# Design of Successively Refinable Unrestricted Polar Quantizer

Huihui Wu, *Student Member, IEEE* and Sorina Dumitrescu, *Senior Member, IEEE*

***Abstract*—This paper addresses the design of two-stage successively refinable unrestricted polar quantizers for bivariate circularly symmetric sources in the entropy-constrained and fixed-rate cases. The proposed solutions are globally optimal when the thresholds of the magnitude quantizers are confined to finite discretizations of the interval $[0, \infty)$. The algorithm developed for the entropy-constrained case involves a series of stages including solving the minimum-weight path problem for multiple node pairs in certain weighted directed acyclic graphs. The asymptotical time complexity is $O(K_1 K_2^2 P_{max})$, where $K_1$ and $K_2$ are the sizes of the sets of possible magnitude thresholds of the coarse and refined unrestricted polar quantizers (UPQs), respectively, while $P_{max}$ is an upper bound on the number of phase levels in any phase quantizer of the coarse UPQ. The solution algorithm for the fixed-rate case is based on solving a succession of dynamic programming problems for multiple coarse quantizer bins. The time complexity in the fixed-rate case amounts to $O(K_1 K_2 N^2 N_1)$, where $N_1$ is the number of cells of the coarse UPQ and $N$ is the ratio between the number of bins of the fine and coarse UPQs. Extensive experimental results on a bivariate circularly symmetric Gaussian source show the effectiveness of the proposed schemes.**

***Index Terms*—Unrestricted polar quantization, successively refinable quantizer, globally optimal algorithm, minimum-weight path problem, dynamic programming, Monge property.**

## I. Introduction

A polar quantizer quantizes the magnitude and the phase of a two dimensional source vector represented in polar coordinates. The phase quantizer is uniform while the magnitude quantizer may be nonuniform. Therefore, polar quantizers represent a natural choice for the quantization of bivariate sources with circularly symmetric densities, which has been extensively investigated either for the general case or for the specific Gaussian case [1]– [17]. In practice, polar quantization is useful in numerous applications, such as image processing [6], [10], for the encoding of discrete Fourier transform coefficients [1], [2], in holographic image processing [9], as well as for the quantization of sinusoid signals with application in audio coding [11]. More recently, polar quantization was also used for wireless receiver design in [14].

Generally, polar quantizers can be divided into two categories: strictly polar quantizers (SPQs) and unrestricted polar quantizers (UPQs). In SPQ the phase and the magnitude are quantized independently, while in UPQ the phase quantizer depends on the magnitude level. The UPQ is superior to SPQ as shown in [4]. Moreover, the experimental results on bivariate circularly symmetric Gaussian source in [16] demonstrate that the entropy-constrained UPQ strictly outperforms the entropy-constraint scalar quantizer for rates between $0.5$ and $2$ bits/sample with a gap that can reach over $0.1$ dB. The empirical evidence given in [16] also supports the conclusion that the performance of the entropy-constrained UPQ is only $0.135$ to $0.175$ dB away from the asymptotical performance of the two-dimensional entropy-constrained vector quantizer for rates ranging between $2$ and $6$ bits/sample. In the meanwhile, the implementation complexity of polar quantizers is lower than that of vector quantizers, as a polar quantizer consists of two scalar quantizers applied in succession.

A successively refinable quantizer encodes the source into a sequence of embedded bitstreams, which enables the decoder to reconstruct the source in a progressively refinable manner. Specifically, a coarse reconstruction can be obtained by decoding the base layer, while the quality of the reconstruction improves as more refinement layers are decoded. As a promising technique for broadcasting multimedia to heterogeneous devices with fluctuating bandwidth or over unreliable networks, the study of successively refinable source coding has drawn significant attention from the research community [18]– [32]. Notably, the simplified bit plane coding variant of the successively refinable quantization has been adopted as the baseline quantization method of the JPEG 2000 image compression standard [33], [34].

Consequently, it is a matter of interest to investigate the design of successively refinable UPQ. Up to our knowledge, paper [12] is the only work addressing the design of successively refinable UPQ, where only the fixed-rate case is considered. The algorithm of [12] takes the optimal two-level UPQ as the reference UPQ $Q^{(1)}$, and then greedily designs a sequence of successively refinable UPQs $Q^{(k)},\ k \geq 2$, each $Q^{(k)}$ consisting of $2^k$ levels (i.e., with rate of $k/2$ bits/sample). Specifically, the UPQ $Q^{(k)}$ is constructed based on $Q^{(k-1)}$ by choosing the optimal refinement for each magnitude level, i.e., by determining the method achieving the smallest distortion between partitioning in two the magnitude cell or doubling the number of phase levels of the corresponding phase quantizer. Thus, the design approach of [12] has two shortcomings: 1) it is greedy placing a higher priority on the optimization of the coarser UPQs; 2) does not consider the entropy-constrained case.

This paper proposes algorithms for the optimal design of successively refinable UPQs with two refinement stages, referred to as SRUPQs. The proposed solutions overcome the shortcomings of [12] by considering both the fixed-rate and

H. Wu was with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON L8S 4K1, Canada. He is now with the Department of Electrical Engineering, Columbia University, New York, NY 10027, USA (e-mail: hw2712@columbia.edu).

S. Dumitrescu is with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON L8S 4K1, Canada (e-mail: sorina@mail.ece.mcmaster.ca).

This work was supported in part by an NSERC Discovery Grant.

entropy-constrained cases and by aiming at the construction of any SRUPQ which is optimal in the sense of being situated on the lower convex hull of the set of rate-distortion tuples.

In order to apply the algorithms, we first choose two finite sets $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A} \subseteq \mathcal{B}$, which discretize the interval $[0, \infty)$. We design globally optimal SRUPQs under the constraint that the magnitude quantizer thresholds of the coarse and fine UPQs are drawn from the sets $\mathcal{A}$ and $\mathcal{B}$, respectively. Intutively, the finer the discretizations provided by $\mathcal{A}$ and $\mathcal{B}$, the closer the performance of the proposed constructions is to the globally optimal SRUPQs without any constraints on the choice of the magnitude thresholds.

The optimization problem for the entropy-constrained SRUPQ (EC-SRUPQ) is formulated as the minimization of a weighted sum of the distortions and entropies of the coarse and fine component UPQs. This formulation further enables the approach of converting the cost function (after optimizing the portion of the refined UPQ and the phase quantizers corresponding to each possible coarse magnitude bin) to a summation of the costs of individual magnitude bins of the coarse UPQ. Therefore, this problem can be modeled as a minimum-weight path (MWP) problem in a certain weighted directed acyclic graph (WDAG), where each edge represents a possible bin of the coarse magnitude quantizer. To achieve this goal, the proposed solution proceeds in a series of steps including solving the MWP problems for multiple node pairs in another WDAG, which corresponds to the refined UPQ. This process is aided by an efficient algorithm for evaluating the optimal number of phase levels for all possible magnitude bins of the refined UPQ. The overall running time of the solution to the optimal EC-SRUPQ design problem is $O(K_1 K_2^2 P_{max})$, where $K_1$ and $K_2$ are the sizes of the sets $\mathcal{A}$ and $\mathcal{B}$, respectively, and $P_{max}$ is an upper bound on the number of phase levels of the coarse UPQ.

Another contribution of this work lies in the optimal design of fixed-rate SRUPQ (FR-SRUPQ). In the fixed-rate case, the total number of quantization bins $N_1$ and $N_2$ of the coarse and fine UPQs, respectively, are fixed. The proposed solution algorithm allows for the design of the optimal FR-SRUPQ for any values of $N_1$ and $N_2$. The algorithm involves solving a series of dynamic programming problems, where each problem is similar in spirit to the single-resolution fixed-rate UPQ (FRUPQ) design problem [17]. The overall time complexity amounts to $O(K_1 K_2 N^2 N_1)$, where $N = N_2/N_1$, considering that the cost function corresponding to the fine UPQ satisfies the so-called Monge property [17].

We point out that recently we have proposed efficient algorithms for the optimal design of single-resolution entropy-constrained UPQ (ECUPQ) [16] and of FRUPQ [17]. The aforementioned algorithms also ensure global optimality in the case when the thresholds of the magnitude quantizers are restricted to a finite discretization of the interval $[0, \infty)$. The algorithm of [16] relies on solving a single MWP problem in a certain WDAG in conjunction with a procedure to compute the edge weights. The algorithm proposed in the current work for optimal EC-SRUPQ design is much more involved and needs to solve the MWP problem for multiple node pairs. As a result, it also has a higher time complexity than the algorithm of [16], the latter algorithm running in $O(K^2 + KP_0)$ time, where $K$ is the size of the predefined set of possible thresholds and $P_0$ is the maximum number of phase levels in a phase quantizer. Similar observations can be made for the FR-SRUPQ case. The algorithm of [17] for optimal FRUPQ design solves a dynamic programming problem, which is only one component in the proposed FR-SRUPQ design framework. As a consequence, the asymptotic running time also raises from $O(KN_0^2)$ in [17], where $N_0$ is the number of quantizer levels for the single-resolution FRUPQ, to $O(K_1 K_2 N^2 N_1)$ for FR-SRUPQ.

