Tracking an Auto-Regressive Process with Limited Communication

Rooji Jinan  Parimal Parag  Himanshu Tyagi

Abstract—Samples from a high-dimensional AR[1] process are quantized and sent over a time-slotted communication channel of finite capacity. The receiver seeks to form an estimate of the process in real-time. We consider the slow-sampling regime where multiple communication slots occur between two sampling instants. We propose a successive update scheme which uses communication between sampling instants to update the estimates of the latest sample. We show that there exist quantizers that render the fast but loose version of this scheme, which updates estimates in every slot, universally optimal asymptotically. However, we provide evidence that most practical quantizers will require a judiciously chosen update frequency.

I. INTRODUCTION

We consider the setting of real-time decision systems based on remotely sensed observations. In this setting, the decision maker needs to track the remote observations with high precision and in a timely manner. These are competing requirements, since high precision tracking will require larger number of bits to be communicated, resulting in larger transmission delay and increased staleness of information. Towards this larger goal, we study the following problem.

Consider a discrete time first-order auto-regressive (AR[1]) process $X_t \in \mathbb{R}^n$, $t \geq 0$. A sensor draws a sample from this process, periodically once every $s$ time-slots. In each of these time-slots, the sensor can send $nR$ bits to a center. The center seeks to form an estimate $\hat{X}_t$ of $X_t$ at time $t$, with small mean square error (MSE). Specifically, we are interested in minimizing the time-averaged error $\sum_{t=1}^T E \|X_t - \hat{X}_t\|^2 / T$ to enable timely and accurate tracking of $X_t$.

We propose and study a successive update scheme where the encoder computes the error in the estimate of the latest sample at the decoder and sends its quantized value to the decoder. The decoder adds this value to its previous estimate to update the estimate of the latest sample, and uses it to estimate the current value using a linear predictor. We instantiate this scheme with a general gain-shape quantizer for error-quantization.

Note that we can send this update several times between two sampling instances. In particular, our interest lies in comparing a fast but loose scheme where an update is sent every slot to a slower update every $p$ communication slots. The latter allows the encoder to use more bits for the update, but the decoder will need to wait longer. We analyze this scheme for a universal setting and show that the fast but loose successive update scheme, used with an appropriately selected quantizer, is optimal asymptotically in the dimension.

To show this optimality, we use a random construction for the quantizer, based on the spherical code given in [8], [28]. Roughly speaking, this ideal quantizer $Q$ yields $E \|y - Q(y)\|^2 \leq \|y\|^2 2^{-2R}$ for bounded $y$. However, in practice, at finite $n$, such quantizers need not exist. Most practical vector quantizers have an extra additive error, i.e., the error bound takes the form $E \|y - Q(y)\|^2 \leq \|y\|^2 \theta + n \varepsilon^2$. We present our analysis for such general quantizers. Interestingly, for such a quantizer (which is all we have at a finite $n$), the optimal choice of $p$ can differ from 1. Thus, we present a theoretically sound guideline for choosing the frequency of updates $1/p$ for practical quantizers.

There is a large body of prior work on related problems. The structure of real-time encoders for source coding have been studied in [9], [11], [23]–[25], [27], [29]. Closer to our work, remote estimation problem under communication constraints of various kind have been studied in [1], [10], [16]–[18], [22], [26]. Recursive state estimation algorithms under communication constraints have been considered in [2], [14], [20], [21]. In [6], [12], [13], [30], authors discuss encoding and reconstruction of a stationary source using noisy, rate-limited samples. More recently, [5] considers the problem of quantizing a Gauss-Markov process.

To the best of our knowledge, none of these prior works consider communication delays or study the tradeoff between accuracy and delay that our analysis brings out. The closest related work to our setting is [5], where the authors use a Gaussian codebook for quantization. Note that after the first round of quantization, the error vector need not be Gaussian, and the analysis in [5] can only be applied after showing a closeness of the error vector distribution to Gaussian in the Wasserstein distance of order 2. While the original proof [5] overlooks this technical point, this gap can be filled using a recent result from [7] if spherical codes are used. However, we follow an alternative approach and show a direct analysis using vector quantizers. In fact, our analysis is valid for any AR[1] process with bounded fourth moment increments. Also, unlike [5], we account for transmission delays and consider fixed-length coding schemes.

