{\epsilon}{2})^2}{2}},
\end{aligned}
\tag{10}
$$

where $p = 1 + \frac{2}{\epsilon}$. To bound the expected regret after quantization we observe that $R_n = \sum_{t=1}^{n} \mathbb{E}(\mu_t^* - r_t) = \sum_{t=1}^{n} \mathbb{E}(\mu_t^* - \hat{r}_t) = (1 + \frac{\epsilon}{2}) \sum_{t=1}^{n} \mathbb{E}(\frac{\mu_t^* - \hat{r}_t}{1 + \frac{\epsilon}{2}})$. We have that $\frac{\hat{r}_t}{(1+\frac{\epsilon}{2})}$ is σ^2-subgaussian. Applying the bandit algorithm using $\frac{\hat{r}_t}{(1+\frac{\epsilon}{2})}$ results in $\sum_{t=1}^{n} \mathbb{E}(\frac{\mu_t^* - \hat{r}_t}{(1+\frac{\epsilon}{2})}) \le R_n^U(\{\Delta_i/(1 + \frac{\epsilon}{2})\})$, hence

$$
R_n \le (1 + \frac{\epsilon}{2}) R_n^U(\{\Delta_i/(1 + \frac{\epsilon}{2})\}).
\tag{11}
$$

$\square$

C   Proof of Theorem 1

Proof. We have that B_t can be bounded as

$$B_t \leq 3 + \mathbf{1}[\frac{r_t}{M_t} - \lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor > 3] + \mathbf{1}[\lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor - \frac{r_t}{M_t} > 2]$$
$$+ 2(\mathbf{1}[\frac{r_t}{M_t} - \lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor > 4]\lceil \log(\frac{r_t}{M_t} - \lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor - 3)\rceil)$$
$$+ 2(\mathbf{1}[\lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor - \frac{r_t}{M_t} > 3]\lceil \log(\lfloor \frac{\hat{\mu}(t)}{M_t} \rfloor - \frac{r_t}{M_t} - 2)\rceil)$$
$$\leq 3 + \mathbf{1}[|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}| > 2] + 2(\mathbf{1}[|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}| > 3])$$
$$+ 2(\mathbf{1}[|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}| > 3]\log(|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}| - 2)). \tag{12}$$

Hence for each $\delta > 0$, we have

$$B_t \leq 3 + \mathbf{1}[|\frac{r_t - \mu_{A_t}}{\sigma}| > 2(1-\delta)] + \mathbf{1}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| > 2\delta]$$
$$+ 2(\mathbf{1}[|\frac{r_t - \mu_{A_t}}{\sigma}| > 3(1-\delta)] + \mathbf{1}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| > 3\delta])$$
$$+ 2(\mathbf{1}[|\frac{r_t - \mu_{A_t}}{\sigma}| > 3])\log(|\frac{r_t - \hat{\mu}(t)}{\sigma}| - 2). \tag{13}$$

Taking the expectation of both sides, we get that

$$\mathbb{E}[B_t] \leq 3 + \mathbb{P}[|\frac{r_t - \mu_{A_t}}{\sigma}| > 2(1-\delta)] + \mathbb{P}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| > 2\delta]$$
$$+ 2(\mathbb{P}[|\frac{r_t - \mu_{A_t}}{\sigma}| > 3(1-\delta)] + \mathbb{P}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| > 3\delta])$$
$$+ 2\mathbb{E}[(\mathbf{1}[|\frac{r_t - \mu_{A_t}}{\sigma}| > 3])\log(|\frac{r_t - \hat{\mu}(t)}{\sigma}| - 2)]. \tag{14}$$

Hence, there are universal constants C, C' such that

$$\mathbb{E}[B_t] \leq 3.32 + C'\mathbb{E}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}|] + 2\mathbb{E}[\mathbf{1}[|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}| > 3](|\frac{r_t}{M_t} - \frac{\hat{\mu}(t)}{M_t}| - 3)]$$
$$\leq 3.32 + C'\mathbb{E}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}|] + 2\mathbb{E}[\mathbf{1}[|\frac{r_t - \mu_{A_t}}{\sigma}| > 3(1-\delta)]|\frac{r_t - \mu_{A_t}}{\sigma}| - 3|]$$
$$+ 2\mathbb{E}[\mathbf{1}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| > 3\delta]|\frac{r_t - \mu_{A_t}}{\sigma}| - 3|] + 2\mathbb{E}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}|]$$
$$\leq 3.32 + (C' + 2)\mathbb{E}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}|] + 2\sum_{i=3}^{\infty}|i(1-\delta) - 3|\mathbb{P}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| > i(1-\delta)]$$
$$+ 2\mathbb{E}[\mathbf{1}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| > 3\delta]]\mathbb{E}[|\frac{r_t - \mu_{A_t}}{\sigma}| - 3|] + 2\mathbb{E}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}|]$$
$$\leq 3.4 + C\mathbb{E}[|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}|] \tag{15}$$

From (13), $\mathbb{E}[|r_t - \mu_{A_t}|^2|A_t] \leq \sigma^2$, Markov property and the strong law of large numbers for martingales, we also have that there is a universal constant C such that

$$\lim_{n\to\infty} \frac{1}{n}\sum_{t=1}^{n} B_t \leq 3.4 + \lim_{n\to\infty} \frac{C}{n}\sum_{t=1}^{n}|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}| \quad \text{almost surely.} \tag{16}$$

It then remains to analyze $|\mu_{A_t} - \hat{\mu}(t)|$ for the three proposed choices of $\hat{\mu}(t)$.