It is also important to discuss the relation between this work and the work on the design of successively refinable scalar quantizers (SRSQ) [22]–[25], [27]. The SRSQ design algorithms in the aforementioned work also include steps resembling solving the MWP problem for multiple node pairs in a WDAG. The connection/similarity with the SRSQ stems from the existence of the embedded partitions of the magnitude quantizers in the SRUPQ. On the other hand, as the SRUPQ is essentially a two-dimensional quantizer, the need to optimize the phase quantizer for each magnitude bin adds an additional level of complexity to the design problem. More specifically, it makes the computation of the edge weights more involved than in the SRSQ case.

We point out that the proposed EC-SRUPQ design algorithm was first presented in the conference paper [35]. In the current work, the description of the proposed algorithm is more refined. This paper additionally includes some key theoretical results (i.e., Propositions 1, 4 and 5) which imply the finiteness of the number of phase regions in the optimal EC-SRUPQ. Besides that, an experimental comparison between the proposed EC-SRUPQ design and the single-description ECUPQ scheme of [16] is provided. Moreover, this work also proposes a design algorithm for the FR-SRUPQ case, which was not included in [35] and which differs significantly from the entropy-constrained case. Accordingly, substantial experimental comparisons for the proposed FR-SRUPQ design are discussed as well.

The rest of the paper is organized as follows. The next section introduces the necessary definitions and notations. Section III formulates the problem of optimal EC-SRUPQ design and presents the proposed solution algorithm. The formulation of the optimal FR-SRUPQ design problem and its solution are presented in Section IV. The experimental results and their discussion follow in Section V and finally Section VI concludes this paper.

## II. Definitions and Notations

Consider a bivariate random variable with the following circularly symmetric density, as a function of the polar coordinates $r$ and $\theta$,

$$p(r, \theta) = \frac{1}{2\pi} g(r), \ 0 \leq r < \infty, \ 0 \leq \theta < 2\pi.$$

Note that $g(r)$ is the marginal probability density function (pdf) of the magnitude variable, while the phase variable is uniformly distributed over the interval $[0, 2\pi)$. Additionally, notice that the magnitude and phase variables are independent.

An example of such a variable is a two-dimensional memoryless Gaussian vector $(X_1, X_2)$, i.e., where $X_1$ and $X_2$ are independent and have identical marginal pdfs. The quantization of Gaussian variables is interesting since it has numerous practical applications. For example, the joint distribution of discrete Fourier transform coefficients of a stationary data sequence is asymptotically Gaussian [2]. Also, the probability density function of the prediction error signal in a differential pulse code modulation coder for moving pictures can be modeled as Gaussian [36].

For any integer $n \geq 2$, an ascending $n$-sequence is an $n$-tuple $\mathbf{r} = (r_0, r_1, r_2, \cdots, r_{n-1})$, with $r_i \in [0, \infty)$, for $0 \leq i \leq n-2$, and $r_{n-1} \in [0, \infty]$, where $r_0 < r_1 < r_2 < \cdots < r_{n-2} < r_{n-1}$. For any $n \geq 2$, $a \in [0, \infty)$ and $b \in [0, \infty]$, with $a < b$, let $\mathcal{S}_n(a, b)$ denote the set of all ascending $n$-sequences such that $r_0 = a$ and $r_{n-1} = b$.

An SRUPQ can be represented as an ordered pair of embedded UPQs $\mathbf{Q} = (Q_1, Q_2)$, where $Q_1$ is the coarse UPQ, while $Q_2$ is the refined UPQ.

Let $M_1$ denote the number of magnitude levels of UPQ $Q_1$ and let $\mathbf{r} = (r_0, r_1, \cdots, r_{M_1})$ denote the ascending sequence corresponding to the thresholds of the magnitude quantizer. For $1 \leq i \leq M_1$, let $C_i$ denote the $i$-th cell (or bin) of the magnitude quantizer, i.e., $C_i = \{r | r_{i-1} \leq r < r_i\}$. Further, denote $\mathbf{P} = (P_1, P_2, \cdots, P_{M_1})$, where $P_i$ is the number of phase regions of the phase quantizer corresponding to $C_i$, $1 \leq i \leq M_1$. Each phase quantizer is uniform, consequently, each quantization bin of the UPQ $Q_1$ can be represented as

$$\mathcal{R}(i,k) = \left\{ re^{\jmath\theta} | r_{i-1} \leq r < r_i, (k-1)\frac{2\pi}{P_i} \leq \theta < k\frac{2\pi}{P_i} \right\},$$

for $1 \leq i \leq M_1$ and $1 \leq k \leq P_i$, where $\jmath$ is the imaginary unit (i.e., $\jmath^2 = -1$). Clearly, the total number of quantization bins of $Q_1$ is $N(Q_1) = \sum_{i=1}^{M_1} P_i$. In this work, we use the squared error as a distortion measure. It is known that, for each $1 \leq i \leq M_1$ and $1 \leq k \leq P_i$, the reconstructed magnitude-phase pair which minimizes the distortion is $A_i e^{\jmath\theta_{i,k}}$ given by [1], [4]

$$\begin{aligned} \theta_{i,k} &= (2k-1)\pi/P_i, \\ A_i &= \text{sinc}\left(\tfrac{1}{P_i}\right) x(C_i), \end{aligned} \tag{1}$$

where $\text{sinc}(\frac{1}{P_i}) = \frac{\sin(\pi/P_i)}{\pi/P_i}$ and for $C \subseteq [0, \infty)$, $x(C) = \frac{\int_C r g(r) dr}{\int_C g(r) dr}$.

The magnitude partition of the refined UPQ $Q_2$ is embedded in the partition $\mathbf{r}$. This means that each cell $C_i$, for $1 \leq i \leq M_1$, is further partitioned into $M_{2,i}$ cells of the magnitude quantizer for $Q_2$. Let us denote by $\mathbf{s}_i = (s_{i,0}, s_{i,1}, \cdots, s_{i,M_{2,i}}) \in \mathcal{S}_{M_{2,i}+1}(r_{i-1}, r_i)$, the ascending sequence of thresholds for this refined partition. We will use the notation $C_{i,j} = [s_{i,j-1}, s_{i,j})$ for $1 \leq i \leq M_1$, and $1 \leq j \leq M_{2,i}$. Let us also denote by $\bar{\mathbf{s}}$ the $M_1$-tuple $(\mathbf{s}_1, \cdots, \mathbf{s}_{M_1})$, and by $\mathbf{M}_2$ the $M_1$-tuple $(M_{2,1}, \cdots, M_{2,M_1})$. The fact that $Q_2$ is a refinement of $Q_1$ implies that the number of phase regions of the phase quantizer corresponding to magnitude level $C_{i,j}$, denoted by $\tilde{P}_{i,j}$, is a multiple of $P_i$, i.e., $\tilde{P}_{i,j} = P_i P_{i,j}$, for some $P_{i,j} \in \mathbb{Z}_+$, where $\mathbb{Z}_+$ denotes the set of positive integers. Further, let us denote $\mathbf{P}_i = (P_{i,1}, P_{i,2}, \cdots, P_{i,M_{2,i}})$, $1 \leq i \leq M_1$, and denote by $\bar{\mathbf{P}}$ the $M_1$-tuple $(\mathbf{P}_1, \cdots, \mathbf{P}_{M_1})$. Accordingly, each quantization bin of the UPQ $Q_2$ can be represented as

$$\mathcal{R}(i,j,k') = \left\{ re^{\jmath\theta} | s_{i,j-1} \leq r < s_{i,j}, (k'-1)\frac{2\pi}{\tilde{P}_{i,j}} \leq \theta < k'\frac{2\pi}{\tilde{P}_{i,j}} \right\},$$

for $1 \leq i \leq M_1$, $1 \leq j \leq M_{2,i}$ and $1 \leq k' \leq \tilde{P}_{i,j}$. The total number of quantization bins of $Q_2$ is then $N(Q_2) = \sum_{i=1}^{M_1} \sum_{j=1}^{M_{2,i}} \tilde{P}_{i,j}$. The optimal reconstructed magnitude-phase pair, for each $1 \leq i \leq M_1$, $1 \leq j \leq M_{2,i}$ and $1 \leq k' \leq \tilde{P}_{i,j}$ is $A_{i,j} e^{\jmath\theta_{i,j,k'}}$, where

$$\begin{aligned} \theta_{i,j,k'} &= (2k'-1)\pi/(\tilde{P}_{i,j}), \\ A_{i,j} &= \text{sinc}\left(\tfrac{1}{\tilde{P}_{i,j}}\right) x(C_{i,j}). \end{aligned} \tag{2}$$

Notice that the tuples $\mathbf{r}$, $\mathbf{P}$, $\bar{\mathbf{s}}$ and $\bar{\mathbf{P}}$ completely specify the SRUPQ.