II. PROBLEM FORMULATION

We begin by providing a formal description of our problem. For $\alpha \in (0, 1)$, we consider a discrete time auto-regressive
process of order $1$ (AR[1] process) in $\mathbb{R}^n$,
\begin{equation}
X_t = \alpha X_{t-1} + \xi_t, \quad t \geq 0,
\end{equation}
where $(\xi_t \in \mathbb{R}^n, t \geq 1)$ is an independent and identically distributed (i.i.d.) random sequence with zero mean and covariance matrix $\sigma^2(1-\alpha^2)I_n$. For simplicity, we assume that $X_0 \in \mathbb{R}^n$ is a zero mean random variable with covariance matrix $\sigma^2I_n$. This implies that the variance of $X_t \in \mathbb{R}^n$ is $\sigma^2t^2$ for all $t \geq 0$. In addition, we assume that there exists $\kappa > 0$ such that $\sup_{x \in \mathbb{R}^n} \frac{1}{\pi} \sqrt{E\|X_t\|^2} \leq \kappa$. We denote the set of processes $X$ satisfying the assumptions above by $\mathcal{X}_n$ and the class of all such processes for different choices of dimension $n$ as $\mathcal{X}$.

This discrete time process is sub-sampled periodically at sampling frequency $1/s$, for some $s \in \mathbb{N}$, to obtain samples $(X_{ks} \in \mathbb{R}^n, k \geq 0)$. The sampled process $(X_{ks}, k \geq 0)$ is passed to an encoder which converts it to a bit stream. The encoder operates in real-time and sends $nR$s bits between any two sampling instants. Specifically, the encoder is given by a sequence of mappings $(\phi_t)_{t \geq 0}$, where the mapping at any discrete time $t = ks$ is denoted by $\phi_t : \mathbb{R}^{n(k+1)} \rightarrow \{0,1\}^{nR}$. The encoder output at time $t = ks$ is denoted by the codeword $C_t \triangleq \phi_t(X_0, X_s, \ldots, X_{ks})$. We represent this codeword by an $s$-length sequence of binary strings $C_t = (C_{t,0}, \ldots, C_{t,s-1})$, where each term $C_{t,i}$ takes values in $\{0,1\}^{nR}$. For $t = ks$ and $0 \leq i \leq s - 1$, we can view the binary string $C_{t,i}$ as the communication sent at time $t + i$.

The output bit-stream of the encoder is sent to the receiver via an error-free communication channel. We assume slotted transmission with synchronization where in each slot the transmitter sends $nR$ bits of communication error-free. Note that there is a delay of a time-unit (corresponding to one slot) in transmission of each $nR$ bits. Therefore, the vector $C_{ks,i}$ of $nR$ bits transmitted at time $ks + i$ is received at time instant $ks + i + 1$ for $0 \leq i \leq s - 1$. We use the notation $I_k \triangleq \{ks, \ldots, (k+1)s-1\}$ and $I_k = I_k + 1 = \{ks+1, \ldots, (k+1)s\}$, respectively, for the set of transmit and receive times for the strings $C_{ks,i}$, $0 \leq i \leq s - 1$.

We describe the operation of the receiver at time $t \in I_k$, for some $k \in \mathbb{N}$, such that $i = t - ks \in \{0, \ldots, s - 1\}$. Upon receiving the codewords $C_s, C_{2s}, \ldots, C_{(k-1)s}$ and the partial codeword $(C_{ks,0}, \ldots, C_{ks,i-1})$ at time $t = ks + i$, the decoder estimates the current-state $X_t$ of the process using the estimator mapping $\psi_t : \{0,1\}^{nRt} \rightarrow \mathbb{R}^n$. We denote the overall communication received by the decoder until time instant $t$ by $C^{t-1}$. Further, we denote by $\hat{X}_{t|t}$ the real-time causal estimate $\psi_t(C^{t-1})$ of $X_t$ formed at the decoder at time $t$. Thus, the overall real-time causal estimation scheme is described by the mappings $(\phi_t, \psi_t)_{t \geq 0}$. As a convention, we assume that $\hat{X}_{0|0} = 0$.