- **avg-pt** $(\hat{\mu}(t) = \frac{1}{t-1}\sum_{j=1}^{t-1}\hat{r}_j)$:

We have that for $t > 1$

$$
\begin{aligned}
\frac{|\mu_{A_t} - \hat{\mu}(t)|}{\sigma} &\leq \frac{|\mu_{A_t} - \mu^*|}{\sigma} + \frac{|\mu^* - \hat{\mu}(t)|}{\sigma} \\
&= \frac{\Delta_{A_t}}{\sigma} + |\frac{\sum_{j=1}^{t-1}\mu^* - \mu_{A_j} + \mu_{A_j} - \hat{r}_j}{(t-1)\sigma}| \\
&\leq \frac{\Delta_{A_t}}{\sigma} + |\frac{\sum_{j=1}^{t-1}\mu^* - \mu_{A_j}}{(t-1)\sigma}| + |\frac{\sum_{j=1}^{t-1}\mu_{A_j} - \hat{r}_j}{(t-1)\sigma}| \\
&= \frac{\Delta_{A_t}}{\sigma} + \frac{\sum_{i=1}^{k}\Delta_i T_i(t-1)}{(t-1)\sigma} + |\frac{\sum_{j=1}^{t-1}\mu_{A_j} - \hat{r}_j}{(t-1)\sigma}|.
\end{aligned}
\tag{17}
$$

For $t = 1$ we have

$$
\begin{aligned}
\frac{|\mu_{A_t} - \hat{\mu}(t)|}{\sigma} &\leq \frac{|\mu_{A_t} - \mu^*|}{\sigma} + \frac{|\mu^* - \hat{\mu}(t)|}{\sigma} \\
&= \frac{\Delta_{A_1}}{\sigma} + \frac{|\mu^*|}{\sigma}.
\end{aligned}
\tag{18}
$$

We then have that

$$
\begin{aligned}
\frac{1}{n}\sum_{t=1}^{n}\log(1+|\frac{\mu_{A_t} - \hat{\mu}(t)}{\sigma}|) &\leq \frac{\log(1+\frac{|\mu^*|}{\sigma})}{n\sigma} + \frac{1}{n}\sum_{t=1}^{n}\log(1+\frac{\Delta_{A_t}}{\sigma}) \\
&+ \frac{1}{n}\sum_{t=2}^{n}\log(1+\frac{\sum_{i=1}^{k}\Delta_i T_i(t-1)}{(t-1)\sigma}) + \log(1+|\frac{\sum_{j=1}^{t-1}\mu_{A_j} - \hat{r}_j}{(t-1)\sigma}|) \\
&\leq \frac{\log(1+\frac{|\mu^*|}{\sigma})}{n\sigma} + \frac{1}{n}\left(\frac{\sum_{i=1}^{k}\Delta_i T_i(n)}{\sigma} + \sum_{t=1}^{n-1}\frac{\sum_{i=1}^{k}\Delta_i T_i(t)}{t\sigma} + |\frac{\sum_{j=1}^{t}\mu_{A_j} - \hat{r}_j}{t\sigma}|\right).
\end{aligned}
\tag{19}
$$

We have that since $\mathbb{E}[|r_t - \mu_{A_t}|^2|A_t] \leq \sigma^2$, and Markov property, then by the strong law of large numbers for martingales $\lim_{t\to\infty}\frac{\sum_{j=1}^{t-1}\mu_{A_j} - \hat{r}_j}{(t-1)\sigma} = 0$ almost surely. We then have that if the limit of average regret is 0 almost surely (or in probability), then from (16) and (19) we get that

$$
\lim_{n\to\infty}\frac{1}{n}\sum_{t=1}^{n}B_t \leq 3.4 \quad \text{almost surely (or in probability).}
\tag{20}
$$

By observing that we can generate a long sequence of rewards from each arm before the process starts and since $\mathbb{E}[|r_t - \mu_{A_t}|^2|A_t] \leq \sigma^2$, then by the triangle inequality we have that

$$
\begin{aligned}
\frac{1}{n}\sum_{t=1}^{n}\mathbb{E}[|\frac{\sum_{j=1}^{t-1}\mu_{A_j} - \hat{r}_j}{(t-1)\sigma}|] &\overset{(i)}{\leq} \frac{2}{n}\sum_{t=1}^{n}\frac{1}{\sqrt{t}} \\
&= \frac{2}{n}\sum_{t=1}^{n}\frac{1}{\sqrt{t}} \\
&\leq \frac{2}{n}(1+\int_{t=1}^{n}\frac{1}{\sqrt{t}}dt) \\
&\leq \frac{4}{\sqrt{n}},
\end{aligned}
\tag{21}
$$

where (i) follows from the fact that $\mu_{A_j} - \hat{r}_j, \mu_{A_i} - \hat{r}_i$ are uncorrelated for all $i < j$ since

$$
\mathbb{E}[(\mu_{A_j} - \hat{r}_j)(\mu_{A_i} - \hat{r}_i)] = \mathbb{E}[\mathbb{E}[(\mu_{A_j} - \hat{r}_j)(\mu_{A_i} - \hat{r}_i)|A_j, A_i, \hat{r}_i]] = 0.
\tag{22}
$$

We conclude that there is a universal constant C such that

$$\hat{B}(n) \leq 3.4 + (C/n)\left(1 + \log(1 + |\mu^*|/\sigma) + R_n/\sigma + \sum_{t=1}^{n-1} R_t/(\sigma t)\right) + C/\sqrt{n} \tag{23}$$

• **avg-arm-pt $(\hat{\mu}(t) = \hat{\mu}_{A_t}(t-1))$:**
We have that for $T_{A_t}(t-1) > 0$

$$\frac{|\mu_{A_t} - \hat{\mu}(t)|}{\sigma} = |\frac{\sum_{j=1}^{t-1}(\mu_{A_t} - \hat{r}_j)\mathbf{1}(A_j = A_t)}{T_{A_t}(t-1)\sigma}|. \tag{24}$$