As pointed out earlier, we use the squared error as a distortion measure. Therefore, the expected distortion (per sample) of $Q_1$ and $Q_2$ can be expressed, respectively, as [1], [4], [6]

$$D(Q_1) = \frac{1}{2}\left( \int_0^\infty r^2 g(r) dr - \sum_{i=1}^{M_1} A_i^2 q(C_i) \right), \tag{3}$$

$$D(Q_2) = \frac{1}{2}\left( \int_0^\infty r^2 g(r) dr - \sum_{i=1}^{M_1} \sum_{j=1}^{M_{2,i}} A_{i,j}^2 q(C_{i,j}) \right), \tag{4}$$

where for $C \subseteq [0, \infty)$, $q(C) = \int_C g(r) dr$.

Let $R(Q_1)$ and $R(Q_2)$ denote the rate (in bits/sample) of $Q_1$ and $Q_2$, respectively. In the EC-SRUPQ case, the rates can be expressed as [16]

$$R(Q_1) = \frac{1}{2} \sum_{i=1}^{M_1} q(C_i)(-\log_2 q(C_i) + \log_2 P_i), \tag{5}$$

$$R(Q_2) = \frac{1}{2} \sum_{i=1}^{M_1} \sum_{j=1}^{M_{2,i}} q(C_{i,j}) \left(-\log_2 q(C_{i,j}) + \log_2(P_i P_{i,j})\right). \tag{6}$$

In the FR-SRUPQ case we have

$$R(Q_1) = \frac{1}{2}\log_2 N(Q_1), \quad R(Q_2) = \frac{1}{2}\log_2 N(Q_2). \tag{7}$$

As mentioned in the introduction, when designing the SRUPQs we will impose the thresholds of the coarse and fine magnitude quantizers to take values in some finite sets $\mathcal{A}$ and $\mathcal{B}$, respectively, where $\mathcal{A} \subseteq \mathcal{B}$. These sets can be obtained by finely discretizing a large enough interval $[0, B]$, chosen such that the probability that the magnitude $r$ is larger than $B$ to be sufficiently small. Let $\mathcal{A} = \{a_1, a_2, \cdots, a_{K_1}\}$ and $\mathcal{B} = \{b_1, b_2, \cdots, b_{K_2}\}$, where $a_i < a_{i+1}$, for $1 \leq i \leq K_1 - 1$, and $b_j < b_{j+1}$, for $1 \leq j \leq K_2 - 1$. Additionally, let us denote $a_0 = b_0 = 0$, $a_{K_1+1} = b_{K_2+1} = \infty$, $\bar{\mathcal{A}} = \mathcal{A} \cup \{a_0, a_{K_1+1}\}$ and $\bar{\mathcal{B}} = \mathcal{B} \cup \{b_0, b_{K_2+1}\}$. Since $\bar{\mathcal{A}} \subseteq \bar{\mathcal{B}}$ it follows that there is an injective mapping $\nu : \{0, 1, \cdots, K_1+1\} \to \{0, 1, \cdots, K_2+1\}$ such that $a_i = b_{\nu(j)}$.

## III. Optimal EC-SRUPQ Design Algorithm

This section first presents the optimal EC-SRUPQ design problem followed by the description of the major steps of solution algorithm, after which the details of the solution at each step are exposed.

### A. Problem Formulation

Let us denote by $\mathfrak{Q}(\mathcal{A},\mathcal{B})$ the set of all SRUPQs such that the thresholds $r_i$ are from the set $\bar{\mathcal{A}}$ and the thresholds $s_{i,j}$ are from the set $\bar{\mathcal{B}}$. Following prior work on entropy-constrained quantization [11], [27], [37] we formulate the problem of optimal EC-SRUPQ design as the minimization of a weighted sum of distortions and entropies, i.e.,

$$\min_{\mathbf{Q}\in\mathfrak{Q}(\mathcal{A},\mathcal{B})} \mathcal{L}_{EC}(\mathbf{Q}), \tag{8}$$

where

$$\mathcal{L}_{EC}(\mathbf{Q}) \triangleq \phi D(Q_1)+(1-\phi)D(Q_2)+\lambda_1 R(Q_1)+\lambda_2 R(Q_2),$$

for some fixed $0<\phi<1$ and $\lambda_1,\lambda_2>0$, and with $R(Q_1)$ and $R(Q_2)$ given in (5) and (6). It is known [38], [39] that any EC-SRUPQ $\mathbf{Q}\in\mathfrak{Q}(\mathcal{A},\mathcal{B})$ for which the quadruple $(R(Q_1),R(Q_2),D(Q_1),D(Q_2))$ lies on the lower boundary of the convex hull of the set of all such quadruples is a solution to problem (8) for some choice of $\phi$, $\lambda_1$ and $\lambda_2$.

### B. Major Steps of The Solution Algorithm

In this subsection, by analyzing the cost function, we will see how the problem can be split into a series of intermediate optimization problems.

Notice that the first terms in (3) and in (4) are both constant, therefore we can remove them from the cost function. Further, by taking into account relations (1), (2), (5) and (6), problem (8) becomes equivalent to minimizing $\mathcal{F}_{EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}},\bar{\mathbf{P}})$, which is given in (9) at the bottom of this page.

By examining the cost function $\mathcal{F}_{EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}},\bar{\mathbf{P}})$ we notice that, for each pair $(i,j)$, the variable $P_{i,j}$ appears only in the term $\eta(C_{i,j},P_i,P_{i,j})$. Thus, $P_{i,j}$ can be optimized separately for fixed $C_{i,j}$ and $P_i$. We do not know beforehand what the final $C_{i,j}$ and $P_i$ will be, but we can compute the optimal $P_{i,j}$ for each possible choice of $C_{i,j}$, i.e., for each interval $[b_m,b_n)$, $0\le m<n\le K_2+1$, and for each possible choice of $P_i$, i.e., for each positive integer $P$. Let us denote by $P^*_{[b_m,b_n),P}$ this optimal $P_{i,j}$, i.e.,

$$P^*_{[b_m,b_n),P} = \arg\min_{P'\in\mathbb{Z}_+} \eta([b_m,b_n),P,P'), \tag{10}$$

where $\mathbb{Z}_+$ denotes the set of positive integers[1]. Note that, if there are more minimizers in (10), the smallest one is taken. Further, let

$$\eta^*([b_m,b_n),P) = \eta([b_m,b_n),P,P^*_{[b_m,b_n),P}). \tag{11}$$

Now replace in $\mathcal{F}_{EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}},\bar{\mathbf{P}})$ $P_{i,j}$ by $P^*_{C_{i,j},P_i}$, for each $1\le i\le M_1$ and $1\le j\le M_{2,i}$, and denote by $\mathcal{F}_{1,EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}})$ the expression obtained. In other words,

$$\mathcal{F}_{1,EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}}) \triangleq \frac{1}{2}\sum_{i=1}^{M_1}\left(\varphi(C_i,P_i)+\sum_{j=1}^{M_{2,i}}\eta^*(C_{i,j},P_i)\right).$$

Since $\mathcal{F}_{EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}},\bar{\mathbf{P}}) \ge \mathcal{F}_{1,EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}})$, problem (8) can be reduced to minimizing $\mathcal{F}_{1,EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}})$. The expression of $\mathcal{F}_{1,EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}})$ indicates that, if the values $\eta^*(C_{i,j},P_i)$ are known for each possible pair $(C_{i,j},P_i)$, then the partition of $C_i$ into cells $C_{i,j}$ can be optimized separately for each pair $(C_i,P_i)$. We will denote by $\mathbf{s}^*(C_i,P_i)$ this optimal partition. We can compute this optimal partition for each possible choice of $C_i$, i.e., for each interval $[a_u,a_v)$, $0\le u<v\le K_1+1$, and for each possible choice of $P_i$, i.e., for each positive integer $P$. In other words, we can find

$$\mathbf{s}^*([a_u,a_v),P) = \arg\min_{M,\mathbf{s}}\sum_{j=1}^{M}\eta^*([s_{j-1},s_j),P), \tag{12}$$

where $\mathbf{s}=(s_0,\cdots,s_M)\in\mathcal{S}_{M+1}(a_u,a_v)\cap\bar{\mathcal{B}}^{M+1}$. Let also $\gamma^*([a_u,a_v),P)$ denote the cost obtained at optimality in (12), i.e.,

$$\gamma^*([a_u,a_v),P) = \sum_{j=1}^{M^*}\eta^*([s^*_{j-1},s^*_j),P), \tag{13}$$

where $\mathbf{s}^*([a_u,a_v),P) = (s^*_0,\cdots,s^*_{M^*})$. By replacing in $\mathcal{F}_{1,EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}})$ each $\mathbf{s}_i$ by the optimal partition $\mathbf{s}^*(C_i,P_i)$, the cost becomes only a function of $\mathbf{r}$ and $\mathbf{P}$ and we denote it by $\mathcal{F}_{2,EC}(\mathbf{r},\mathbf{P})$, i.e.,

$$\mathcal{F}_{2,EC}(\mathbf{r},\mathbf{P}) \triangleq \frac{1}{2}\sum_{i=1}^{M_1}\left(\varphi(C_i,P_i)+\gamma^*(C_i,P_i)\right).$$