We call the encoder-decoder mapping sequence $(\phi_t, \psi_t)_{t \geq 0}$ a tracking code of rate $R$ and sampling period $s$. The tracking error of our tracking code at time $t$ for process $X$ is measured by the mean squared error (MSE) per dimension given by $D_t(\phi, \psi, X) \triangleq \frac{1}{n}E\|X_t - \hat{X}_{t|t}\|^2$. Our goal is to design $(\phi, \psi)$ with low average tracking error $\overline{D_T}(\phi, \psi, X)$ given by $\overline{D_T}(\phi, \psi, X) \triangleq \frac{1}{T} \sum_{t=0}^{T-1} D_t(\phi, \psi, X)$. We restrict to a finite time horizon setting. Instead of the MSE, a more convenient parameterization for us is that of the accuracy, given by $\delta^T(\phi, \psi, X) = 1 - \frac{\overline{D_T}(\phi, \psi, X)}{\sigma^2}$.

**Definition 1 (Maxmin tracking accuracy).** The worst-case tracking accuracy for $\mathcal{X}_n$ attained by a tracking code $(\phi, \psi)$ is given by $\delta^*(\phi, \psi, X_n) = \inf_{X \in \mathcal{X}_n} \delta^T(\phi, \psi, X)$. The maxmin tracking accuracy for $\mathcal{X}_n$ at rate $R$ and sampling period $s$ is given by $\delta^*_n(R, s, X_n) = \sup_{(\phi, \psi)} \delta^T(\phi, \psi, X_n)$, where the supremum is over all tracking codes $(\phi, \psi)$.

The maxmin tracking accuracy $\delta^*_n(R, s, X_n)$ is the fundamental quantity of interest for us. Recall that $n$ denotes the dimension of the observations in $X_t$ for $X \in \mathcal{X}_n$ and $T$ the time horizon. However, we will only characterize $\delta^*_n(R, s, X_n)$ asymptotically in $n$ and $T$. Specifically, we define the asymptotic maxmin tracking accuracy as $\delta^*(R, s, X) = \lim_{n \rightarrow \infty} \lim_{T \rightarrow \infty} \sup_{(\phi, \psi)} \delta^T(\phi, \psi, X_n)$, where we will provide a characterization of $\delta^*(R, s, X)$ and show the existence of a tracking code that attains it.

### III. The Successive Update Scheme

In this section, we present our main contribution in this paper: namely the **Successive Update** tracking code. Before we describe the scheme completely, we present its different components.

**Decoder structure.** Once the quantized information is sent by the transmitter, at the receiver end, the decoder estimates the state $X_t$ using the codewords received until time $t$. Since we are interested in forming estimates with small MSE, the decoder simply forms the minimum mean square error (MMSE) estimate using all the observations until that point. Specifically, for $t \geq u$, denoting by $\hat{X}_{u|t}$ the MMSE estimate $X_u$ formed by the communication $C^{t-1}$ received until time $t$, given by $(c.f. [19])$, $\hat{X}_{u|t} = E[X_u|C^{t-1}]$. The following result presents a simple structure for $\hat{X}_{u|t}$ for our AR[1] model. The simple proof is given in the extended version.

**Lemma 1 (MMSE Structure).** The MMSE estimates $\hat{X}_{u|t}$ and $\hat{X}_{t-i|t}$, respectively, of samples $X_t$ and $X_{t-i}$ at any time $t \in I_k$ and $i = t - ks$ using communication $C^{t-1}$ are related as $\hat{X}_{t-i|t} = \alpha^i \hat{X}_{t-i|t} = \alpha^i E[X_{ks}|C^{t-1}]$.

**Encoder structure: Refining the error successively.** The structure of the decoder exposed in Lemma 1 gives an important insight for encoder design: The communication sent between two sampling instants is used only to form estimates of the latest sample. This principle can be applied (as a heuristic) for any estimate $\hat{X}_{ks|i}$ for $X_{ks}$ formed at the receiver at time $t$ (which need not be the MMSE estimate $\hat{X}_{ks|i}$). Our encoder computes and quantizes the error in the receiver estimate of the latest process sample at each time instant $t = ks + i$ and sends it as communication $C_{ks,i}$.