For $T_{A_t}(t-1) = 0$, we have that $\hat{\mu}(t) = 0$. Then

$$\frac{1}{n}\sum_{t=1}^{n} \mathbb{E}[\log(1 + \frac{|\mu_{A_t} - \hat{\mu}(t)|}{\sigma})] \overset{(i)}{\leq} \frac{1}{n}\sum_{i=1}^{k} \log(1 + \frac{|\mu_i|}{\sigma}) + \frac{2}{n}\sum_{t=1}^{n} \frac{1}{\sqrt{t}}$$

$$\overset{(ii)}{\leq} \frac{1}{n}\sum_{i=1}^{k} \log(1 + \frac{|\mu_i|}{\sigma}) + \frac{4}{\sqrt{n}} \tag{25}$$

where (ii) is as in (21), and (i) can be seen by observing that we can generate a long sequence of rewards from each arm before the process starts, from the fact that $\hat{r}_j - \mu_{A_j}, \hat{r}_i - \mu_{A_i}$ are uncorrelated for all $i \neq j$ and since $\mathbb{E}[|r_t - \mu_{A_t}|^2 | A_t] \leq \sigma^2$.

We conclude that there is a universal constant C such that

$$\hat{B}(n) \leq 3.4 + \frac{C}{n}\sum_{i=1}^{k} \log(1 + \frac{|\mu_i|}{\sigma}) + \frac{C}{\sqrt{n}}. \tag{26}$$

The fact that $\lim_{n\to\infty} \frac{1}{n}\sum_{t=1}^{n} B_t \leq 3.4$ almost surely, can be seen using the strong law of large numbers by observing that we can generate a long sequence of rewards from each arm before the process starts, the number of arms is finite, and if $\lim_{n\to\infty} T_i(n) < \infty$ then the contribution of arm i in the number of bits decays to zero almost surely as $n \to \infty$.

• **stochastic linear bandits $(\hat{\mu}(t) = \langle \theta_t, A_t \rangle)$:**
The results follow directly from (13), (15), (16) and choice of $\hat{\mu}(t)$.

For the case where $\epsilon \neq 1$, it is easy to see that for small values of ϵ, the number of transmitted bits increases by $2\log(\frac{1}{\epsilon})$ bits. This can be further decreased to $\log(\frac{1}{\epsilon}) + \log(\log(\frac{1}{\epsilon}))$ bits using the encoding in App. A.

$\square$

D Proof of the High Probability Bound

From App.A, we have that

$$B_t \leq 3 + \mathbf{1}[\frac{r_t}{M_t} - \lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor > 3] + \mathbf{1}[\lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor - \frac{r_t}{M_t} > 2]$$

$$+ \mathbf{1}[\frac{r_t}{M_t} - \lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor > 4]\left(\lceil\log(\frac{r_t}{M_t} - \lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor - 3)\rceil + \lceil\log(\log(\frac{r_t}{M_t} - \lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor - 3))\rceil\right)$$

$$+ \mathbf{1}[\lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor - \frac{r_t}{M_t} > 3]\left(\lceil\log(\lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor - \frac{r_t}{M_t} - 2)\rceil + \lceil\log(\log(\lfloor\frac{\hat{\mu}(t)}{M_t}\rfloor - \frac{r_t}{M_t} - 2))\rceil\right)$$

$$\leq 4 + \log(\frac{\hat{\mu}(t)}{\sigma} - \frac{r_t}{\sigma} - 2) + \log(\log(\frac{\hat{\mu}(t)}{\sigma} - \frac{r_t}{\sigma} - 2)). \tag{27}$$

Let the event G be that $\forall t \in \{1, ..., n\} : |r_t - \mu_{A_t}| \leq \sigma\sqrt{4\log(n)}$. From the subgaussian assumption and applying the union bound we have that

$$\mathbb{P}[G] > 1 - \sum_{t=1}^{n} e^{-2\log(n)}. \tag{28}$$

We have that if G holds then for t with $T_t(A_t) > 0$, we have that $|\hat{\mu}(t) - \mu_{A_t}| \leq \sigma, |r_t - \mu_{A_t}| \leq \sigma$. Hence, $|\hat{\mu}(t) - r_t| \leq 2\sigma$. Substituting in (27), we get the desired result.

E   Proof of Lower Bound (Theorem 3)

Proof. To simplify notations, we omit the time index t and only mention it when it is necessary. Normalizing the rewards by σ, it suffices to consider the case where $\sigma = 1$. We will also consider quantization schemes where the set of quantization boundaries is deterministic and fixed over time. The case where the set of quantization boundaries is random follows by similar arguments, while the case where the set of quantization boundaries changes over time follows by observing that the proved properties will hold except for a sublinear number of iterations. We start by showing that to satisfy (i), it suffices to consider unbiased quantization schemes, i.e., quantization schemes that satisfy:

(iii) $\mathbb{E}[\hat{r}|r] = r$.

Let P, P' denote reward distributions with means μ_1, μ_2 respectively. We have that, for any given algorithm, either: **Case 1**: $\forall P, P'$ with $\mu_1 \neq \mu_2$, we have that $\mathbb{E}_P[\hat{r}] \neq \mathbb{E}_{P'}[\hat{r}]$; or **Case 2**: $\exists P, P'$ with $\mu_1 \neq \mu_2$, and $\mathbb{E}_P[\hat{r}] = \mathbb{E}_{P'}[\hat{r}]$.