Now it can be seen that, if the values $\gamma^*(C_i,P_i)$ are known for all possible pairs $(C_i,P_i)$, then the optimal $P_i$ can be found independently for each $C_i$. Thus, we can evaluate the optimal $P_i$ for each possible choice of $C_i$, i.e., for each interval

[1]The fact that the minimum in (10) exists will be proved in the following subsection as Proposition 1.

$$\mathcal{F}_{EC}(\mathbf{r},\mathbf{P},\bar{\mathbf{s}},\bar{\mathbf{P}}) \triangleq \frac{1}{2}\sum_{i=1}^{M_1}\Bigg(\underbrace{q(C_i)\left(-\phi\ \mathrm{sinc}^2\left(\frac{1}{P_i}\right)x^2(C_i)-\lambda_1\log_2 q(C_i)+(\lambda_1+\lambda_2)\log_2 P_i\right)}_{\varphi(C_i,P_i)}+$$
$$\sum_{j=1}^{M_{2,i}}\underbrace{q(C_{i,j})\left(-(1-\phi)\ \mathrm{sinc}^2\left(\frac{1}{P_iP_{i,j}}\right)x^2(C_{i,j})+\lambda_2\left(-\log_2 q(C_{i,j})+\log_2 P_{i,j}\right)\right)}_{\eta(C_{i,j},P_i,P_{i,j})}\Bigg). \tag{9}$$

$[a_u, a_v)$, $0 \le u < v \le K_1+1$. Let $P^*_{[a_u,a_v)}$ denote this optimal $P_i$, i.e.,

$$P^*_{[a_u,a_v)} = \arg \min_{P\in\mathbb{Z}_+} \left(\varphi([a_u,a_v),P) + \gamma^*([a_u,a_v),P)\right), \tag{14}$$

where the smallest one is taken if there are multiple minimizers[2]. By replacing $P_i$ in $\mathcal{F}_{2,EC}(\mathbf{r},\mathbf{P})$ with $P^*_{C_i}$, we obtain a new cost function which only depends on $\mathbf{r}$,

$$\mathcal{F}_{3,EC}(\mathbf{r}) \triangleq \frac{1}{2}\sum_{i=1}^{M_1}\left(\varphi(C_i,P^*_{C_i}) + \gamma^*(C_i,P^*_{C_i})\right).$$

Thus, the optimization problem reduces to

$$\begin{aligned} &\min_{M_1,\mathbf{r}} \mathcal{F}_{3,EC}(\mathbf{r}) \\ &\text{subject to} \quad r_i \in \mathcal{A}, \quad 1 \le i \le M_1 - 1. \end{aligned} \tag{15}$$

The above discussion suggests the following procedure to solve problem (8). Note that $P_{max}$ is an integer guaranteed to be higher than the number of phase levels in any phase quantizer of the coarse UPQ and it will be discussed in the next section.

Step 1) For each pair $(b_m, b_n)$, $0 \le m < n \le K_2+1$, and each positive integer $P \le P_{max}$, compute $P^*_{[b_m,b_n),P}$ defined in (10).

Step 2) For each pair $(a_u, a_v)$, $0 \le u < v \le K_1+1$, and each positive integer $P \le P_{max}$, compute the best partition $\mathbf{s}^*([a_u,a_v),P)$ defined in (12) and the corresponding cost $\gamma^*([a_u,a_v),P)$ given in (13).

Step 3) For each pair $(a_u, a_v)$, $0 \le u < v \le K_1+1$, compute $P^*_{[a_u,a_v)}$ defined in (14).

Step 4) Solve problem (15).

Next we present the details for solving each step starting with Step 1.

### C. Solution for Step 1

For any $y > 0$, denote $f(y) = -\text{sinc}^2(\frac{1}{y})$ and $h(y) = \ln y$. For fixed $P \in \mathbb{Z}_+$, consider the following minimization problem

$$\min_{P'\in\mathbb{Z}_+} (f(PP') + \delta h(PP')), \tag{16}$$

where $\delta > 0$. We point out that the optimal solution to (16) will not be changed if we replace $h(PP')$ by $h(P')$, since $h(PP') = h(P)+h(P')$ and $P$ is fixed. Then it can be easily verified that $P^*_{[b_m,b_n),P}$ is the optimal solution to problem (16) for $\delta = \frac{\lambda_2}{(1-\phi)x([b_m,b_n))^2\ln 2}$.

*Proposition 1:* For any $\delta > 0$ there is an integer $P'^*_\delta$ achieving the minimum in (16).

*Proof:* For any positive integer $m$, let $S(m)$ denote the point in the plane having coordinates $(h(m), f(m))$. Additionally, for any integer $P \ge 1$, let $\mathcal{U}_P = \{S(PP')|P' \in \mathbb{Z}_+\}$, let $\hat{\mathcal{U}}_P$ denote the lower boundary of the convex hull of $\mathcal{U}_P$ and let $\hat{\mathcal{P}}_P$ denote the set of integers $P'$ such that $S(PP')$ is in $\hat{\mathcal{U}}_P$. It is known [38], [39] that some value $P'^*_\delta$ minimizes the cost in (16) if and only if the point $P'^*_\delta \in \hat{\mathcal{P}}_P$ and the line of slope $-\delta$ passing through $S(PP'^*_\delta)$ is a support line for $\mathcal{U}_P$. The latter condition is equivalent to

$$lslope_P(P'^*_\delta) \le -\delta \le rslope_P(P'^*_\delta), \tag{17}$$

where, for any $P' \in \hat{\mathcal{P}}_P$, $lslope_P(P')$ (respectively, $rslope_P(P')$) denotes the slope of the convex hull edge of $\hat{\mathcal{U}}_P$ situated to the left (respectively, right) of $S(PP')$, if such an edge exists, while $lslope_P(1) = -\infty$. Further, the fact that a value $P'^*_\delta$ satisfying relations (17) exists follows from the fact that $\delta > 0$ and that the slopes of the convex hull edges of $\hat{\mathcal{U}}_P$ approach 0 as the abscissa approaches infinity, result which is proved in the appendix (stated as Lemma 1). ■

The following proposition characterizes the set $\hat{\mathcal{P}}_P$.

*Proposition 2:* $\hat{\mathcal{P}}_P = \begin{cases} \mathbb{Z}_+ \setminus \{2\}, & \text{if } P = 1, \\ \mathbb{Z}_+, & \text{if } P \ge 2. \end{cases}$

*Proof:* It was proved in [16, Proposition 1] that $\hat{\mathcal{P}}_1 = \mathbb{Z}_+ \setminus \{2\}$. Now consider the case $P \ge 2$. If $PP' \ge 3$, then $PP' \in \hat{\mathcal{P}}_1$, therefore $P' \in \hat{\mathcal{P}}_P$. This implies that for $P \ge 3$ we have $\hat{\mathcal{P}}_P = \mathbb{Z}_+$, while for $P = 2$ we have $\mathbb{Z}_+ \setminus \{1\} \subseteq \hat{\mathcal{P}}_P$. The fact that $1 \in \hat{\mathcal{P}}_2$ can be verified easily concluding the proof. ■

The monotonicity property established by the following result will be exploited when computing the value $P^*_{[b_m,b_n),P}$. Although its proof is similar to the proof of [16, Proposition 2], we include it here for the purpose of clarity and self-containment.

*Proposition 3:* For any integers $m, m', n, n'$ such that $0 \le m < n \le K_2+1$, $0 \le m' < n' \le K_2+1$, $m \le m'$ and $n \le n'$, and for any $P \in \mathbb{Z}_+$ the following inequality holds

$$P^*_{[b_m,b_n),P} \le P^*_{[b_{m'},b_{n'}),P}. \tag{18}$$

*Proof:* The discussion at the beginning of this subsection implies that

$$\begin{aligned} lslope_P(P^*_{[b_m,b_n),P}) &\le -\frac{\lambda_2}{(1-\phi)x([b_m,b_n))^2\ln 2} \\ &\le rslope_P(P^*_{[b_m,b_n),P}). \end{aligned} \tag{19}$$

Recall that $x([b_m,b_n))$ denotes the centroid of the interval $[b_m,b_n)$ with respect to the pdf $g(r)$. It is known that $x([b_m,b_n))$ is a non-decreasing function of both $b_m$ and $b_n$[3]. Then, under the conditions specified in the hypothesis, it follows that

$$0 < x([b_m,b_n)) \le x([b_{m'},b_{n'})),$$

which further leads to

$$-\frac{\lambda_2}{(1-\phi)x([b_m,b_n))^2\ln 2} \le -\frac{\lambda_2}{(1-\phi)x([b_{m'},b_{n'}))^2\ln 2},$$

since $\lambda_2$, $\ln 2$ and $1-\phi$ are positive. The above inequality, together with relation (19) and with the fact that the slopes of the lower convex hull edges are non-decreasing as we proceed from left to right, imply inequality (18). This proves the claim. ■

As a consequence, Algorithm 1 in [16] can be utilized to determine all values $P^*_{[b_m,b_n),P}$, for fixed $P$, in $O(K_2P'_{P,max} + K_2^2)$ time, where $P'_{P,max} = P^*_{[b_{K_2},b_{K_2+1}),P}$ is the maximum of

[2]The proof of the fact that the minimum in (14) exists follows the same lines as the proof of Proposition 5.