Even within this structural simplification, a very interesting question remains. Since the process is sampled only in $s$ time slots, we have, potentially, $nRs$ bits to encode the latest sample. At any time $t \in I_k$, the receiver has access to $(C_0, \ldots, C_{(k-1)s})$ and the partial codewords $(C_{ks,0}, \ldots, C_{ks,+1})$ for $i = t - ks$. A simple approach for the encoder is to use the complete codeword to express the
This approach will result in slow but accurate updates of the sample estimates. An alternative fast but loose approach will send $nR$ quantizer codewords to refine estimates in every communication slot. Should we prefer fast but loose estimates or slow but accurate ones? Our results will shed light on this conundrum.

**The choice of quantizers.** In our description of the encoder structure above, we did not specify a key design choice, namely the choice of the quantizer. We will restrict to using the same quantizer to quantize the error in each round of communication. Roughly speaking, we allow any gain-shape [3] quantizer which separately sends the quantized value of the gain $\|y\|_2$ and the shape $y/\|y\|_2$ for input $y$. Formally, we use the following abstraction.

**Definition 2** ($(\theta, \varepsilon)$-quantizer family). Fix $0 < M < \infty$. For $0 \leq \theta \leq 1$ and $0 \leq \varepsilon$, a quantizer $Q$ with dynamic range $M$ specified by a mapping $Q : \mathbb{R}^n \to \{0, 1\}^{nR}$ constitutes an $nR$ bit $\theta, \varepsilon$-quantizer if for every vector $y \in \mathbb{R}^n$ such that $\|y\|_2 \leq nM^2$, we have $E[|y - Q(y)|^2] \leq \|y\|_2^2 \theta(R) + n^{2\varepsilon}$.

Further, for a mapping $\theta : \mathbb{R}_+ \to [0, 1]$, which is a decreasing function of rate $R$, a family of quantizers $Q = \{Q_R : R > 0\}$ constitutes an $(\theta, \varepsilon)$-quantizer family if for every $R$ the quantizer $Q_R$ constitutes an $nR$ bit $(\theta(R), \varepsilon)$-quantizer.

Here, the expectation is taken with respect to the randomness in the quantizer, which is assumed to be shared between the encoder and the decoder for simplicity. The parameter $M$, termed the dynamic range of the quantizer, specifies the domain of the quantizer. When the input $y$ does not satisfy $\|y\|_2 \leq \sqrt{n}M$, the quantizer simply declares a failure, indicated by a symbol $\perp$.

**Description of the successive update scheme.** All the conceptual components of our scheme are ready. We use the structure of Lemma 1 and focus only on updating the estimates of the latest observed sample $X_{ks}$ at the decoder. Our encoder successively updates the estimate of the latest sample at the decoder by quantizing and sending estimates for errors in the estimate.

To decide on the appropriate frequency of update, we opt for a more general scheme where the $nR$s bits available between two samples are divided into $m = s/p$ sub-fragments of length $nRp$ bits each. We use an $nRp$ bit quantizer to refine error estimates for the latest sample $X_{ks}$ (obtained at time $t = ks$) every $p$ slots, and send the resulting quantizer codewords as partial tracking codewords $(C_{ks,jp}, \ldots, C_{ks,(j+1)p-1})$, $0 \leq j \leq m - 1$. Specifically, the $k$th codeword transmission interval $I_k$ is divided into $m$ sub-fragments $I_{k,j}$, $1 \leq j \leq m$ given by $(C_{ks,jp}, \ldots, C_{ks,(j+1)p-1})$, $0 \leq j \leq m - 1$, and $(C_{ks,jp}, \ldots, C_{ks,(j+1)p-1})$ is transmitted in communication slots in $I_{k,j}$.