For any algorithm that satisfies the property of **Case 1**, we will show that we can create another algorithm that satisfies (iii) and achieves the same performance as the original algorithm. In particular, we will prove that, for any algorithm satisfying Case 1, the function that maps x_i to $\mathbb{E}_{P_{x_i}}[x_i]$ has to be of the form $\mathbb{E}_{P_{x_i}}[x_i] = c_1 x_i + c_2$ for some constants c_1, c_2, and thus, by a proper shift and scaling, the quantization algorithm can be modified to be unbiased without affecting the number of transmitted bits or the performance. To do so, we first construct distributions P and P' as follows. Consider the set of distributions $\{P_x\}_{x \in \mathbb{R}}$, where P_x represents the random variable that takes the value x almost surely. Let $\{x_i, p_i, p'_i\}_{i=1}^3$ be real values such that $x_i \neq x_j \forall i \neq j$, $\sum_{i=1}^3 p_i = \sum_{i=1}^3 p'_i = 1$ and $p_i, p'_i \geq 0 \forall i \in \{1, 2, 3\}$. We design P to be the distribution of a random variable that takes the value x_i with probability p_i, and P' be the distribution of a random variable that takes the value x_i with probability p'_i.

Note that, for the condition of Case 1 to be satisfied, we want to have that $\mathbb{E}_P[\hat{r}] = \mathbb{E}_{P'}[\hat{r}]$ *only if* $\sum_{i=1}^3 p_i x_i = \sum_{i=1}^3 p'_i x_i$. Hence, we need $\sum_{i=1}^3 (p_i - p'_i) \mathbb{E}_{P_{x_i}}[x_i] = 0$ *only if* $\sum_{i=1}^3 (p_i - p'_i) x_i = 0$. This implies that the null space of the matrix $[\mathbb{E}_{P_{x_1}}[x_1], \mathbb{E}_{P_{x_2}}[x_2], \mathbb{E}_{P_{x_3}}[x_3]; 1, 1, 1]$ is the same as the null space of the matrix $[x_1, x_2, x_3; 1, 1, 1]$. Equivalently, the function that maps x_i to $\mathbb{E}_{P_{x_i}}[x_i]$ is a linear function. By replacing x_3 with arbitrary x, it is easy to see that the function that maps x_i to $\mathbb{E}_{P_{x_i}}[x_i]$ is the same for all chosen values of $\{x_i\}$. This completes the proof in this case.

For **Case 2**, if we consider a MAB instance with two arms with distributions P, P' that witness the property in **Case 2**, then even if we have infinite samples from the quantization scheme we cannot achieve better than $O(|\mu_1 - \mu_2|n)$ regret.

This shows that it suffices to consider quantization schemes that satisfy (iii). We also note that: (iv) for any $\delta > 0$, the maximum distance between any consecutive quantization levels cannot exceed $1 + \delta$, lest there is a reward r, that is in the middle of the two quantization levels, mapped to $\hat{r}$ with $|\hat{r} - r| \geq \frac{1+\delta}{2}$ which violates (ii).

We are now ready to prove 1. Suppose towards a contradiction that $\exists b, t$ such that $\mathbb{P}[B_n \leq b] = 1 \forall n > t$. Pick n arbitrary large, we have that b can describe at most 2^b quantization levels. As shown previously, we have that the maximum distance between quantization levels is bounded by a value $1 + \delta$ for any $\delta > 0$. Hence, either the interval $(-\infty, -2(2^b + 1)]$ or the interval $[2(2^b + 1), \infty)$ will have no quantization levels. We assume without loss of generality that the interval $[2(2^b + 1), \infty)$ has no quantization levels. Hence, all the values in that interval will be mapped to values in $(-\infty, 2(2^b + 1))$. If the distribution of the reward is Gaussian for example, then the interval $(-\infty, 2(2^b + 1))$ will have non-zero probability. This contradicts (iii).

To prove 2 we consider a gaussain reward distribution with zero mean and unit variance. We start by showing that to minimize the expected number of bits while satisfying i, ii, iii, the distance between the quantization levels need to be 1 with one quantization level at 0. Let $a \in \mathcal{Q}$ be the quantization level that is represented with the least number of bits, where $\mathcal{Q}$ is the set of quantization levels. We note that to minimize the average number of bits, we need to maximize the probability of transmitting a quantization level represented by a small number of bits. Pick $r > a$ such that $r - a < 1$, and $Q^+ = \{q \in \mathcal{Q} | q > r\}$, $Q^- = \{q \in \mathcal{Q} | q \leq r, q \neq a\}$. To satisfy (iii) we need $r = \mathbb{E}[\hat{r}|r] = a p_a + \sum_{q \in \mathcal{Q}} q p_q$, where p_q is the probability of quantizing r to q. Hence,

$$p_a = \frac{\sum_{q \in \mathcal{Q}^+} (q - r) p_q - \sum_{q \in \mathcal{Q}^-} (r - q) p_q}{r - a}. \tag{29}$$

Thus, to maximize p_a, we maximize q for $q \in \mathcal{Q}^+$ and choose $p_q = 0$ for $q \in \mathcal{Q}^-$. From (iv) this happens when

the distance between the quantization levels in $\mathcal{Q}^+$ is 1. Applying the same argument for $r < a$, we get that to maximize p_a, we need the distance between the quantization levels in $\mathcal{Q}^-$ to be 1. Moreover, due to (ii) the optimal choice for a is 0, lest from the fact that the density of a Gaussian distribution increases as we approach the mean, we can replace values that are mapped to a with higher probabilities with values that are closer to 0 (hence having higher densities), thus increasing p_a. Applying the same logic, to optimize the probability of a', the quantization level represented by the second minimum number of bits, we require the distance between the quantization levels to be 1 and $a' = 1$. The same logic can be applied for the quantization level represented by the k-th minimum number of bits showing that the optimal set of quantization levels is the set of integer numbers.

