Pure Exploration Over Bit-Constrained Channels

Fathima Zarin Faizal
EE Department
IIT Bombay, India
fathima@ee.iitb.ac.in

Adway Girish
IC School
EPFL, Switzerland
adway.girish@epfl.ch

Manjesh Kumar Hanawal
IEOR Department
IIT Bombay, India
mhanawal@iitb.ac.in

Nikhil Karamchandani
EE Department
IIT Bombay, India
nikhilk@ee.iitb.ac.in

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

The *multi-armed bandit* problem is a sequential decision-making game between a learner and an environment. At each round, the learner chooses from K actions or arms and receives a reward drawn from the distribution of that particular arm, which is governed by the environment. The objective is to find the "best" arm, which we take to be the one with the highest average reward. We consider the *pure exploration* setting [1], where the algorithm outputs a recommended arm after an initial exploration phase, and the algorithm is evaluated only based on its final recommendation. Thus a "good" algorithm is one that recommends the best arm with high probability (w.h.p.) while reducing the number of samples required from the reward distributions. Best arm identification is a problem that has been well-studied for a while now, and the readers are referred to [2]–[4] for a detailed review.

We consider a variant where the above interaction between the learner and the environment is not possible directly, and instead takes place through an intermediate agent that acts as an interface for a specific arm. Unlike the traditional online learning setup, the learner no longer has access to the rewards with full precision, i.e., the agents observe the rewards obtained and communicate the rewards to the learner over noiseless, bit-constrained channels. A key point is that each agent "represents" a single arm, i.e., it pulls and observes rewards from only one fixed arm.

Motivation: Such a separation between the learner and the environment naturally arises in real-world systems for a wide variety of reasons. For example, when learning from privacy-sensitive data [5], there may be a need to hide the exact rewards obtained. The agent can then aggregate the rewards in such a way that the specific contents remain unintelligible to the learner, but there is still enough information to carry out the task of identifying the best arm. The separation could also be a consequence of the system architecture employed, where it may be infeasible or simply inefficient for the learner to constantly collect and process the data that it receives from a large number of arms. It may then be easier to have the agents perform some of the computations locally, before passing on information in a more condensed format that makes it easier for the learner to come to a decision. Additionally, the channel constraint models applications involving remote, low-complexity agents, where the communication cost of sending the rewards could become a performance bottleneck and reducing the number of bits transmitted would result in lower power consumption and wireless interference. This is particularly significant in IoT networks where devices are typically resource-constrained and battery-powered [6]–[9].

Contributions: We develop a quantization scheme based on the popular Successive Elimination framework [1] that requires only 3 bits for each round of communication. We also show that this algorithm has order-optimal performance with the quantization scheme only contributing a constant factor. Through simulations, we show that this scheme performs better than other quantization schemes in the multi-armed bandit literature.

II. Related work

The pure exploration setting was introduced by [1] and one can consult [10] for a simple presentation of the setting. The book [10] also contains a review of most conventional multi-armed bandit algorithms, all of which assume access to full-precision rewards in every round. Pure exploration has been studied in great detail and several classic algorithms exist, such as Successive Elimination [1], LUCB [11], lil'UCB [12], and Track-and-Stop [13].

Our problem statement is mainly inspired by the works [14] and [15]. In [14], the authors propose a quantization scheme that can be used over any multi-armed bandit algorithm in the regret minimization setting that achieves order-optimal regret and also provide lower bounds that nearly match it. In [15], the authors propose an adaptive quantization scheme and a decision-making policy for the Linear Stochastic Bandit setting, and show that $B = \mathcal{O}(d)$ bits guarantees order-optimal regret, where d is the dimension of the arm set. A fundamental difference between these works and ours is in the scope and capabilities of the agents. In [14], each agent has access to the entire arm set, but they cannot store past rewards — they must encode and send the rewards to the learner as soon as they are observed. This allows for flexible systems where the agents may leave and join at any time, and hence, cannot rely on the agents knowing past rewards. In [15], there is only one

agent, which has access to all arms but can communicate only a bit-constrained version of the received rewards to the learner, which makes the decisions.

The main difference between the settings in these two papers and our problem statement is that their analysis focuses on the regret minimization setting, whereas we focus on the pure exploration setting which is more appropriate for learning channel characteristics using pilot signals as there is a clear exploration phase. The pure exploration setting is a fundamentally different problem that requires strategies that are very different from the optimal strategies in the regret minimization setting even when the samples are not quantized.

For the pure exploration setting, the paper [16] considers a scenario where a learner and multiple clients try to find the best arm together, but each client is allotted a disjoint subset of the arms. Like us, they also propose a quantization scheme for communicating rewards between the clients and the learner. The main difference in the performance of their quantization scheme and the one that we propose is that it is hard to control the number of bits used by their algorithm. See Section V for a more detailed comparison. Another federated setup with a slightly different objective is proposed in [17] but their setting does not involve a bit-constrained channel; they instead add a communication cost per usage of the channel from the agent to the server to penalize communication. They do not require a quantization scheme, as the channel can transmit messages with arbitrary precision. Each agent also has access to the entire arm set.

Our goal is to develop efficient quantization schemes that ensure order-optimal performance for the pure exploration multi-armed bandit problem where the learner is separated from the arms through interfacing agents. We compare our quantization scheme with existing schemes from the literature mentioned above and show .....

III. Problem setup

In this section, we define notation that will be used throughout the paper and formalize our system model. Broadly, we study a fixed-confidence distributed multi-armed bandit (MAB) problem where the decision-making and observing entities are separated. They must communicate their "results" (decisions or observations) to each other over a noiseless channel of finite capacity with the overall goal of performing a learning task.

The distributed MAB. There is a central learner and K distributed agents separated by a rate-constrained channel working together to solve a fixed-confidence stochastic MAB problem consisting of K arms. Each agent $i \in [K]$[1] has access to arm i of the MAB and samples from the reward distribution ν_i associated with arm i if the learner instructs it to. We assume that the distributions $\{\nu_i\}_{i=1}^K$ are σ^2-subgaussian[2] and bounded on $[a, b]$. For $i \in [K]$, let μ_i denote the mean of the distribution ν_i and $r_{i,t}$ denote the t^{th} reward sample drawn from ν_i. We also assume that the means are ordered monotonically, i.e., $\mu_1 \geq \cdots \geq \mu_K$. This assumption is made only for notational convenience; the learner is not aware of such an ordering among the arms. Also, define $\Delta_i = \mu_1 - \mu_i$ for $i \in [K]$.