At time instant $t = ks + jp + 1$ the decoder receives the $j$th sub-fragment $(C_{ks,t-ks}, t \in I_{k,j})$ of $nRp$ bits, and uses it to refine the estimate of the latest source sample $X_{ks}$. Note that the fast but loose and slow but accurate regimes described above correspond to $p = 1$ and $p = s$, respectively. In the middle of the interval $I_{k,j}$, the decoder ignores the partially received quantization code and retains the estimate $\hat{X}_{ks}$ of $X_{ks}$ formed at time $ks + (j - 1)p + 1$. It forms an estimate of the current state $X_{ks+t}$ by simply scaling $\hat{X}_{ks}$ by a factor of $\alpha^t$, as suggested by Lemma 1.

Finally, we impose one more additional simplification to the decoder structure. Instead of using MMSE estimates for the latest sample, we simply update the estimate by adding to it the quantized value of the error. Thus, the decoder has a simple linear structure.

Recall that we denote the estimate of $X_u$ formed at the decoder at time $t \geq u$ by $\hat{X}_{u|t}$. We start by initializing $\hat{X}_{0|0} = 0$ and then proceed using the encoder and the decoder algorithms outlined above. Note that our quantizer $Q_p$ may declare failure symbol $\perp$, in which case the decoder must still yield a nominal estimate. We will simply declare the estimate as 0 once a failure happens. We give a formal description of our encoder and decoder algorithms below.

**The encoder.**

1. Initialize $k = 0$, $j = 0$, $\hat{X}_{0|0} = 0$.
2. At time $t = ks + jp$, use the decoder algorithm (to be described below) to form the estimate $\hat{X}_{ks|t}$ and compute the error $Y_{k,j} \triangleq \hat{X}_{ks} - \hat{X}_{ks|t}$, where we use the latest sample $X_{ks}$ available at time $t = ks + jp$.
3. Quantize $Y_{k,j}$ to $nR$ bit as $Q_p(Y_{k,j})$.
4. If quantize failure occurs and $Q_p(Y_{k,j}) = \perp$, send $\perp$ to the receiver and terminate the encoder.
5. Else, send a binary representation of $Q_p(Y_{k,j})$ as the communication $(C_{ks,0}, \ldots, C_{ks,p-1})$ to the receiver over the next $p$ communication slots.
6. If $j < m - 1$, increase $j$ by 1; else set $j = 0$ and increase $k$ by 1. Go to Step 2.

**The decoder.**

1. Initialize $k = 0$, $j = 0$, $\hat{X}_{0|0} = 0$.
2. At time $t = ks + jp$, if encoding failure has not occurred until time $t$, compute $X_{ks|kp} = \hat{X}_{ks|kp} + (j - 1)p + 1$, and output $\hat{X}_{t|t} = \alpha^{t-ks} \hat{X}_{t|t-1}$.
3. Else, if encoding failure has occurred and the symbol is received declare $\hat{X}_{u|t} = 0$ for all subsequent time instants $u \geq t$.
4. At time $t = ks + jp + i$, for $i \in [p - 1]$, output $\hat{X}_{t|t} = \alpha^{t-ks} \hat{X}_{t|t-1}$.
5. If $j < m - 1$, increase $j$ by 1; else set $j = 0$ and increase $k$ by 1. Go to Step 2.

**IV. MAIN RESULTS**

**Characterization of the maximum tracking accuracy.** To describe our result, we define functions $\delta_0 : \mathbb{R}_+ \to [0, 1]$ and $g : \mathbb{R}_+ \to [0, 1]$ as

$$\delta_0(R) \triangleq \frac{\alpha^2(1 - 2^{-2R})}{1 - \alpha^2 - 2^{-2R}}, \quad g(s) \triangleq \frac{(1 - \alpha^2 s)}{s(1 - \alpha^2)}.$$

Our main result are as follows.

1. For simplicity, we do not account for the extra message symbol needed for sending $\perp$.
2. We ignore the partial quantizer codewords received as $(C_{ks,jp+1}, C_{ks,jp+2}, \ldots, C_{ks,jp+i-1})$ until time $t$. 
**Theorem 2** (Lower bound for maxmin tracking accuracy: The achievability). For $R > 0$ and $s \in \mathbb{N}$, the asymptotic maxmin tracking accuracy is bounded below as

$$\delta^*(R, s, X) \geq \delta_0(R)g(s).$$

Furthermore, this bound can be obtained by a successive update scheme with $p = 1$ and an appropriately chosen $Q_p$.