Pick $r \in \mathbb{R}$, and let $p_z^{(r)}$ be the probability of quantizing r to z. Let p_z denote the probability of transmitting the quantization level at z. Then, $p_z = \int_{-\infty}^{\infty} p_z^{(x)} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx$. By observing that for $x > \frac{z}{2}$, $e^{-x^2/2}$ is decays at least as $e^{-z^2/8}$, while for $x \leq \frac{z}{2}$ we have that $p_z^{(x)}$ decays at least as $e^{-z^2/8}$ due to (ii), we get that p_z is exponentially decaying in z. Hence, the optimal prefix-free encoding for levels is to assign 1 bit for the quantized value 0, 2 bits for the value 1, 3 bits for the value -1 and so on [Cover, 1999]. Note that in this proof/theorem, as well as in the upper bound, we consider prefix-free codes, yet very similar arguments provide arguments for non-prefix free codes as well.

By solving (iii) and $\sum_{z=-\infty}^{\infty} p_z^{(r)} = 1$ together we get that $p_{\lfloor r \rfloor}^{(r)} = \lceil r \rceil - r + \sum_{i=2}^{\infty}(i-1)p_{\lceil r \rceil + i}^{(r)} - \sum_{i=2}^{\infty}(i-1)p_{\lfloor r \rfloor - i}^{(r)}$. Due to (ii) we can bound the quantization levels' probabilities as

$$P_i \geq \int_{i-1}^{i} (|x - i| - \sum_{i=1}^{\infty} i e^{-2(\lceil x \rceil + i)^2}) \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx + \int_{i}^{i+1} (|x - i| - \sum_{i=1}^{\infty} i e^{-2(\lfloor x \rfloor - i)^2}) \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx. \quad (30)$$

By computing this bound for $p_{-3}, ...p_3$, we get that average number of bits lower bounded by 2.5 bits.

$\square$

F   Proofs of Corollaries 2, 3

The expected regret bounds follow directly from Theorem 1. To bound the average number of bits used for the avg-pt, we only need to bound the decay rate of $\frac{1}{n}\sum_{t=1}^{n-1}\frac{R_t}{\sigma t}$.

Corollary 2:
From Theorem 1 and [Auer et al., 2002], we have that for $QuBan$ with UCB, there is a constant C such that $R_n \leq C\sigma\sqrt{kn\log(n)}$. Then,

$$\frac{1}{n}\sum_{t=1}^{n-1}\frac{R_t}{\sigma t} \leq C\frac{1}{n}\sum_{t=1}^{n}\frac{\sqrt{kt\log(t)}}{t}$$
$$\leq \frac{C\sqrt{k\log(n)}}{n}\sum_{t=1}^{n}\frac{1}{\sqrt{t}}$$
$$\leq \frac{C\sqrt{k\log(n)}}{n}(1 + \int_{t=1}^{n}\frac{1}{\sqrt{t}})$$
$$\leq C\sqrt{k\log(n)n}. \quad (31)$$

Corollary 3:
From Theorem 1 and [Auer et al., 2002], we have that for $QuBan$ with ϵ-greedy, there is a constant C such that

(a) Setup 1 (larger Δ_i values).

(b) Setup 2 (smaller Δ_i values).
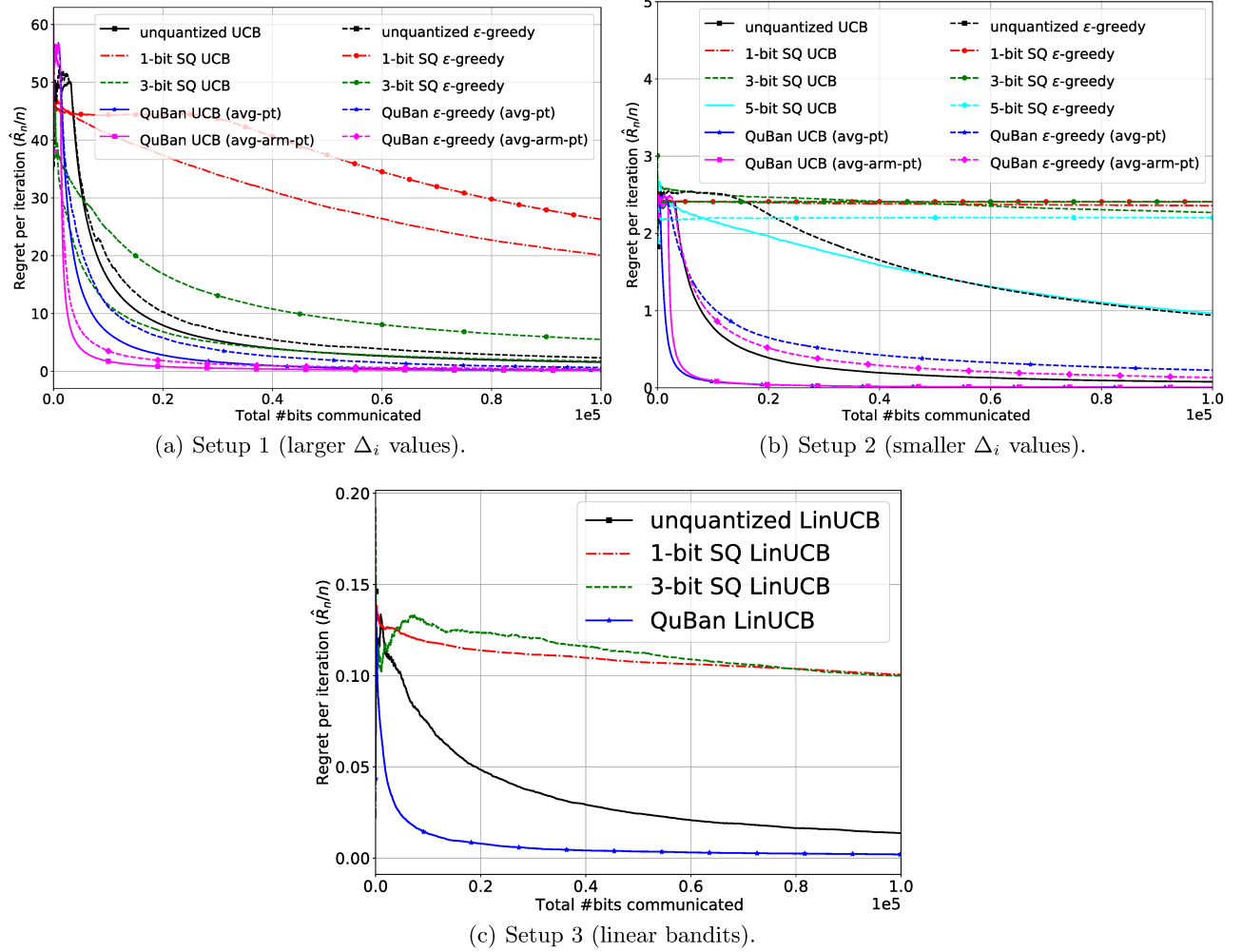
(c) Setup 3 (linear bandits).