The communication model. We assume that the agents are only capable of collecting samples from their associated distributions, aggregating the information from the samples it has seen so far and transmitting it to the central server. The agents cannot share information between each other and can communicate only with the central server. This is commonly the case with low-complexity devices such as drones and sensors. Moreover, they are only allowed to use a finite number of bits for each transmission to the learner. We assume that information sent by the agents to the learner that uses a finite number of bits are delivered without any errors or erasures. Note that we do not assume that communication from the learner to the agents is bit-constrained as we assume that the learner has enough resources to communicate. A similar assumption was made in [14]. The broad objective for the learner is to identify the arm with the highest mean reward by sequentially selecting arms and sampling from their associated reward distributions in a causal manner. In other words, time is slotted and based on the information the learner has seen till time $t-1$, the learner can choose to sample from arm A_t in round t.

A learning algorithm for this distributed MAB setup has the following components: (1) a sampling rule for the learner that prescribes what arm is to be pulled in each time slot, (2) a communication rule at the agent that prescribes the time slots at which the agent will communicate with the server, (3) a stopping rule for the learner that ensures that the learner has seen enough samples to find the best arm, and (4) a recommendation rule for the learner that

The learning process is sequential and proceeds in a synchronized, causal manner. At the beginning of *communication round* i, based on the information the learner has seen till $i-1$, it broadcasts an action c_i to all the K agents. After receiving appropriate responses from all or a subset of the agents, the learner updates its estimate of the best arm based on the information it has seen so far till communication round i. This constitutes one communication round. Each agent communicates at most once in each communication round.The overall setup has been illustrated in Figure 1.

Performance metric. We consider the pure exploration problem in the multi-armed bandit literature where there is an initial exploration phase. During this exploration phase, the learner is allowed to sample from the distributions associated with the arms as it pleases without incurring any penalty in a sequential and causal manner. After the exploration phase, the learner is expected to recommend an arm that it thinks is the optimal arm. The learner is evaluated only based on its recommendation. Thus a learner would need a good sampling strategy that provides enough information about the optimality of each arm and a good recommendation strategy that combines the information from the exploration phase to

[1] $[K] := \{1, \ldots, K\}$.

[2] A random variable X is said to be σ^2-subgaussian if for any $t > 0$, $\mathbb{P}\left(|X - E[X]| > t\right) \leq 2\exp\left(-2t^2/\sigma^2\right)$.

find the best arm. Let J_τ denote the arm recommended at the end of τ such rounds.

The pure exploration problem has two variants in the sequential learning literature: the fixed budget setting and the fixed confidence setting, where the latter variant is the focus of our work. In the *fixed confidence setting*, we fix the confidence level δ and work with 'sound' strategies that find the optimal arm in finite time with the given confidence level, i.e., if the strategy stops after τ_δ samples and outputs arm J_{τ_δ}, it satisfies $\mathbb{P}(\tau_\delta < \infty, J_{\tau_\delta} \neq 1) \leq \delta$. The goal is to find sound strategies that minimize τ_δ (which is also called the *sample complexity*), i.e., the number of samples needed to estimate the best arm at the confidence level δ.

Objective. Conventional algorithms for the fixed confidence setting such as Successive Elimination and LUCB assume access to full-precision rewards. We propose to develop learning algorithms for the fixed confidence setting under the communication model outlined above that reduces the cumulative number of bits used to achieve order-optimal performance. The main technical difficulty is the accumulated quantization errors that could potentially lead to sub-optimal performance. On the other hand, using a higher number of bits would result in higher power consumption for the agents.

As MAB algorithms have been implemented for various real-world tasks, the aim is to develop a quantization scheme that works for a large class of conventional algorithms for the fixed-confidence setting, drawing inspiration from [14] where a quantization scheme that ensures order-optimal performance for regret-minimization algorithms was proposed.

In a nutshell, the overall goal is to develop a good quantization scheme that reduces the number of bits used by the agent that will work on top of a large class of conventional fixed-confidence algorithms for the learner and ensures order-optimal sample complexity in the fixed-confidence setting (with respect to the unquantized setting). We would also like to characterize a trade-off between the number of bits used and the sample complexity to understand how each quantity affects the other.

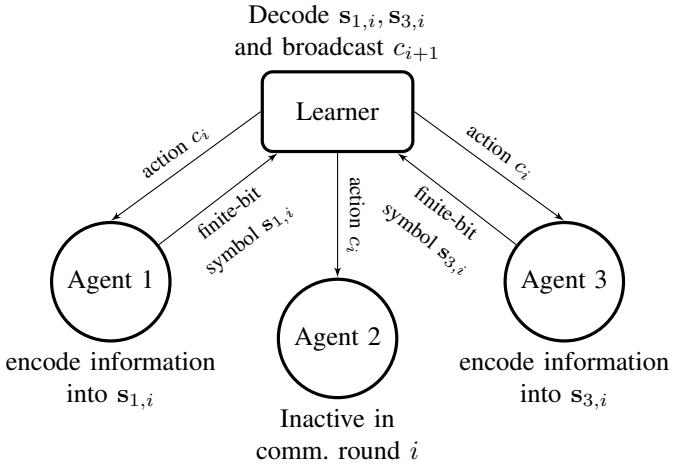

Fig. 1: Block diagram illustrating the overall setup, shown here for the case with 3 agents, i.e., $K = 3$

IV. Proposed algorithm

The algorithm Q-SE is formally described below: Algorithm 1 describes the actions to be taken by the learner while Algorithm 2 describes the agent operation. The main features of this algorithm are the following:

- It uses a constant number of bits B in each communication round. Moreover, any $B \geq 1$ works.
- Communication between the agents and the learner happens exponentially sparsely, i.e., agents communicate with the learner only at time slots that are of the form $\alpha^i, i \in \mathbb{N}$ for some $\alpha > 0$.
- Theorem 5 shows that the sample complexity of this algorithm is higher only by a constant factor of the sample complexity achieved by Successive Elimination without any quantization.
- While the quantization scheme proposed here is used on top of the Successive Elimination framework, this quantization scheme can in fact be used on top of any algorithm for the fixed-confidence setting that uses confidence bounds.