[3]A proof of this result can be found in [40].

$P^*_{[b_m,b_n),P}$ over all intervals $[b_m, b_n)$, in virtue of Proposition 3. Performing this for all $P, 1 \le P \le P_{max}$, amounts to $O(K_2 \sum_{P=1}^{P_{max}} P'_{P,max} + K_2^2 P_{max})$ operations. In order to find a closed form for the expression of the running time, the following result will be useful. Its proof is deferred to the appendix.

*Proposition 4:* For each integer $P \ge 2$, the following holds

$$P'_{P,max} \le \frac{P'_{1,max}}{P} + 1.$$

The following proposition clarifies how to compute $P_{max}$. Its proof is deferred to the appendix.

*Proposition 5:* Consider $P_{max} = \max\{P'_{1,max} + 1, P''\}$, where $P''$ is the solution to problem (16) for $P = 1$ and $\delta = \frac{\lambda_1}{\phi x([b_{K_2}, b_{K_2+1})^2 \ln 2}$. Then there is an optimal EC-SRUPQ such that the phase quantizer corresponding to any magnitude level of the coarse UPQ has no more than $P_{max}$ levels.

Further, by using $P_{max}$ defined in Proposition 5 and aided by Proposition 4 one obtains

$$\begin{aligned}\sum_{P=1}^{P_{max}} P'_{P,max} &\le P'_{1,max} \sum_{P=1}^{P_{max}} \frac{1}{P} + P_{max} \\ &\le P'_{1,max}(\ln P_{max} + 1) + P_{max} \\ &\le P_{max}(\ln P_{max} + 2),\end{aligned}$$

where the second inequality follows from the known upper bound on the partial sum of the Harmonic series, i.e., $\sum_{P=1}^{P_{max}} \frac{1}{P} \le 1 + \int_1^{P_{max}} \frac{1}{P}\, dP = \ln P_{max} + 1$. Thus, the running time of Step 1 becomes $O(K_2 P_{max}(\ln P_{max} + K_2))$. If $\ln P_{max} < K_2$, which is the case in our experiments, the time complexity amounts to $O(K_2^2 P_{max})$.

### D. Solution for Steps 2-4

For convenience we will jump now to the discussion of Step 4. We will show that problem (15) is equivalent to an MWP problem in the WDAG $G = (V_A, E_A, w)$ where $V_A = \{0, 1, \cdots, K_1 + 1\}$ is the vertex set and $E_A = \{(u, v) \in V_A^2 \mid 0 \le u < v \le K_1 + 1\}$ denotes the edge set. The weight of each edge $(u, v) \in E_A$ is $w(u, v)$ defined as

$$w(u, v) \triangleq \varphi([a_u, a_v), P^*_{[a_u,a_v)}) + \gamma^*([a_u, a_v), P^*_{[a_u,a_v)}).$$

The source node in this graph is 0 and the final node is $K_1+1$. A path in the graph is any sequence of connected edges. Its length is the number of component edges. The weight of a path equals the sum of the weights of its edges. The MWP problem in this graph is the problem of finding the path of minimum weight from the source to the final node. Each ascending sequence $\mathbf{r} \in \mathcal{S}_{M_1+1}(0, \infty)$, with components in $\bar{\mathcal{A}}$, can be associated the path $(z_0, z_1, \cdots, z_{M_1})$ from the source to the final node with $a_{z_i} = r_i$, for each $0 \le i \le M_1$. In other words, the $i$-th edge on this path, which is $(z_{i-1}, z_i)$, corresponds to the $i$-th magnitude level $[r_{i-1}, r_i)$. It can be easily seen that this correspondence is one-to-one and that the weight of the path equals $\mathcal{F}_{3,EC}(\mathbf{r})$, which is the optimization objective in (15). This observation implies that problem (15) is equivalent to the MWP problem in the WDAG $G$. If all edge weights are available, which is the case since they were computed at Step 3, this MWP problem can be solved in $O(|V|+|E|) = O(K_1^2)$ operations.

Next we will consider the problem at Step 2. We will show that for each $P$, the problem is equivalent to multiple MWP problems in another WDAG. For each positive integer $P$, construct the WDAG $G_P = (V, E, w_P)$, where $V = \{0, 1, \cdots, K_2 + 1\}$ is the vertex set and $E = \{(m, n) \in V^2 | 0 \le m < n \le K_2 + 1\}$ is the edge set. For each edge $(m, n) \in E$ define the weight $w_P(m, n)$ as

$$w_P(m, n) \triangleq \eta^*([b_m, b_n), P),$$

where $\eta^*([b_m, b_n), P)$ is defined in (11). Let us fix an arbitrary pair $(u, v)$, $0 \le u < v \le K_1 + 1$, and consider finding the optimal partition $\mathbf{s}^*([a_u, a_v), P)$ of the interval $[a_u, a_v)$, defined in (12). Recall that $[a_u, a_v) = [b_{\nu(u)}, b_{\nu(v)})$. Consider an arbitrary partition $\mathbf{s}$ of $[a_u, a_v)$, consisting of $M$ cells with thresholds in $\bar{\mathcal{B}}$. Then $\mathbf{s}$ corresponds to the path $(t_0, \cdots, t_M)$ in $G_P$ from node $\nu(u)$ to $\nu(v)$ with $b_{t_j} = s_j$, $0 \le j \le M$. Clearly, the weight of the path equals $\sum_{j=1}^{M} \eta^*([s_{j-1}, s_j), P)$, which is the cost function in (12). Therefore, finding $\mathbf{s}^*([a_u, a_v)P)$ is equivalent to finding the MWP path from $\nu(u)$ to $\nu(v)$. In order to find the MWP from $\nu(u)$ to $\nu(v)$ for all $v$ satisfying $u < v \le K_1 + 1$, we will solve the single source MWP problem corresponding to the source $\nu(u)$. This is the problem of finding the MWP from the source to any other graph node reachable from the source and can be solved in $O(|V| + |E|) = O(K_2^2)$ time if each graph edge is computable in constant time. Doing so for each $u$ and $P$ amounts to $O(K_1 K_2^2 P_{max})$ operations.

The problem at Step 3 is straightforward and can be solved in $O(K_1^2 P_{max})$ operations. Additionally, the cumulative probabilities, first and second order moments corresponding to elements of the set $\bar{\mathcal{B}}$ are precomputed and stored in a preprocessing step as in [16], as the set $\bar{\mathcal{B}}$ is finer than $\bar{\mathcal{A}}$, and this requires only $O(K_2)$ operations. Then each $x([b_m, b_n))$, $x([a_u, a_v))$, $q([b_m, b_n))$ and $q([a_u, a_v))$ can be evaluated in constant time.

In conclusion, problem (8) can be solved in $O(K_2 P_{max}(K_1 K_2 + \ln P_{max} + K_2))$ time. Note that if $\ln P_{max} + K_2 \le K_1 K_2$, which we found to be true in our experiments, then the time complexity of the solution is $O(K_1 K_2^2 P_{max})$.

The pseudocode of the solution algorithm to problem (8) is presented in Algorithm 1. We point out that for each $P$, $W(m, n, P)$ denotes the minimum weight of the MWP from node $m$ to node $n$ in $G_P$, which is computed at Step 2, while $\varepsilon(m, n, P)$ records the second last node on this path. Moreover, for the problem at Step 4, we denote by $\hat{W}(v)$ the weight of the MWP from the source node 0 to node $v$ in $G$, and by $\epsilon(v)$ the node preceding $v$ on this optimal path. At the end, the MWPs corresponding to the coarse and fine UPQs can be traced back by utilizing the values of $\epsilon(v)$ and $\varepsilon(\nu(u), \nu(v), P^*_{[a_u,a_v)})$, respectively.

## IV. Optimal FR-SRUPQ Design Algorithm

This section first presents the optimal FR-SRUPQ design problem followed by the solution algorithm based on dynamic programming.