Figure 5: Total number of bits versus regret per iteration.

$R_n \leq C\sigma k \log(1 + \frac{n}{k})$. Then,

$$
\begin{aligned}
\frac{1}{n} \sum_{t=1}^{n-1} \frac{R_t}{\sigma t} &\leq \frac{Ck}{n} \sum_{t=1}^{n-1} \frac{\log(1+t)}{t} \\
&\leq \frac{Ck \log(1+n)}{n} \sum_{t=1}^{n-1} \frac{1}{t} \\
&\leq \frac{Ck \log(1+n)}{n} \left(1 + \int_1^{n-1} \frac{1}{t}\right) \\
&\leq \frac{Ck (\log(1+n))^2}{n}.
\end{aligned}
\tag{32}
$$

G   Proof of Corollary 4

We observe that the LinUCB parameters, β_t, can be chosen such that $\max_{t \in \{1,\ldots,n\}} \sup_{a \in \mathcal{A}_t} \langle \theta_t - \theta_*, a \rangle \leq \sqrt{\beta_n}$. Hence, by Cauchy–Schwarz we have that
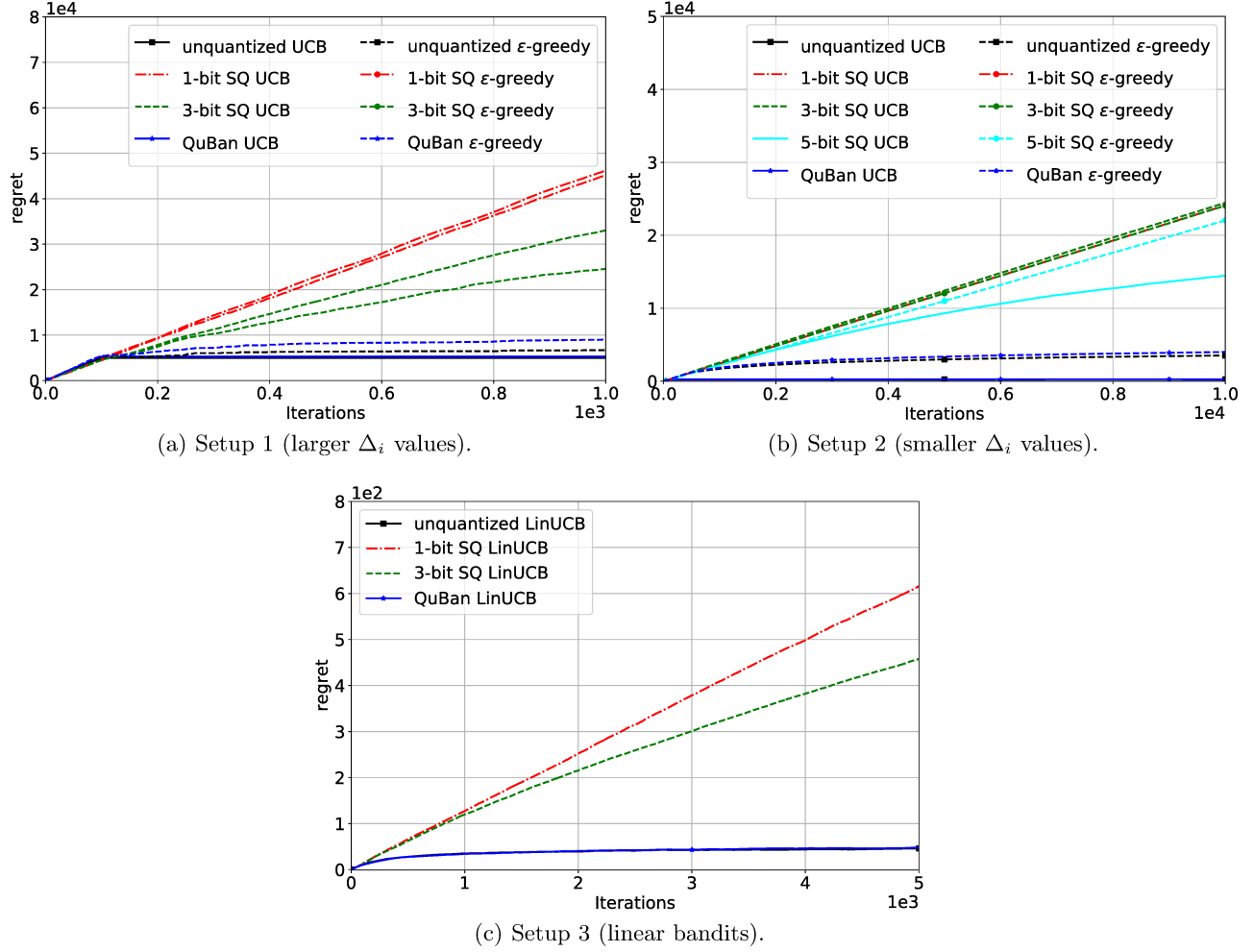
(a) Setup 1 (larger Δ_i values).