Thus, by using just a finite number of bits for each round of communication, the algorithm achieves order-optimal sample-complexity. The rest of this section will be devoted to providing an informal intuition for this algorithm.

High-level description: The learner constantly keeps track of a set of active arms, i.e., the set of arms still in contention to be the recommended arm. The set S is initialized to be $[K]$. As we would like to reduce the number of bits used, it is inefficient for the agent to communicate each sample that it sees. We thus consider a batched approach that ensures that communication happens in a sparse manner. The learner pulls arms (through the agents) in batches, where b_i is the batch size of communication round i. We also define t_i to be the cumulative sum of arm pulls for each arm till round i, i.e., $t_i = \sum_{j=1}^{i} b_j$.

At the beginning of round i, the learner instructs each active agent in S to sample from their associated distributions b_i times. Each active agent sends to the learner a quantized estimate of the empirical mean of the rewards obtained after a total of t_i cumulative arm pulls. The learner first decodes the quantized estimate, then decides which arms will remain in the active set using the Successive Elimination framework. This marks the end of a communication round. The algorithm terminates when there is only one arm left in the active set, which is the recommended arm.

Before describing the working of the algorithm in more detail, it is instructive to look at the quantization part separately.

Quantization scheme: The agent calculates the estimated mean from the rewards. which must first be quantized and encoded into a bit string to be sent over the bit-constrained channel. Similarly, at the server, we must be able to obtain a decoded estimate of the empirical mean from the encoded bit string. This is achieved as follows. We first fix an interval that we "expect" the estimated mean to belong to with high probability (this will become clear later), divide it into 2^B equal bins, then transmit a bit string that will be decoded at the server as the midpoint of the bin. We formalize this below.

Let $[a, b]$ be the "expected" real interval as described above. First, divide $[a, b]$ into 2^B equal bins $\{[\ell_{i-1}, \ell_i]\}_{i=1}^{2^B}$, where $\ell_i \triangleq a + \frac{i}{2^B}(b-a)$. Further, if $x \in [a, b]$, define $i(x) = \max\{j : \ell_j < x\}$, which takes values in $\{0, 1, \ldots, 2^B - 1\}$. Let $\text{bin} : \mathbb{Z}_+ \to \{0,1\}^B$ be the function that maps nonegative integers into their length B binary representations, and let $\text{num} : \{0,1\}^B \to \mathbb{Z}_+$ be its inverse, mapping binary strings into the nonnegative integers that they represent.

Finally, we define the encoder $\mathsf{enc}(\cdot, B, [a, b]) : \mathbb{R} \to \{0,1\}^B$ by

$$\mathsf{enc}(x, B, [a, b]) = \begin{cases} \text{bin}(0) & x < a, \\ \text{bin}(i(x)) & a \le x \le b, \\ \text{bin}(2^B - 1) & b < x, \end{cases}$$

and the decoder $\mathsf{dec}(\cdot, B, [a, b]) : \{0,1\}^B \to \mathbb{R}$ by

$$\mathsf{dec}(\mathbf{s}, B, [a, b]) = \frac{\ell_{\text{num}(\mathbf{s})} + \ell_{\text{num}(\mathbf{s})+1}}{2}.$$

We may simply refer to them as $\mathsf{enc}(x)$ and $\mathsf{dec}(\mathbf{s})$ when B and the interval are clear from context. Thus, $\mathsf{enc}(x, B, [a, b])$ divides the interval $[a, b]$ into 2^B bins of equal length and identifies the bin into which x falls. If x does not lie in $[a, b]$, then x is mapped to its nearest bin. At the decoder, each bin is mapped to its midpoint. A key observation is that if $x \in [a, b]$, the quantization error between x and the decoded quantized value $\mathsf{dec}(\mathsf{enc}(x))$ is at most $\frac{b-a}{2 \cdot 2^B}$, i.e., if x is known to lie within the interval, then the quantization error is at most a factor of $\frac{1}{2^{B+1}}$ times the width of the interval. The quantization scheme is summarized in Figure 2.

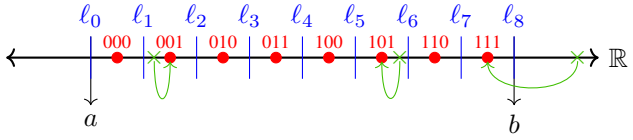

Fig. 2: Illustration of the quantization scheme when $B = 3$ — the blue lines given by ℓ_i mark the 2^B equal bins, the red points denote the midpoints of these bins, and the green "$\times$"s (the values to be quantized) get mapped to their nearest midpoints.

Low-level description: At round i, consider the agent $j \in [K]$ making its k^{th} cumulative arm pull, where $1 \le k \le t_i$. It observes a reward $r_{j,k}$. At the end of this round, it calculates the empirical mean of the rewards from arm j observed over all rounds upto and including round i, $\hat{\mu}_{j,i} = \frac{1}{t_i}\sum_{k=1}^{t_i} r_{j,k}$. Since the communication channel is bit-constrained, the agents cannot simply transmit the infinite precision real number $\hat{\mu}_{j,i}$ as is — we transmit a quantized version of $\hat{\mu}_{j,i}$ as described above. Let $\tilde{\mu}_{j,i}$ be the decoded estimate of the mean of arm j that the server recovers during round i.