**Theorem 3** (Upper bound for maxmin tracking accuracy: The converse). For $R > 0$ and $s \in \mathbb{N}$, the asymptotic maxmin tracking accuracy is bounded above as

$$\delta^*(R, s, X) \leq \delta_0(R)g(s).$$

Furthermore, the upper bound is obtained by considering a Gauss-Markov process.

Thus, $\delta^*(R, s, X) = \delta_0(R)g(s)$ with the fast but loose successive update scheme being universally (asymptotically) optimal and the Gauss-Markov process being the most difficult process to track. Clearly, the best possible choice of sampling period is $s = 1$ and the highest possible accuracy at rate $R$ is $\delta_0(R)$, whereby we cannot hope for an accuracy exceeding $\delta_0(R)$.

**Guidelines for choosing a good $p$.** The proof of Theorem 2 entails the analysis of the successive update scheme for $p = 1$. In fact, we can analyze this scheme for any $p \in \mathbb{N}$ and for any $(\theta, \varepsilon)$-quantizer family; we term this tracking code the $p$-successive update ($p$-SU) scheme. This analysis can provide a simple guideline for the optimal choice of $p$ depending on the performance of the quantizer.

However, there are some technical caveats. The quantizer family will operate only as long as the input $y$ satisfies $\|y\|_2 \leq M$. If a $y$ outside is observed, the quantizer will declare $\perp$ and the tracking code encoder, in turn, will declare a failure. We denote by $\tau$ the stopping time at which encoder failure occurs for the first time, i.e., $\tau \triangleq \min\{ks + jp : Q_p(Y_{k,j}) = \perp, 0 \leq k, 0 \leq j \leq m - 1\}$. Further, denote by $A_t$ the event that failure does not occur until time $t$, i.e., $A_t \triangleq \{\tau > t\}$. We characterize the performance of a $p$-SU in terms of the probability of encoder failure in a finite time horizon $T$.

**Theorem 4** (Performance of $p$-SU). For fixed $\theta, \varepsilon, \beta \in [0, 1]$, consider the $p$-SU scheme with an $nRp$ bit $(\theta, \varepsilon)$-quantizer $Q_p$, and denote the corresponding tracking code by $(\phi_p, \psi_p)$. Suppose that for a time horizon $T \in \mathbb{N}$, the tracking code $(\phi_p, \psi_p)$ satisfies $P(\tau \leq T) \leq \beta^2$. Then, $\sup_{X_0 \in X_n} \sup_{\tau \leq T} I(\phi_p, \psi_p, X) \leq B_T(\theta, \varepsilon, \beta)$, where $B_T(\theta, \varepsilon, \beta)$ satisfies

$$\lim_{T \to \infty} \sup_{\tau \leq T} B_T(\theta, \varepsilon, \beta) \leq \sigma^2\left[1 - \frac{g(s)}{1 - \frac{\alpha^{2p}}{1 - \frac{\alpha^{2p}}{1 - \frac{\theta}{1 - \frac{\alpha^{2p}}{1 - \frac{\theta}{1 - \frac{\theta}{1 - 2\theta}}}}}}\right] + 
\kappa\beta g(s)\left(1 - \frac{\alpha^{2(\sigma^2 + p)}}{1 - \frac{\alpha^{2p}}{1 - \frac{\theta}{1 - \frac{\alpha^{2p}}{1 - \frac{\theta}{1 - 2\theta}}}}\right).$$

We remark that $\beta$ can be made small by choosing $M$ to be large for a quantizer family. Furthermore, the inequality in the upper bound for the MSE in the previous result (barring the dependence on $\beta$) comes from the inequality in the definition of a $(\theta, \varepsilon)$-quantizer, rendering it a good proxy for the performance of the quantizer. The interesting regime is that of very small $\beta$ where the encoder failure doesn’t occur during the time horizon of operation. If we ignore the dependence on $\beta$, the accuracy of the $p$-SU does not depend either on $s$ or on the bound for the fourth moment $\kappa$. Motivated by these insights, we define the accuracy-speed curve of a quantizer family as follows.