(b) Setup 2 (smaller Δ_i values).

(c) Setup 3 (linear bandits).

Figure 6: Regret versus number of iterations.

$$\sum_{t=1}^{n} |\langle \theta_t - \theta_*, A_t \rangle| \leq \sqrt{n \sum_{t=1}^{n} |\langle \theta_t - \theta_*, A_t \rangle|^2}$$

$$\leq \sqrt{n \sum_{t=1}^{n} \min\{\beta_n, \langle \theta_t - \theta_*, A_t \rangle^2\}}. \tag{33}$$

The proof of the expected regret and average number of bits bounds then follows as in [Lattimore and Szepesvári, 2020, Theorem 19.2] using Theorem 1.

H Numerical Results

In this appendix we provide additional numerical results.

We start by describing the 3 setups that we considered in Section 6.
• **Setup 1: (Figs $3 - 6$(a)).** We use $k = 100, \lambda = 100, C = 10$, the arms' means are picked from a Gaussian distribution with mean 0 and standard deviation 10 and the reward distributions are conditionally Gaussian given the actions A_t with variance 0.1. The parameter σ_q is set to be 0.1 for $QuBan$ and $200/2^r - 1$ for the r-bit SQ.
• **Setup 2: (Figs $3 - 6$(b))** This differs from the previous only in that the means are picked from a Gaussian
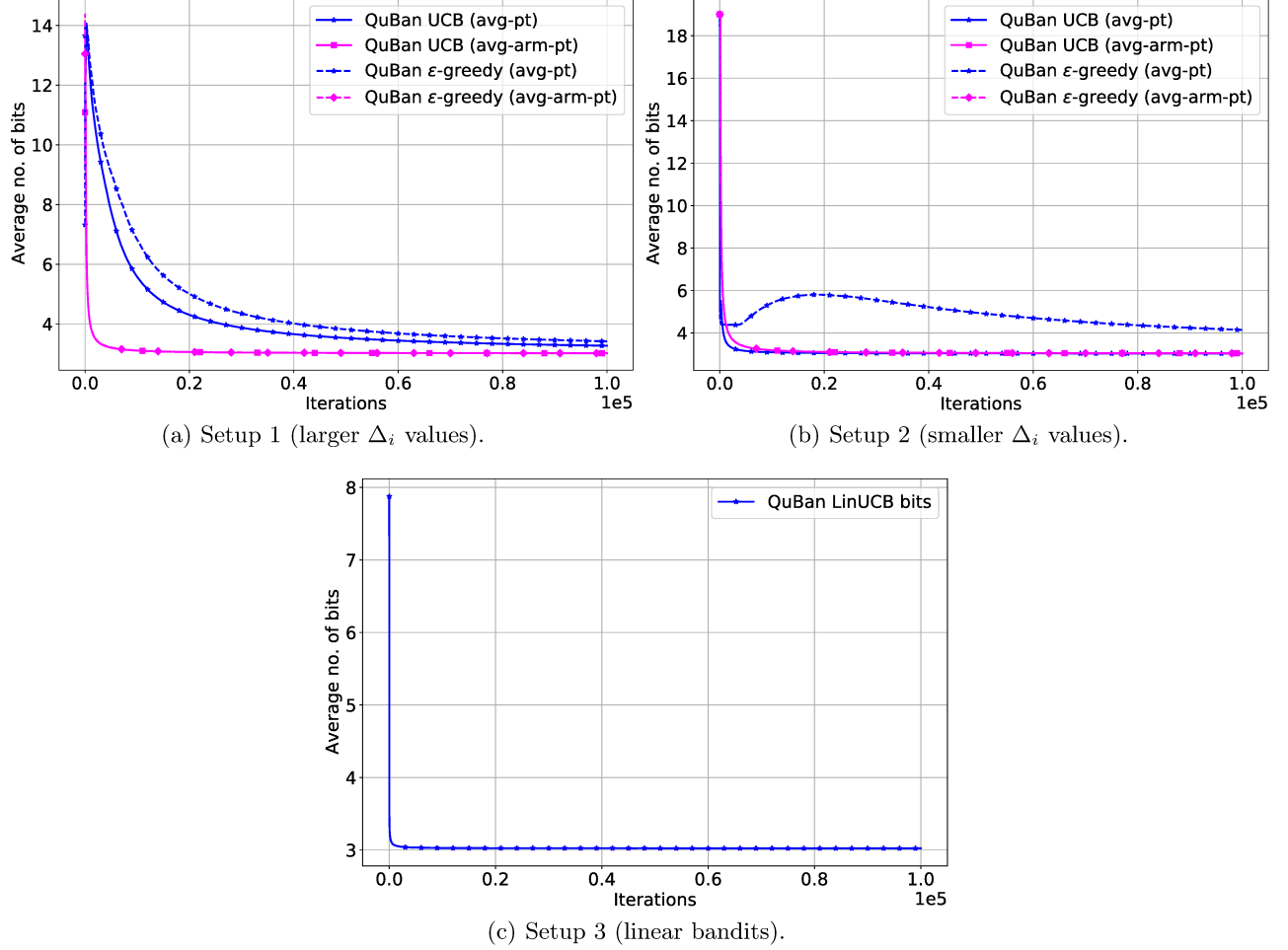
(a) Setup 1 (larger Δ_i values).