The broad idea behind the Successive Elimination framework is to characterize high-confidence bounds for the means of the distributions of each arm. By defining the confidence width

$$U'(i, \delta) = \sigma\sqrt{\frac{2\log(4Kt_i^2/\delta)}{t_i}}, \tag{1}$$

for round i and arbitrary $\delta > 0$, it follows from the subgaussian concentration inequality [10] that

$$\mathbb{P}\left(|\hat{\mu}_{j,i} - \mu_j| > U'(i, \delta)\right) \le \frac{\delta}{2Kt_i^2}. \tag{2}$$

Thus, the actual mean of arm j lies in the interval $[\hat{\mu}_{j,i} - U'(i,\delta), \hat{\mu}_{j,i} + U'(i,\delta)]$ at any round i w.h.p. The upper limit is called the Upper Confidence Bound (UCB) and the lower limit is called the Lower Confidence Bound (LCB). At the end of a communication round, if the UCB of any arm k lies below the LCB of any other arm j, then arm k is removed from the active set. Thus, in the high probability event that these confidence bounds contain the actual mean, removing arms whose UCBs lie below the LCB of some other arm would guarantee that the algorithm is removing only suboptimal arms. The algorithm will make a mistake only when these high probability events do not happen. In particular, we will show that for any fixed $\delta > 0$, these events fail to happen with a probability of at most δ, and hence, the algorithm will recommend the best arm with probability at least $1 - \delta$.

Had there been no separation between the learner and the agents (as in a standard MAB setting), we would be done. However, to account for the potential increase in error due to the quantization necessitated by the bit-constrained channel, we introduce a slack in the confidence interval through a different confidence width $U(i, \delta)$. The goal is to obtain a concentration bound for $\tilde{\mu}_{j,i} - \mu_j$ in terms of $U(i, \delta)$, in a form similar to (1). We now provide an intuitive explanation

to motivate an expression of $U(i,\delta)$ that achieves exactly this. (Note that this is not meant to be a proof; that this does indeed work will be shown in Section V). First note that a straightforward application of the triangle inequality gives

$$|\tilde{\mu}_{j,i} - \mu_j| \leq |\tilde{\mu}_{j,i} - \hat{\mu}_{j,i}| + |\hat{\mu}_{j,i} - \mu_j|. \tag{3}$$

The first term corresponds to the quantization error and the second term corresponds to the error in the empirical mean itself. The latter is taken care of by the bound (1), so it is enough to bound the quantization error. Recall from the discussion following the definition of enc and dec, that the quantization error is at most $\frac{1}{2^{B+1}}$ times the width of the interval if the estimated mean is known to originally lie in the interval. Thus, our task is to find an appropriate interval to perform the quantization.

As the only information that the learner has access to in communication round i is the quantized estimate of round $i-1$, i.e., $\tilde{\mu}_{j,i-1}$, the interval constructed must be centered around $\tilde{\mu}_{j,i-1}$. We also want that this interval contains the new empirical mean at round i, i.e., $\hat{\mu}_{j,i}$ w.h.p. To obtain a recursive expression for $U(i,\delta)$, we make the inductive assumption that $\tilde{\mu}_{j,i-1}$ lies in $[\mu_j - U(i-1,\delta), \mu_j + U(i-1,\delta)]$ w.h.p. Combined with the knowledge that $\hat{\mu}_{j,i}$ lies in $[\mu_j - U'(i-1,\delta), \mu_j + U'(i-1,\delta)]$ w.h.p. from (1), we have that the interval $[\tilde{\mu}_{j,i-1} - U'(i,\delta) - U(i-1,\delta), \tilde{\mu}_{j,i-1} + U'(i,\delta) + U(i-1,\delta)]$ contains $\hat{\mu}_{j,i}$ w.h.p. This is illustrated in Figure 3. Note that we have found an interval that we "expect" the estimated mean to belong to, as promised when defining the quantization scheme relative to an interval. Thus, we perform the quantization over an interval of width $2[U'(i,\delta) + U(i-1,\delta)]$, and hence, the quantization error is at most $\frac{1}{2^B}[U'(i,\delta) + U(i-1,\delta)]$.

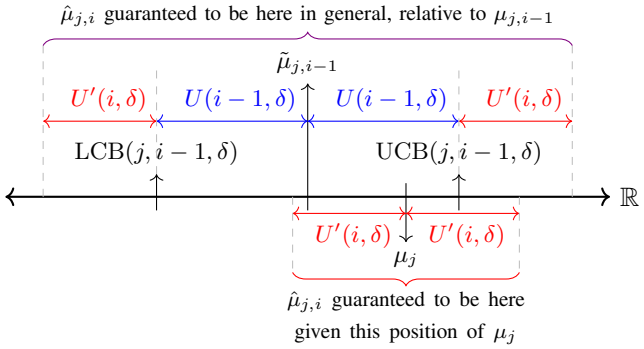

Fig. 3: Motivation for the recursive definition of $U(i,\delta)$ in terms of $U(i-1,\delta)$ — Start with $\tilde{\mu}_{j,i-1}$, then by the inductive hypothesis, μ_j should lie within $U(i-1,\delta)$ of $\tilde{\mu}_{j,i-1}$ (blue interval). From (1), $\hat{\mu}_{j,i}$ lies within $U'(i,\delta)$ of μ_j (red interval). Putting them together, $\hat{\mu}_{j,i}$ lies within $U(i-1,\delta) + U'(i,\delta)$ of $\tilde{\mu}_{j,i-1}$.

Motivated by the above, define, for $i \geq 1, j \in [K]$,

$$U(i,\delta) = U'(i,\delta) + \frac{1}{2^B}\left[U'(i,\delta) + U(i-1,\delta)\right], \tag{4}$$

where we set $U(0,\delta) = b - a$ to ensure that the induction hypothesis (namely, that $\tilde{\mu}_{j,i-1}$ satisfies its confidence bounds) holds for the index $i-1=0$, and the second equality follows by converting the equation into a nonrecursive form. Now that we have an appropriate confidence width $U(i,\delta)$ that we expect (heuristically, from the above discussion; this will be proved in Section V) to provide a similar concentration inequality for $\tilde{\mu}_{j,i}$ as (1) does for $\hat{\mu}_{j,i}$, we define the appropriate LCB and UCB for our algorithm, as

$$\mathrm{LCB}(j,i,\delta) = \tilde{\mu}_{j,i} - U(i,\delta),\ \mathrm{UCB}(j,i,\delta) = \tilde{\mu}_{j,i} + U(i,\delta).$$

The results proved in Section V are for the case where $t_i = \alpha^i$, i.e., communication happens at exponentially sparse time slots. The sample complexity remains order-optimal primarily due to Lemma 4, i.e., $U(i,\delta)$ can be shown to be a constant multiple of $U'(i,\delta)$. Thus the performance of this algorithm changes only by a constant factor.