**Algorithm 1:** Solution algorithm for problem (8).

**Preprocessing Stage**
**begin**
  /* *Step 1* */
  **for** $P = 1$ **to** $P_{max}$ **do**
    **for** $n = 1$ **to** $K_2 + 1$ **do**
      $P^*_{[b_{n-1},b_n),P} :=$
       $\min \arg\min_{P' \in \hat{\mathcal{P}}_P} \eta([b_{n-1}, b_n), P, P')$
      $W(n-1, n, P) := w_P(n-1, n)$
      $\varepsilon(n-1, n, P) := n-1$
      **for** $m = n-2$ **down to** $0$ **do**
        $P^*_{[b_m,b_n),P} :=$
        $\min \arg\min_{P^*_{[b_m,b_{n-1}),P} \le P' \le P^*_{[b_{m+1},b_n),P}} \eta([b_m, b_n), P, P')$
        $W(m, n, P) := w_P(m, n)$
        $\varepsilon(m, n, P) := m$
  /* *Step 2* */
  **for** $P = 1$ **to** $P_{max}$ **do**
    **for** $u = 0$ **to** $K_1$ **do**
      **for** $v = u + 1$ **to** $K_1 + 1$ **do**
        **for** $k = \nu(u) + 1$ **to** $\nu(v) - 1$ **do**
          **if** $W(\nu(u), \nu(v), P) >$
           $W(\nu(u), k, P) + W(k, \nu(v), P)$ **then**
            $W(\nu(u), \nu(v), P) :=$
             $W(\nu(u), k, P) + W(k, \nu(v), P)$
            $\varepsilon(\nu(u), \nu(v), P) := k$
  /* *Steps 3 and 4* */
  $\hat{W}(0) = 0$
  **for** $u = 0$ **to** $K_1$ **do**
    **for** $v = u + 1$ **to** $K_1 + 1$ **do**
      **for** $P = 1$ **to** $P_{max}$ **do**
        Evaluate $\gamma^*([a_u, a_v), P)$ using (13)
        Evaluate $P^*_{[a_u,a_v)}$ using (14)
      **if** $\left(\hat{W}(u) + w(u, v) < \hat{W}(v)\right)$ **then**
        $\hat{W}(v) := \hat{W}(u) + w(u, v)$
        $\epsilon(v) := u$
  Restore the vectors $\mathbf{r}$ and $\mathbf{P}$ corresponding to the coarse UPQ $Q_1$
  Restore the vectors $\bar{\mathbf{s}}$ and $\bar{\mathbf{P}}$ corresponding to the fine UPQ $Q_2$

### A. Problem Formulation

In the fixed-rate case, according to (7), the rates $R(Q_1)$ and $R(Q_2)$ are determined by the number of quantization cells $N(Q_1)$ and $N(Q_2)$, respectively. Therefore, the problem of optimal FR-SRUPQ design can be formulated as the constrained problem of minimizing a weighted sum of the distortions with constraints on the number of quantizer levels, i.e.,

$$\begin{aligned} \min_{M_1,\mathbf{r},\mathbf{P},\bar{\mathbf{s}},\bar{\mathbf{P}}} \quad & \phi D(Q_1) + (1-\phi) D(Q_2) \\ \text{subject to} \quad & \sum_{i=1}^{M_1} P_i = N_1, \quad \sum_{j=1}^{M_{2,i}} P_{i,j} = N, \\ & r_i \in \mathcal{A},\ s_{i,j} \in \mathcal{B},\ 1 \le i \le M_1 - 1,\ 1 \le j \le M_2 - 1, \end{aligned} \tag{20}$$

for fixed $1 < \phi < 1$, where $N_1$ and $N_2 = N_1 N$ are the two target values for the numbers of quantization cells of $Q_1$ and $Q_2$, respectively. Recall that $R(Q_i) = \frac{1}{2}\log_2 N_i$ represents the rate (bits/sample) of UPQ $Q_i$, $i = 1, 2$. The constraint $\sum_{j=1}^{M_{2,i}} P_{i,j} = N$ is motivated by the fact that the value $\log_2 N = \log_2 \frac{N_2}{N_1}$ is actually the amount of extra bits appended to each binary index output by the coarse quantizer $Q_1$ to obtain an index of the fine quantizer $Q_2$.

### B. Dynamic Programming Solution

Since the first terms in (3) and (4) are both constant, problem (20) is equivalent to minimizing $\mathcal{F}_{FR}(\mathbf{r}, \mathbf{P}, \bar{\mathbf{s}}, \bar{\mathbf{P}})$, where

$$\mathcal{F}_{FR}(\mathbf{r}, \mathbf{P}, \bar{\mathbf{s}}, \bar{\mathbf{P}}) \triangleq \frac{1}{2} \sum_{i=1}^{M_1} \Bigg( \underbrace{-q(C_i)\phi\ \mathrm{sinc}^2\left(\frac{1}{P_i}\right) x^2(C_i)}_{\varphi'(C_i,P_i)} + (1-\phi) \underbrace{\sum_{j=1}^{M_{2,i}} \left( -q(C_{i,j})\ \mathrm{sinc}^2\left(\frac{1}{P_i P_{i,j}}\right) x^2(C_{i,j}) \right)}_{\xi(C_i,P_i,\mathbf{s}_i,\mathbf{P}_i)} \Bigg). \tag{21}$$

It can be noticed from the above cost function that $\xi(C_i, P_i, \mathbf{s}_i, \mathbf{P}_i)$ can be optimized separately for fixed $C_i$ and $P_i$. Let $\mathbf{s}^*(C_i, P_i)$ denote the optimal partition $\mathbf{s}_i$ and let $\mathbf{P}^*(C_i, P_i)$ denote the optimal $M$-tuple $\mathbf{P}_i$. We can compute them for every possible choice of $C_i$, i.e., for every interval $[a_u, a_v)$, $0 \le u < v \le K_1+1$, and for every possible choice of $P_i$, i.e., for every integer $P, 1 \le P \le N_1$. In other words, we can find $\mathbf{s}^*([a_u, a_v), P)$ and $\mathbf{P}^*([a_u, a_v), P)$, which represent the solution of the following optimization problem

$$\begin{aligned} \min_{M,\mathbf{s},\mathbf{P}'} \quad & \xi([a_u, a_v), P, \mathbf{s}, \mathbf{P}') \\ \text{subject to} \quad & \sum_{j=1}^{M} P'_j = N, \quad \mathbf{P}' \in \mathbb{Z}_+^M, \\ & \mathbf{s} \in \mathcal{S}_{M+1}(a_u, a_v) \cap \bar{\mathcal{B}}^{M+1}. \end{aligned} \tag{22}$$

Further, let $\xi^*([a_u, a_v), P)$ denote the cost achieved at optimality in problem (22). If we replace in (21) $\mathbf{s}_i$ and $\mathbf{P}_i$ by their optimal counterparts, the new cost becomes only a function of $\mathbf{r}$ and $\mathbf{P}$, i.e.,

$$\mathcal{F}_{1,FR}(\mathbf{r}, \mathbf{P}) \triangleq \frac{1}{2} \sum_{i=1}^{M_1} \left( \varphi'(C_i, P_i) + (1-\phi)\xi^*(C_i, P_i) \right),$$

and problem (20) reduces to solving

$$\begin{aligned} \min_{M_1,\mathbf{r},\mathbf{P}} \quad & \mathcal{F}_{1,FR}(\mathbf{r}, \mathbf{P}) \\ \text{subject to} \quad & \sum_{i=1}^{M_1} P_i = N_1, \quad \mathbf{P} \in \mathbb{Z}_+^{M_1}, \\ & \mathbf{r} \in \mathcal{S}_{M_1+1}(0, \infty) \cap \bar{\mathcal{A}}^{M_1+1}. \end{aligned} \tag{23}$$

We conclude that the solution to problem (20) can be broken into the following two steps.

Step 1) For each pair $(a_u, a_v)$, $0 \le u < v \le K_1 + 1$, and each positive integer $P \le N_1$, compute $\mathbf{s}^*([a_u, a_v), P)$,

$\mathbf{P}^*([a_u, a_v), P)$ and $\xi^*([a_u, a_v), P)$ by solving problem (22).

Step 2) Solve problem (23).