**Definition 3** (The accuracy-speed curve). For $\alpha \in [0, 1], \sigma^2$, and $R > 0$, the accuracy-speed curve for a $(\theta, \varepsilon)$-quantizer family $Q$ is given by

$$\Gamma_Q(p) = \frac{\alpha^{2p}}{1 - \alpha^{2p} \theta(Rp)} \left(1 - \frac{\varepsilon^2}{\sigma^2} - \theta(Rp)\right), \quad p > 0.$$
The recipe above can be used to analyze practical quantizers, such as the recently proposed almost optimal quantizer in [15].

V. PROOF SKETCHES

Analysis of the Successive Update Scheme. Since the successive update scheme refines the estimate of $X_{k,s}$ in each interval $I_k$ successively, we can establish the following recursion for its MSE.

**Lemma 5.** For a time instant $t = k + j + i$, $j + 1 \in [m]$, $i \in [p]$ and $k \geq 0$, let $(\phi_p, \psi_p)$ denote the tracking code of a $p$-SU scheme employing an $nR$ bit $(\theta, \epsilon)$-quantizer. Assume that $\mathbb{P}(A^*_k) \leq \beta^2$. Then, we have

$$
D_1(\phi_p, \psi_p, X) \leq \alpha^2 2^{-\epsilon(x_k)} D_{k,s}(\phi_p, \psi_p, X) + \sigma^2 (1 - \alpha^{-2\epsilon(x_k)}) + \alpha^2 2^{-\epsilon(x_k)} \kappa \beta.
$$

This recursion is the main tool in the proof of Theorem 4. **Sketch of proof of Theorem 4:** We can write the average per dimension MSE for the $p$-SU scheme for time-horizon $T = Ks$ as

$$
\mathbb{D}_T(\phi_p, \psi_p, X) = \frac{1}{Ks} \sum_{k=0}^{K-1} \sum_{j=0}^{m-1} \sum_{i=1}^{p} D_{k,s+j+p+1}(\phi_p, \psi_p, X),
$$

whereby an application of Lemma 5 and some further simplification gives

$$
\mathbb{D}_T(\phi_p, \psi_p, X) \leq \sigma^2 + g(s)(\kappa \beta - \sigma^2 + \frac{\epsilon^2}{1 - \theta}) + \frac{(1 - \alpha^{-2\epsilon(x_k)} \sum_{k=0}^{K-1} D_{k,s+j+p+1}(\phi_p, \psi_p, X) - \frac{\epsilon^2}{1 - \theta}).
$$

Next, using Lemma 5 once again, we have

$$
D_{k,s}(\phi_p, \psi_p, X) \leq \alpha^2 2^{-\epsilon(x_k)} D_{k,s-1}(\phi_p, \psi_p, X) + \sigma^2 (1 - \alpha^{-2\epsilon(x_k)}) + \alpha^2 \epsilon^2 (1 - \theta^{-m}) + \alpha^2 \kappa \beta.
$$

Therefore, $\mathbb{D}_T(\phi_p, \psi_p, X)$ can be bounded above further by replacing the term $\frac{1}{K} \sum_{k=0}^{K-1} D_{k,s+j+p+1}(\phi_p, \psi_p, X)$ on the right-side of (2) with the supremum of such an average over all sequences satisfying the previous recursive inequality. We define $B_T(\theta, \epsilon, \beta)$ to be this new bound for $\mathbb{D}_T(\phi_p, \psi_p, X)$, which does not depend on $X$. In the limit as $T$ goes to infinity, we can take the limit of the bound as $m$ and $K$ go to infinity to get the claimed bound. In doing so, we need to take limit for average of a sequence with terms satisfying a recursive inequality. The simple observation below gives the required bound.

**Lemma 6.** For a sequence $(X_k \in \mathbb{R} : k \in \mathbb{Z}_+)$ that satisfies recursive bounds $X_k \leq a X_{k-1} + b$ with constants $a, b \in \mathbb{R}$ such that $b$ is finite and $a \in (-1, 1)$, we have

$$
\lim_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} X_k \leq \frac{b}{1 - a}.
$$

Asymptotic achievability using random quantizer. With Theorem 4 at our disposal, the proof of achievability can be completed by fixing $p = 1$ and showing the existence of appropriate quantizer. But we need to handle the failure event. We do this first. The next result shows that the failure probability depends on the quantizer only through $M$.