Algorithm 1 Q-SE algorithm (learner-side)

procedure Q-SE-LEARNER($K, \delta, B, \{b_i\}$)
  Let $S \leftarrow \{1, \ldots, K\}$
  For $1 \leq j \leq K$, let $\tilde{\mu}_{j,0}$ be sampled uniformly from $[a,b]$
  for $1 \leq i < \infty$ **do**
    for $j \in S$ **do**
      Instruct agent j to sample b_i times
      Receive quantized value $\mathbf{s}_{j,i}$ from agent j
      $L_{i,j} \leftarrow [\mathrm{LCB}(j,i-1,\delta) - U'(i,\delta),\ \mathrm{UCB}(j,i-1,\delta) + U'(i,\delta)]$
      Decode $\tilde{\mu}_{j,i} = \mathsf{dec}(\mathbf{s}_{j,i}, B, L_{i,j})$
    end for
    $S \leftarrow S \setminus \{m \in S : \max_{j \in [K]} \mathrm{LCB}(j,i,\delta) \geq \mathrm{UCB}(m,i,\delta)\}$
    STOP if $|S| = 1$
  end for
  return only element in S
end procedure

Algorithm 2 Q-SE algorithm (agent-side)

procedure Q-SE-AGENT$_j$($\delta, B, i, \tilde{\mu}_{j,i-1}$)
  Pull arm j b_i times
  $L_{i,j} \leftarrow [\mathrm{LCB}(j,i-1,\delta) - U'(i,\delta),\ \mathrm{UCB}(j,i-1,\delta) + U'(i,\delta)]$
  Send $\mathbf{s}_{j,i} = \mathsf{enc}(\hat{\mu}_{j,i}, B, L_{i,j})$
  return quantized value $\mathbf{s}_{j,i}$
end procedure

V. Analysis of the Q-SE algorithm

We now show that the algorithm works with high probability and also calculate the sample complexity of the algorithm. We do so by a sequence of lemmas and theorems, whose proofs can be found in the longer version at ..... The steps

are similar to a standard analysis of Successive Elimination-type algorithms, such as in [1] and [17]. The novelty in our work is the use of Lemma 1 to relate the quantization and subgaussian confidence intervals, so we provide a proof here.

Recall that μ_j is the mean of arm $j \in [K]$, $\hat{\mu}_{j,i}$ is the estimated mean of arm j in round i at the agent (which will then be encoded and sent to the learner), and $\tilde{\mu}_{j,i}$ is the decoded estimate of the mean of arm j in round i at the learner. Note that the only concentration bound we know about these quantities *a prori* is (2), which relates $\hat{\mu}_{j,i}$ and $U'(i,\delta)$. The server, however, observes $\tilde{\mu}_{j,i}$ and constructs confidence intervals of width $U(i,\delta)$. Our goal is for the server to be able to identify w.h.p. the best arm in $[K]$. To this end, it is desirable to have a concentration bound on the quantities at the server, namely, $\tilde{\mu}_{j,i}$ and $U(i,\delta)$.

Lemma 1 below relates the following — (1) the event that the estimated mean of arm j, $\hat{\mu}_{j,i}$, eventually (hence the union over rounds i) falls outside the confidence interval of width $U'(i,\delta)$ centered about the true mean μ_j, and (2) the identical event at the server, except with the decoded mean $\tilde{\mu}_{j,i}$ and confidence width $U(i,\delta)$.

Lemma 1. *For all arms $1 \le j \le K$, and any $\delta > 0$,*

$$\bigcup_{i=1}^{\infty} \{|\tilde{\mu}_{j,i} - \mu_j| > U(i,\delta)\} \subseteq \bigcup_{i=1}^{\infty} \{|\hat{\mu}_{j,i} - \mu_j| > U'(i,\delta)\}.$$

Proof of Lemma 1. First observe that the sets on the left-hand side are empty when $i = 0$, so it is sufficient to show that

$$\bigcup_{i=0}^{\infty} \{|\tilde{\mu}_{j,i} - \mu_j| > U(i,\delta)\} \subseteq \bigcup_{i=1}^{\infty} \{|\hat{\mu}_{j,i} - \mu_j| > U'(i,\delta)\}.$$

We do so by instead proving by induction the following equivalent statement — for each $0 \le i < \infty$,

$$\{|\tilde{\mu}_{j,i} - \mu_j| > U(i,\delta)\} \subseteq \bigcup_{k=1}^{\infty} \{|\hat{\mu}_{j,k} - \mu_j| > U'(k,\delta)\}.$$

The base case, $i = 0$, holds trivially as the left-hand side is $\emptyset$. Assume now that this property holds for index $i-1 \ge 0$, i.e.,

$$\{|\tilde{\mu}_{j,i-1} - \mu_j| > U(i-1,\delta)\} \subseteq \bigcup_{k=1}^{\infty} \{|\hat{\mu}_{j,k} - \mu_j| > U'(k,\delta)\}.$$

Using (3) and (4) respectively, we have

$$\begin{aligned}
&\{|\tilde{\mu}_{j,i} - \mu_j| > U(i,\delta)\} \\
&\subseteq \{|\tilde{\mu}_{j,i} - \hat{\mu}_{j,i}| + |\hat{\mu}_{j,i} - \mu_j| > U(i,\delta)\}, \\
&\subseteq \{|\hat{\mu}_{j,i} - \mu_j| > U'(i,\delta)\} \cup \\
&\quad \left\{|\tilde{\mu}_{j,i} - \hat{\mu}_{j,i}| > \frac{1}{2^B} U'(i,\delta) + \frac{1}{2^B} U(i-1,\delta)\right\},
\end{aligned}$$

where the second step follows as $U(i,\delta) = U'(i,\delta) + \frac{1}{2^B}U'(i,\delta) + \frac{1}{2^B}U(i-1,\delta)$ and $a + b > x + y$ requires at least one of $a > x$ or $b > y$.