Next we discuss the problem at the first step. We will show that problem (22) is similar to the problem of optimal fixed-rate UPQ (FRUPQ) design treated in [17]. To see this, first denote for each positive integers $P$ and $P'$ and for each interval $[b_m, b_n)$, $0 \le m < n \le K_2 + 1$,

$$\omega_{P,P'}(b_m, b_n) \triangleq -\text{sinc}^2\left(\frac{1}{PP'}\right)(x[b_m, b_n))^2\, q([b_m, b_n)).$$

Then the cost in (22), i.e., $\xi([a_u, a_v), P, \mathbf{s}, \mathbf{P}')$, equals $\sum_{j=1}^{M} \omega_{P,P'_j}(s_{j-1}, s_j)$, which can be regarded as a function of the partition $\mathbf{s}$ and the tuple of integers $\mathbf{P}'$. Therefore, we will denote it by $\mathcal{O}_P(\mathbf{s}, \mathbf{P}')$, i.e.,

$$\mathcal{O}_P(\mathbf{s}, \mathbf{P}') \triangleq \sum_{j=1}^{M} \omega_{P,P'_j}(s_{j-1}, s_j).$$

Thus, problem (22) can be rewritten as

$$\begin{aligned} &\min_{M,\mathbf{s},\mathbf{P}'} \mathcal{O}_P(\mathbf{s}, \mathbf{P}') \\ \text{subject to} \quad &\sum_{j=1}^{M} P'_j = N, \quad \mathbf{P}' \in \mathbb{Z}_+^M, \\ &\mathbf{s} \in \mathcal{S}_{M+1}(a_u, a_v) \cap \bar{\mathcal{B}}^{M+1}. \end{aligned} \tag{24}$$

Like in the optimal FRUPQ design problem of [17], we have to optimize a magnitude partition $\mathbf{s}$ and a tuple of positive numbers $\mathbf{P}'$ with a fix sum, where each component of $\mathbf{P}'$ can be regarded as the number of phase levels for the phase quantizer corresponding to a magnitude cell. This can be solved using dynamic programming. In order to do so we have to define some subproblems. Namely, for each integers $u$, $P$, $n$ and $k$ such that $1 \le P \le N_1$, $0 \le u \le K_1 + 1$, $\nu(u) \le n \le K_2 + 1$ and $1 \le k \le N$, denote by $\mathcal{P}_{P,u}(n, k)$, the problem obtained by replacing in (24) $a_v$ by $b_n$ and $N$ by $k$. Let $\hat{\mathcal{O}}_{P,u}(n, k)$ denote the value of the cost function of $\mathcal{P}_{P,u}(n, k)$ at optimality. Thus, with the new notation, problem (24) is $\mathcal{P}_{P,u}(\nu(v), N)$.

As in [17], the solution algorithm relies on the following recurrence relation

$$\hat{\mathcal{O}}_{P,u}(n, k) \triangleq \min_{0 \le t < k} \min_{\nu(u) \le m < n} \left(\hat{\mathcal{O}}_{P,u}(m, t) + \omega_{P,k-t}(b_m, b_n)\right), \tag{25}$$

where $\hat{\mathcal{O}}_{P,u}(\nu(u), 0) = 0$ and $\hat{\mathcal{O}}_{P,u}(\nu(u), t) = \hat{\mathcal{O}}_{P,u}(m, 0) = \infty$, for $t > 0$ and $m \ge 1$.

We point out that for fixed $P$ and $u$, the straightforward dynamic programming solution computes all $\hat{\mathcal{O}}_{P,u}(n, k)$ for $n$ increasing from $\nu(u)$ to $K_2 + 1$ and for $k$ increasing from 1 to $N$, by solving (25) for each $(n, k)$. This process takes $O(K_2^2 N^2)$ time, as there are $O(K_2 N)$ pairs $(k, n)$ in total and solving (25) for one pair $(n, k)$ requires $O(K_2 N)$ time. On the other hand, as in [17], one can exploit the Monge property to reduce the time complexity to $O(K_2 N^2)$. The proof of this fact follows very closely the development in [17, Section IV], therefore we omit it. Finally, the procedure is repeated for all $0 \le u \le K_1$ and $1 \le P \le N_1$, leading to a time complexity of $O(K_1 K_2 N_1 N^2)$ to solve Step 1.

Now let us consider the problem at Step 2. For any pair $(a_u, a_v)$, $0 \le u < v \le K_1 + 1$, and any positive integer $P$, denote

$$\omega'_P(a_u, a_v) \triangleq \frac{1}{2}\left(\phi f(P)\,(x[a_u, a_v))^2\, q[a_u, a_v) + (1 - \phi)\xi^*([a_u, a_v), P)\right). \tag{26}$$

Then problem (23) is equivalent to

$$\begin{aligned} &\min_{M_1,\mathbf{r},\mathbf{P}} \mathcal{O}(\mathbf{r}, \mathbf{P}) \triangleq \sum_{i=1}^{M_1} \omega'_{P_i}(r_{i-1}, r_i), \\ \text{subject to} \quad &\sum_{i=1}^{M_1} P_i = N_1, \quad \mathbf{P} \in \mathbb{Z}_+^{M_1}, \\ &\mathbf{r} \in \mathcal{S}_{M_1+1}(0, \infty) \cap \bar{\mathcal{A}}^{M_1+1}. \end{aligned} \tag{27}$$

The above problem is also similar to the optimal FRUPQ design problem [17] and can be solved using dynamic programming. However, since the weight function defined in (26) is more complex than the counterpart in [17, Equation (3)], the Monge property might not necessarily hold. Thus, solving Step 2 will need $O(K_1^2 N_1^2)$ operations. Specifically, for each pair of positive integers $(v, k)$ with $1 \le v \le K_1 + 1$ and $1 \le k \le N_1$, consider problem $\mathcal{P}(v, k)$ obtained by replacing in (27) $\mathcal{S}_{M_1+1}(0, \infty)$ by $\mathcal{S}_{M_1+1}(0, a_v)$ and $N_1$ by $k$. Thus, problem (23) is equivalent to $\mathcal{P}(K_1 + 1, N_1)$. Additionally, denote by $\hat{\mathcal{O}}(v, k)$ the optimal value of the objective function of $\mathcal{P}(v, k)$. The dynamic programming solution consists of solving all sub-problems $\mathcal{P}(v, k)$, for $1 \le v \le K_1 + 1$ and $1 \le k \le N_1$, using the following recurrence relation

$$\hat{\mathcal{O}}(v, k) \triangleq \min_{0 \le t < k} \min_{0 \le m < v} \left(\hat{\mathcal{O}}(m, t) + \omega'_{n-t}(a_m, a_v)\right),$$

where $\hat{\mathcal{O}}(0, 0) = 0$ and $\hat{\mathcal{O}}(0, t) = \hat{\mathcal{O}}(m, 0) = \infty$, for $t > 0$ and $m \ge 1$.

In conclusion, the time complexity for the proposed FR-SRUPQ design is $O(K_1 N_1 (K_1 N_1 + K_2 N^2))$. Assuming that $N_1 = O(N^2)$ the time complexity becomes $O(K_1 K_2 N_1 N^2)$.

## V. Experimental Results

This section assesses the practical performance of the EC-SRUPQ and FR-SRUPQ design algorithms presented in this paper. The experiments are conducted for a two-dimensional random vector $(X_1, X_2)$, where $X_1$ and $X_2$ are independent and identically distributed Gaussian variables with zero-mean and unit-variance. After conversion to polar coordinates the joint pdf becomes

$$p(r, \theta) = \frac{r}{2\pi} \exp\left(-\frac{r^2}{2}\right), \; 0 \le r < \infty, \; 0 \le \theta < 2\pi,$$

where $r = \sqrt{x_1^2 + x_2^2}$, and $\theta = \tan^{-1}(x_2/x_1)$. It then follows that $g(r) = r \exp(-r^2/2)$.

The sets of possible thresholds $\mathcal{A}$ and $\mathcal{B}$ are obtained by dividing the range $[0, 6]$ into subintervals of size 0.025. In other words, $K_1 = K_2 = 240$ and $a_i = b_i = 0.025i$, for $0 \le i \le K_1$. In this section, the notations $R_i$ and $D_i$ are

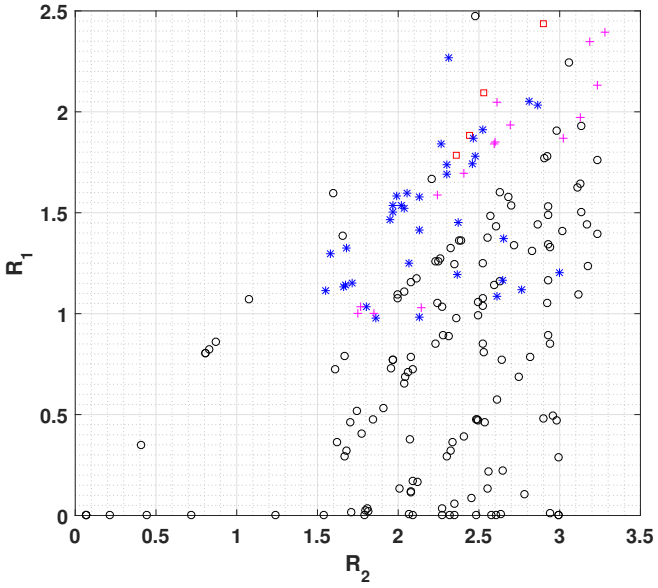



Fig. 1: Rate performance of the proposed EC-SRUPQ.