**Lemma 7.** For fixed $T$ and $n$, consider the $p$-SU scheme with $p = 1$ and an $nR$ bit $(\theta, \epsilon)$-quantizer $Q$ with dynamic range $M$. Then, for every $\eta > 0$, there exists an $M_0$ independent of $n$ such that for all $M \geq M_0$, we get $\mathbb{P}(A^*_T) \leq \eta$.

The bound above is rather loose, but it suffices for our purpose. In particular, it says that we can choose $M$ sufficiently large to make probability of failure until time $T$ less than any $\beta^2$, whereby Theorem 4 can be applied by designing a quantizer for this $M$. Indeed, we can use the quantizer of unit sphere from [8], [28], along with a uniform quantizer for gain (which lies in $[-M, M]$) to get the following performance.

**Lemma 8.** For every $R, \epsilon, \gamma, M > 0$, there exists an $nR$ bit $(2^{-2(\gamma-\epsilon)}, \epsilon)$-quantizer with dynamic range $M$, for all $n$ sufficiently large.

**Proof of Theorem 2:** For any fixed $\beta$ and $\epsilon$, we can make the probability of failure until time $T$ less than $\beta$ by choosing $M$ sufficiently large. Further, for any fixed $R, \gamma > 0$, by Lemma 8, we can choose $n$ sufficiently large to get an $nR$ bit $(2^{-2(\gamma-\epsilon)}, \epsilon)$-quantizer for vectors $y$ with $||y||^2 \leq nM^2$. Therefore, by Theorem 4 applied for $p = 1$, we get that

$$
\delta^*(R, s, X) \geq \sigma^2 \left[1 - \frac{g(s)}{1 - \alpha^{-2-2(\gamma-\epsilon)}} \left(1 - \frac{\epsilon^2}{\sigma^2} - \frac{\epsilon^2}{2^{-2(\gamma-\epsilon)}} \right) \right] + \frac{\kappa \beta}{(1 - \alpha^{-2})} \left(1 - \frac{\epsilon^2}{s(1 - \alpha^{-2})} \right).$$

The proof is completed upon taking the limits as $\epsilon, \gamma$, and $\beta$ go to 0.

**Converse bound.** The converse proof makes use of the standard properties of entropy power $N(X)$ and is similar to the converse proof in [5]. It can be seen after some basic manipulations that the tracking error for any tracking code $(\phi, \psi)$ satisfies

$$
D_T(\phi, \psi, X) \geq \frac{1}{nKs} \sum_{k=0}^{K-1} \sum_{i=0}^{s-1} \alpha^2 \mathbb{E} \left[ ||X_{k,s-1} - \tilde{X}_{k,s}||^2 \right] + \frac{N(\xi)}{\sigma^2 \alpha^2} \left(1 - \frac{1 - \alpha^{-2s}}{s(1 - \alpha^{-2})} \right).
$$

We relate the MSE terms on the right-side to entropy power and use properties of entropy power to get

$$
D_T(\phi, \psi, X) \geq \frac{\alpha^2 s(1 - \alpha^{-2s} - 2Rs)}{s(1 - \alpha^{-2s} - 2Rs)} + \frac{N(\xi)}{\alpha^2 \sigma^2} \left(1 - \frac{1 - \alpha^{-2s}}{s(1 - \alpha^{-2})} \right).
$$

Finally, we adapt [5, eqn. 11] for our case, to obtain

$$
\mathbb{E}[N(X_{k,s}]_{C^{K-s-1}} \geq d_{k-1}' \] \geq d_{k-1}' \] \geq \frac{N(\xi)}{\alpha^2 \sigma^2} \left(1 - \frac{1 - \alpha^{-2s}}{s(1 - \alpha^{-2})} \right).
$$

The proof is completed by taking limits for averaged values of $d_{k,s}$ and noting that $N(\xi)$ is maximized when $\xi$ is Gaussian.
REFERENCES