We now bound the second term in the union on the right-hand side. The estimated mean is encoded using enc on the interval $[\mathrm{LCB}(j,i-1,\delta) - U'(i,\delta), \mathrm{UCB}(j,i-1,\delta) + U'(i,\delta)]$. As long as $\hat{\mu}_{j,i}$ lies in this interval, the learner can decode $\tilde{\mu}_{j,i}$ to an error within $\frac{1}{2} \cdot \frac{1}{2^B}$ times the width of this interval, given by $2\left(U'(i,\delta) + U(i-1,\delta)\right)$. Hence we have

$$\begin{aligned}
&\left\{|\tilde{\mu}_{j,i} - \hat{\mu}_{j,i}| > \frac{1}{2^B} U'(i,\delta) + \frac{1}{2^B} U(i-1,\delta)\right\} \\
&\subseteq \{|\tilde{\mu}_{j,i-1} - \hat{\mu}_{j,i}| > U'(i,\delta) + U(i-1,\delta)\} \\
&\subseteq \{|\tilde{\mu}_{j,i-1} - \mu_j| > U(i-1,\delta)\} \cup \{|\hat{\mu}_{j,i} - \mu_j| > U'(i,\delta)\} \\
&\subseteq \bigcup_{k=1}^{\infty} \{|\hat{\mu}_{j,k} - \mu_j| > U'(k,\delta)\}, \qquad (5)
\end{aligned}$$

where the last step follows from the induction hypothesis. □

Using Lemma 1, we obtain the desired concentration bound on the decoded means at the server, stated formally in Lemma 2. This allows us to also prove Theorem 3, which guarantees w.h.p. that the best arm will not be mistaken for a suboptimal arm and eliminated from our candidate set.

Lemma 2. *For any $\delta > 0$, define the event*

$$\mathcal{E} := \bigcup_{j=1}^{K} \bigcup_{i=1}^{\infty} \{|\tilde{\mu}_{j,i} - \mu_j| > U(i,\delta)\},$$

then $\mathbb{P}(\mathcal{E}) \le \delta$.

Theorem 3. *With probability $\ge 1 - \delta$, the best arm remains in the active set S until termination.*

Note that Theorem 3 partially proves that the algorithm works — any arm that is removed is suboptimal w.h.p. All that is left is to show is that all suboptimal arms are removed at some point before termination, which together with the previous result, guarantees that the best arm is identified w. h. pw.h.p. We do so in the exponentially sparse communication regime, i.e., with $t_i = \alpha^i$. We also require a technical result relating the two confidence widths $U(i,\delta)$ and $U'(i,\delta)$ under this exponentially sparse regime, given below in Lemma 4. In particular, we show that they are within a constant factor of each other. This allows us to calculate the sample complexity of the algorithm, given in Theorem 5.

Lemma 4. *If $t_i = \alpha^i$ where $\alpha \in \mathbb{N}$ such that $\alpha < 2^{2B}$ and $B \ge 1$, then for $i \ge 1$,*

$$U(i,\delta) \le 2c\,U'(i,\delta),$$

where

$$c = \left(1 + \frac{2}{2^B}\right) \frac{2^B}{2^B - \sqrt{\alpha}}.$$

Theorem 5. *Let $t_i = \alpha^i$ such that $\alpha < 2^{2B}$ and $B \ge 1$. If the algorithm successfully identifies the best arm, it will terminate after*

$$\mathcal{O}\left(\sum_{j \ne 1} \frac{410\alpha c^2\sigma^2}{\Delta_j^2} \ln\left(\frac{256c^2\sigma^2\sqrt{4K\delta}}{\Delta_j^2}\right) + 1\right).$$

samples, where c is as defined in Lemma 4, and we assume w.l.o.g. that the optimal arm is $j = 1$.

We thus have that with probability $1-\delta$, the Q-SE algorithm identifies the best arm in $\mathcal{O}\left(\sum_{j\neq 1}\frac{410\alpha c^2\sigma^2}{\Delta_j^2}\ln\left(\frac{256c^2\sigma^2\sqrt{4K\delta}}{\Delta_j^2}\right)+1\right)$ communication rounds. The upper bound is worse by a factor of $4c^2$ compared to the unquantized Successive Elimination scheme communicating exponentially sparsely and shows that this algorithm has order-optimal performance.

Note that c depends on the choice of B, and in fact decreases as B increases. We thus see a trade-off between the performance of the algorithm and the number of bits that it is allowed to use. In the next section, we will see that $B=2$ is enough to obtain a sample complexity that is comparable to other quantization schemes proposed in the bandit literature such as QUBAN and the quantization scheme proposed in [15] while improving upon average the number of bits used by the algorithm to converge.

Remark 1. *The algorithm QuBan proposed in [14] for the regret minimization setting too has a parameter ϵ that provides a trade-off between the number of bits used and the performance of the algorithm. It is not hard to adapt* QUBAN *to the fixed confidence setting and show that it has order-optimal performance. The downside to using this algorithm is that it is not clear how to tune ϵ to ensure that the algorithm satisfies a power constraint. The quantization scheme proposed in [16] does not provide a trade-off between the number of bits used by the algorithm and its performance.*

Remark 2. *The quantization scheme proposed here can in fact be used for any algorithm that uses confidence bounds, such as LUCB and not just for Successive Elimination, to obtain order-optimal performance using a finite number of bits. The crux of this scheme is the recursive definition for $U(\cdot,\cdot)$ that we obtain by separating the quantization error and the error in the empirical mean itself via* (3)*. A similar order-optimality analysis can be carried out for LUCB too that we omit in our work as our focus is on the quantization scheme.*

VI. EXTENDING Q-SE TO UNBOUNDED REWARDS

So far, the performance of Q-SE was analyzed for multi-armed bandits whose associated distributions have a bounded support $[a,b]$. In this section, we will discuss methods to extend the same results to the case with unbounded rewards.

The only place where we use the boundedness of the rewards is in the definition of $U(0,\delta)$. Recall that the algorithm proceeds with an initial guess for the mean of each arm $\{\tilde{\mu}_{j,0}: 1\leq j\leq K\}$. As $U(i,\delta)$ is defined in a recursive fashion, for the induction in the proof of Lemma 1 to hold, $U(0,\delta)$ needs to be such that $\{\tilde{\mu}_{j,0}: 1\leq j\leq K\}$ are good guesses for the actual mean with high probability satisfying Lemma 1. For bounded rewards, (**??**) ensures that this holds. Note that even if it is known that the means of the arms lie in a compact set, then the same analysis holds.