(b) Setup 2 (smaller Δ_i values).

(c) Setup 3 (linear bandits).

Figure 7: Average number of bits versus iterations.

distribution with mean 95 and standard deviation 1 (leading to smaller Δ_i).

• **Setup 3: (Figs** $3-6$**(c)).** This is our contextual bandit setup. We use $d = 20$ dimensions, θ_* picked uniformly at random from the surface of a radius 1 ball centered at the origin, and the noise η_t is picked from a Gaussian distribution with zero mean and 0.1 variance. At each time t we construct the actions set $\mathcal{A}_t$ by sampling 5 actions uniformly at random from the surface of a radius 0.5 ball centered at the origin independently of the previously sampled actions. We evaluate the regret and the average number of bits used by $QuBan$ as well as the 3 and 1 bit stochastic quantizers in the interval $[-10, 10]$ (the interval in which we observe the majority of rewards). These quantization schemes are used on top of the LinUCB algorithm. The LinUCB exploration constant is chosen to be σ_q, where σ_q is set to be 0.1 for $QuBan$ and $\frac{20}{2^r-1}$ for the r-bit SQ.

In Fig. 5 we plot $\frac{\hat{R}_n}{n}$, the regret per iteration, vs. the total number of bits communicated. We observe that $QuBan$ allows for more than 10x saving in the number of bits over the unquantized case to achieve the same regret. We also note that avg-arm-pt uses a smaller number of bits when the means of the arms tend to be well separated (Fig. 5(a)) while avg-pt uses a smaller number of bits when they tend to be closer together (Fig. 5(b)). In Fig. 6 we plot the regret versus number of iterations. We observe that $QuBan$ regret becomes sub-linear (which indicates identifying the best arm) much faster than baseline schemes. In Fig. 7 we plot the average number of bits versus iterations. We observe that the avg-pt tends to perform better for a well-behaved bandit scheme and small gaps between arms means, while the avg-arm-pt performs better when the means gap is large or the algorithm picks sub-optimal arms for many iterations (e.g., ϵ-greedy in Fig. 7(b)). We also observe convergence for the average number of bits to ~ 3 bits.

(a) $\lambda = 1$.
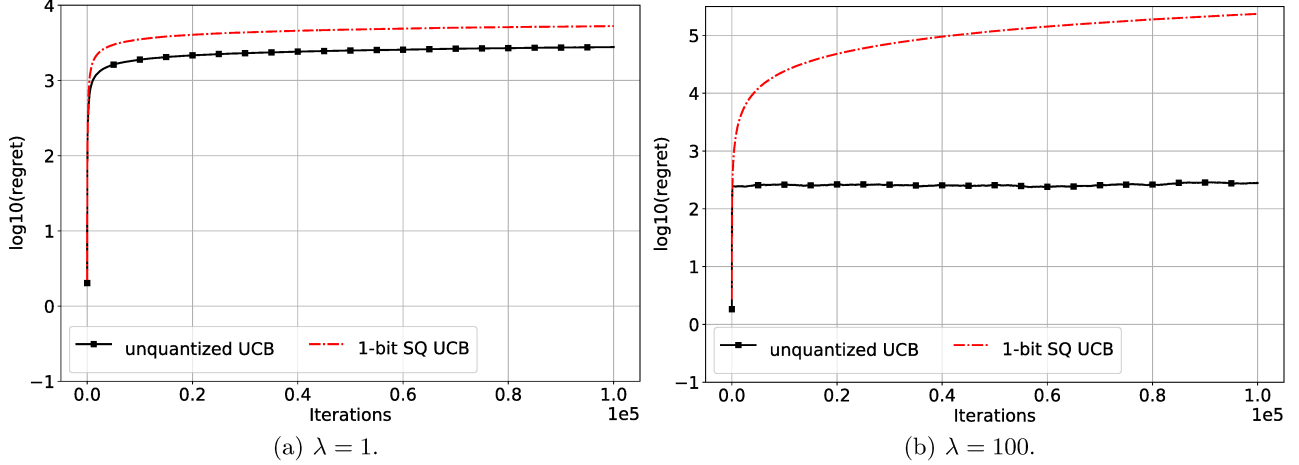
(b) $\lambda = 100$.

Figure 8: Regret versus number of iterations.

I   A Case Where 1 Bit is Sufficient

In this appendix, we provide simulation results for the motivating example in Section 3. We consider the setup in Section 6 with 100 arms where the arms' means are picked from a Gaussian distribution with mean 0 and standard deviation 1 and the reward distributions are conditionally Gaussian given the actions A_t with variance 0.1. The reward distributions are clipped to have support only on an interval $[-\lambda, \lambda]$. The parameter σ_q is set to be 2 when $\lambda = 1$, and 0.1 when $\lambda = 100$ for the unquantized case. For 1-bit SQ, σ_q is set to be 2λ. Fig. 8 shows the regret of unquantized and 1-bit SQ with the UCB algorithm for $\lambda = 1, 100$. As discussed in Section 3, we observe a regret penalty when $\lambda \gg \sigma$.

UCB regret bounds:
The UCB regrets bounds that we use in Section 3 follow directly from the case of reward distributions that are supported on $[0, 1]$ in [Auer et al., 2002]. To see the regret bound for rewards supported on $[-\lambda, \lambda]$, we observe that the expected regret can be written as $R_n = \sum_{t=1}^{n} \mathbb{E}(\mu_t^* - r_t) = \sum_{t=1}^{n} \mathbb{E}(\mu_t^* - \hat{r}_t) = 2\lambda \sum_{t=1}^{n} \mathbb{E}(\frac{(\mu_t^* + \lambda) - (\hat{r}_t + \lambda)}{2\lambda})$, which transforms the problem to one with reward distributions supported on $[0, 1]$.