The main difference when we transition to unbounded rewards is that we can no longer use a constant number of bits in each round. Specifically, in the first round, it is not possible to obtain a high-probability guess for the mean by using a bounded number of bits. One way to rectify this is to use another quantization scheme only in the first round to quantize the empirical mean that ensures that the quantization error is bounded. Then by defining $U(1,\delta)=U'(1,\delta)+M$, where M is the bound on the quantization error, using (3), we have that $U(1,\delta)$ is a valid high-confidence bound. Normal Q-SE can be run starting from round 2. In Section VII, we demonstrate this on Gaussian rewards by running QuBan in the first round.

VII. NUMERICAL EXPERIMENTS

In this section, we present results of numerical experiments comparing the performance of Q-SE with other quantization algorithms proposed for multi-armed bandits in related literature. The main quantization schemes that we compare against are QuBan proposed in [14] and the scheme proposed in [15], which we call Conf-Quant throughout the rest of this work for notational simplicity. For a fair comparison, each of these schemes were implemented on top of the same batched Successive Elimination algorithm that Q-SE uses to highlight the difference between the quantization schemes.

Conf-Quant is the closest quantization scheme to Q-SE: it also uses the idea of confidence bounds to identify where the mean would lie with high probability. In each round i, the entire $[a,b]$ interval is divided into intervals of range $U'(i,\delta)$ and the empirical mean is quantized to the midpoint of one of these intervals. Intuitively, this works, because as noted in Section IV, the algorithm is willing to tolerate quantization errors within a constant factor of the empirical mean's confidence bound. The main drawback of this approach is that the number of bits used in each round is inversely proportional to the confidence bound in each round, i.e., the number of bits used per round grows with the number of rounds, making it hard to control the cumulative number of bits that the algorithm uses.

QuBan was initially proposed for the regret minimization setting where each sample that the agent observes is quantized. To use QuBan in a batched setup such as this, the natural extension is to scale the parameter M_t (see [14] for more details) by $1/\sqrt{t}$. The broad idea behind QuBan is to assign longer codes to quantize samples farther away from the current estimate of the mean at the server and shorter codes to quantize samples close to the current estimate to minimize the expected number of bits used in each round. While this is a sound approach, this might result in a higher number of bits being used unnecessarily. This is because, as noted earlier in Section IV, the SE algorithm can tolerate errors within a constant factor of the empirical mean's confidence bound and quantization schemes that aim for smaller errors would be inefficient. This also highlights the fact that algorithms proposed for the regret minimization setting need not work for the fixed-confidence setting.

The first set of experiments in Figures 4(a), 4(b), 4(c), 4(d) and 4(e) analyze the dependence of the performance of Q-SE

on the parameters α and B. We consider a five-armed multi-armed bandit instance where each arm is associated with a beta distribution whose means are generated uniformly at random from the interval $[0, 1]$. The performance of each algorithm was averaged over 4000 iterations. In Figures 4(a) and 4(b), we observe that the number of communication rounds used by Q-SE to converge decreases with α while the number of samples used increases with α. This is expected because increasing α results in sparser communication between the agents and the server reducing the round complexity while the number of samples used increases. Figure 4(c) shows the dependence of the performance of Q-SE on the cumulative number of bits used by the algorithm to converge (which we call *communication complexity*). The decrease in the communication complexity with α is a natural consequence of the decrease in the communication round complexity. Even though the case with $B = 1$ works theoretically, the number

The second set of experiments in Figure 4(f), 4(g) and 4(h) compare the performance of Q-SE, QuBan and Conf-Quant when the distributions associated with the arms are bounded. We consider a five-armed multi-armed bandit instance where each arm is associated with a beta distribution whose means are generated uniformly at random from the interval $[0, 1]$. The performance of each algorithm was averaged over 4000 iterations. We observe that Q-SE with $B = 2$ performs comparably with QuBan and Conf-Quant while using a much lesser number of bits to converge.

The third set of experiments in Figure 4(i), 4(j) and 4(k) compare the performance of Q-SE and QuBan when the distributions associated with the arms are Gaussian. We consider a five-armed multi-armed bandit instance where each arm is associated with a Gaussian distribution whose means are generated uniformly at random from the interval $[0, 8]$. See Section VI for extending Q-SE to unbounded rewards. The performance of each algorithm was averaged over 4000 iterations. We again observe that Q-SE with $B = 2$ performs comparably with QuBan and Conf-Quant while using a much lesser number of bits to converge.

The fourth set of experiments in Figure 4(l), 4(m) and 4(n) compare the dependence of the performance of Q-SE and QuBan on the hardness of the underlying bandit instance when the distributions associated with the arms are Gaussian. We consider a five-armed multi-armed bandit instance where each arm is associated with a Gaussian distribution. Four of the arms have mean 0 and the remaining arm has mean $\Delta \in [0, 1]$. A lower Δ implies that the mean of the optimal arm being closer to that of the non-optimal arms, resulting in a harder instance. The performance of each algorithm was averaged over 4000 iterations. As expected, the performance improves as the hardness of the instance decreases. We observe that Q-SE with $B = 2$ performs better than QuBan on harder instances.

VIII. Conclusion

References

[1] E. Even-Dar, S. Mannor, and Y. Mansour, "PAC bounds for multi-armed bandit and markov decision processes," in *Computational Learning Theory*. Springer, 2002.

[2] J. Audibert, S. Bubeck, and R. Munos, "Best arm identification in multi-armed bandits," in *COLT*, 2010.

[3] K. Jamieson and R. Nowak, "Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting," in *CISS*, 2014.

[4] S. Bubeck, R. Munos, and G. Stoltz, "Pure exploration in finitely-armed and continuous-armed bandits," *Theoretical Computer Science*, 2011.

[5] B. Liu, M. Ding, S. Shaham, W. Rahayu, F. Farokhi, and Z. Lin, "When machine learning meets privacy: A survey and outlook," *ACM Computing Surveys (CSUR)*, 2021.

[6] P. Dames, P. Tokekar, and V. Kumar, "Detecting, localizing, and tracking an unknown number of moving targets using a team of mobile robots," *The International Journal of Robotics Research*, 2017.

[7] V. Savic, H. Wymeersch, and S. Zazo, "Belief consensus algorithms for fast distributed target tracking in wireless sensor networks," *Signal Processing*, 2014.

[8] L. Song, C. Fragouli, and D. Shah, "Recommender systems over wireless: Challenges and opportunities," in *IEEE Information Theory Workshop (ITW)*, 2018.

[9] K. Ding, J. Li, and H. Liu, "Interactive anomaly detection on attributed networks," in *Proceedings of the 12th ACM international conference on web search and data mining (WSDM)*, 2019.

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

[11] S. Kalyanakrishnan, A. Tewari, P. Auer, and P. Stone, "PAC subset selection in stochastic multi-armed bandits." in *ICML*, 2012.

[12] K. Jamieson, M. Malloy, R. Nowak, and S. Bubeck, "lil'ucb: An optimal exploration algorithm for multi-armed bandits," in *COLT*. PMLR, 2014.

[13] A. Garivier and E. Kaufmann, "Optimal best arm identification with fixed confidence," in *COLT*. PMLR, 2016.

[14] O. A. Hanna, L. Yang, and C. Fragouli, "Solving multi-arm bandit using a few bits of communication," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2022.

[15] A. Mitra, H. Hassani, and G. J. Pappas, "Linear stochastic bandits over a bit-constrained channel," 2022. [Online]. Available: https://arxiv.org/abs/2203.01198

[16] A. Mitra, H. Hassani, and G. Pappas, "Exploiting heterogeneity in robust federated best-arm identification," 2021. [Online]. Available: https://arxiv.org/abs/2109.05700

[17] K. S. Reddy, P. N. Karthik, and V. Y. F. Tan, "Almost cost-free communication in federated best arm identification," *AAAI (to appear)*, 2023. [Online]. Available: https://arxiv.org/abs/2208.09215

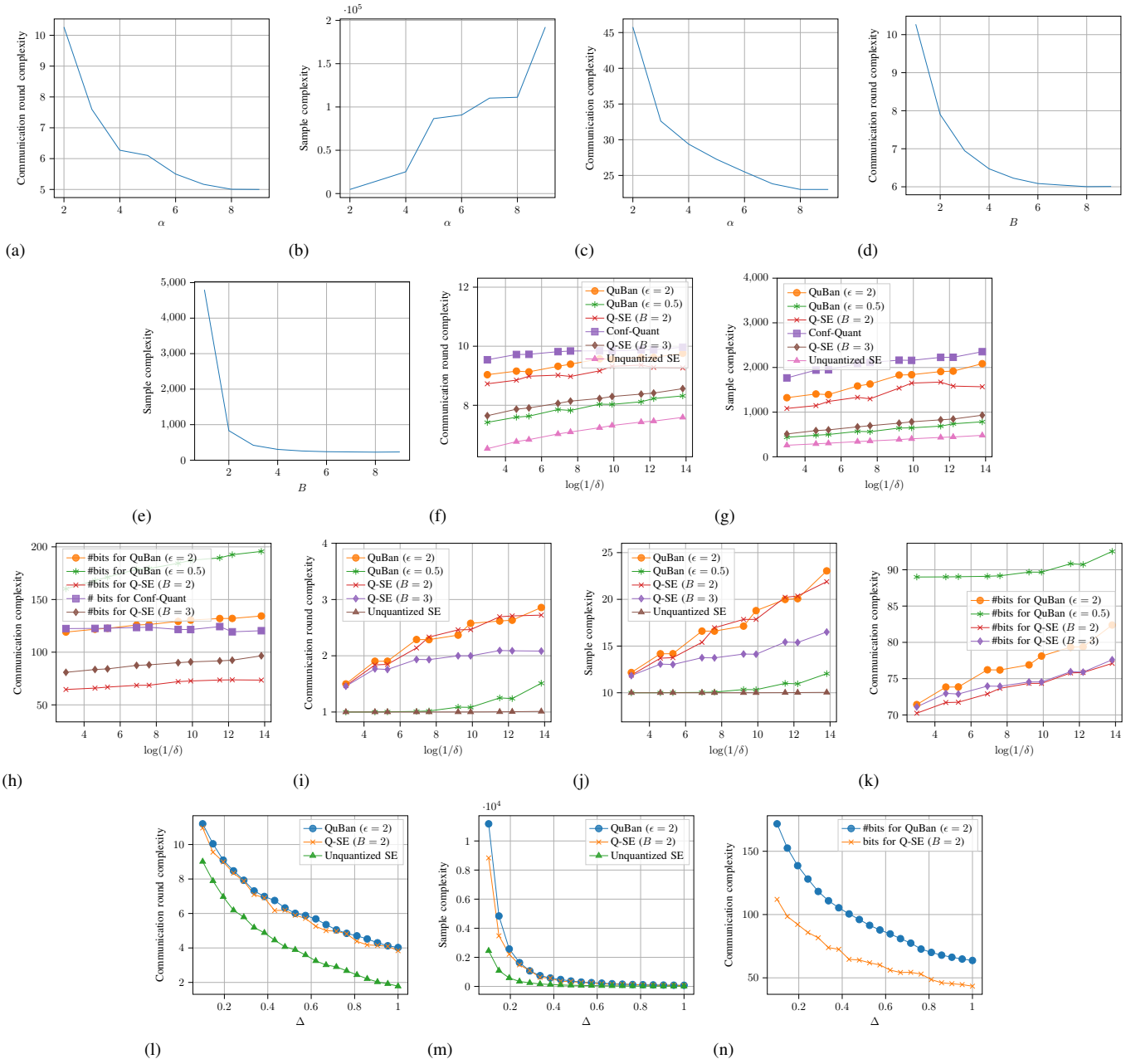

Fig. 4: Figures (a), (b) and (c) demonstrate dependence of the performance of Q-SE on α. Figures (d) and (e) demonstrate the dependence on β. Figures (f), (g) and (h) compare Q-SE with QuBan and Conf-Quant for bounded rewards while Figures (i), (j) and (k) compare Q-SE with QuBan for unbounded rewards. Finally, Figures (l), (m) and (n) compare the dependence of Q-SE and QuBan on the hardness of the underlying instance.