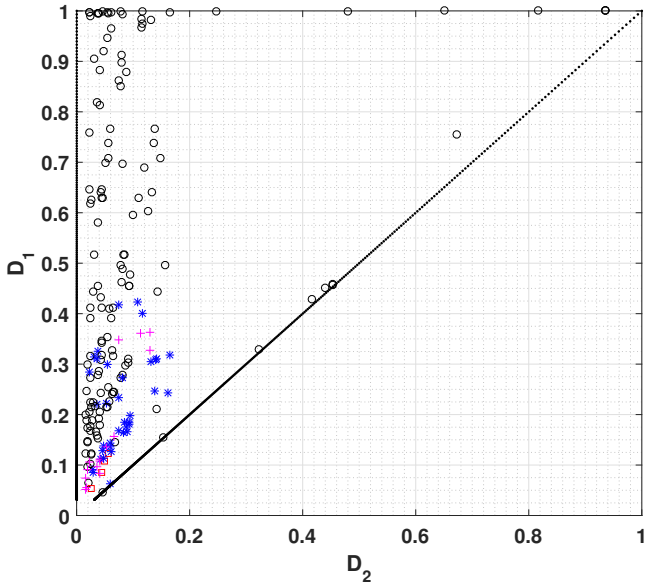



Fig. 2: Distortion performance of the proposed EC-SRUPQ.

utilized instead of $R(Q_i)$ and $D(Q_i)$, respectively, for $i = 1, 2$. Additionally, let $R(D_i)$ denote the rate-distortion function for the Gaussian source, i.e., $R(D_i) = -0.5 \log_2(D_i)$.

We first consider the case of EC-SRUPQ, and compare it with the theoretical bounds. We have run the proposed algorithm for optimal EC-SRUPQ design for the following values of $\phi$, $\phi = 0.03, 0.05, 0.1, 0.15, 0.2, 0.3, 0.5, 0.6, 0.7, 0.8, 0.9$. Figures 1, 2 and 3 illustrate the performance of the proposed EC-SRUPQ, in terms of the rate pair $(R_1, R_2)$, distortion pair $(D_1, D_2)$, and the rate-gap pair $(R_1 - R(D_1), R_2 - R(D_2))$, respectively.

Note that the Gaussian source is known to be successively refinable, i.e., the pair of rates $(R(D_1), R(D_2))$ is achievable by successively refinable code schemes as the block length approaches infinity. Since our scheme uses finite dimension quantization, the existence of a gap to the rate-distortion bound is expected. In particular, the rate gap between the optimum single-description entropy-constrained UPQ and the rate-distortion limit was proved in [11] to be $\frac{1}{2}\log_2 \frac{2\pi e}{12} = 0.2546$ bits/sample at high resolution. Unfortunately, an asymptotic analysis as the rates approach $\infty$ is not available for the EC-SRUPQ, up to our knowledge. However, it is expected that at finite rates the optimal ECUPQ is not successively refinable, i.e., if $R_2 > R_1$, the optimal ECUPQ for rate $R_2$ is not necessarily a refinement of the optimal ECUPQ for $R_1$[4]. This implies that the coarse and fine ECUPQs in an EC-SRUPQ cannot achieve the optimal performance simultaneously. Therefore, it is expected that the rate gap $R_i - R(D_i)$, for $i = 1, 2$, in the case of EC-SRUPQ is larger than its counterpart in the single description case.

It can be noticed from Figure 3 that in most cases the gap $R_2 - R(D_2)$ is within $0.275$ bits/sample, which is very close to the value of $0.2546$ bits/sample. Most of the points in this

[4]This statement is true for the ECUPQ design of [16], as it can be seen from [16, Table II].

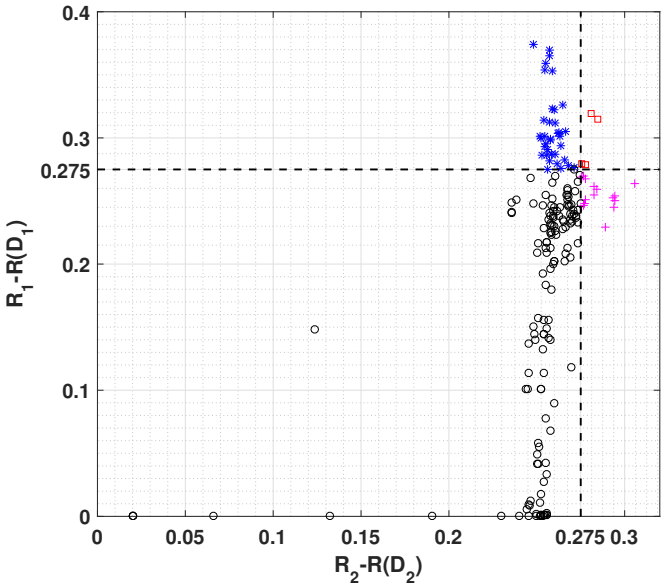



Fig. 3: Gap in rate versus the theoretical lower bounds.

category also have the value of $R_1 - R(D_1)$ within this limit. However, there are also cases in which there is some additional loss either only in $R_1$, or only in $R_2$, and very rarely in both. These cases are represented in the three figures with stars, crosses, and squares, respectively. We can see that the cases with extra loss occur mostly when both distortions are small (see Figure 2). The existence of such extra loss in rate could be attributed to the additional tension induced in the optimization by the competing requirements at the two decoders, as opposed to only one decoder.

It is instructive to compare the performance of the proposed EC-SRUPQ with the single-description ECUPQ designed in [16]. This comparison is presented in Table I for several rate pairs. The notation $(R_1, R_2)$ is for the rate pairs for our

TABLE I: Performance comparison of the proposed EC-SRUPQ with the ECUPQ of [16].

| $(R_1, R_2)$ | $(D_1, D_2)$ | $(\Delta_1, \Delta_2)$ |
|---|---|---|
| $(0.656, 2.036)$ | $(-2.869, -10.706)$ | $(0.004, 0.059)$ |
| $(1.053, 2.242)$ | $(-4.993, -11.945)$ | $(0.006, 0.042)$ |
| $(1.600, 2.629)$ | $(-8.172, -14.183)$ | $(0.012, 0.118)$ |
| $(1.852, 2.598)$ | $(-9.642, -13.872)$ | $(0.021, 0.248)$ |
| $(2.049, 2.611)$ | $(-10.746, -13.878)$ | $(0.093, 0.312)$ |

scheme, while $(D_1, D_2)$ denotes the corresponding distortion pairs in dB, when $\phi = 0.5$. For $i = 1, 2$, we denote $\Delta_i = D_i - D_i^{[16]}$, i.e., the gap in dB between the performance of our scheme at decoder $i$ and the optimal ECUPQ designed using the algorithm of [16] for a rate $R_i^{[16]}$ satisfying $|R_i - R_i^{[16]}| \leq 0.0011$. A very interesting observation is that the values of $\Delta_1$ and $\Delta_2$ increase as the difference $R_2 - R_1$ decreases. This could be attributed to the fact that the condition that the partitions of the coarse and fine UPQ are embedded becomes more restrictive when $R_2 - R_1$ is small, making it more difficult to find embedded partitions close to the optimal ECUPQ partitions. We also see from Table I that the coarse ECUPQ of the proposed scheme performs extremely close to the single-description ECUPQ of [16], especially for $R_1 \leq 1.852$ and $R_2 - R_1 \geq 0.746$. The gap $\Delta_2$ is larger than $\Delta_1$, but we still obtain $\Delta_2 < 0.06$ dB for $R_2 \leq 2.242$ and $R_2 - R_1 \geq 1.189$.

Next we assess the performance of the proposed FR-SRUPQ design algorithm in comparison with the practical successively refinable UPQ scheme developed in [12] based on the asymptotic quantization theory. We ran the proposed algorithm for optimal FR-SRUPQ design for $N_1 = 2^i$, $i = 1, 2, 3, 4, 5$, and $N_2 = 2^j$, $j \in [i+1, 6] \cap \mathbb{Z}_+$. Three values of $\phi$ were examined, namely, $\phi = 0.1, 0.5$ and $0.9$.

The algorithm of [12] starts from the optimal 2-cell UPQ with $\mathbf{r} = (0, +\infty)$ and $\mathbf{P} = (2)$, which is taken as the reference quantizer $Q^{(1)}$. Subsequently, a sequence of successively refinable UPQs $Q^{(k)}$ is constructed greedily, each $Q^{(k)}$ consisting of $2^k$ quantizer levels. More specifically, for $k \geq 2$, the UPQ $Q^{(k)}$ is the best one-bit refinement of $Q^{(k-1)}$. This is achieved by refining either the magnitude (i.e., dividing the magnitude cell into two) or the phase (i.e., doubling the number of phase regions) of each magnitude region of $Q^{(k-1)}$. Thus, for each magnitude level of $Q^{(k-1)}$, one has to choose between the magnitude refinement and the phase refinement the one which gives the smallest distortion. The authors of [12] derived a solution for this choice based on the high-rate approximation, namely, if the inequality

$$\frac{P_i}{2\pi} \geq \left( \frac{r_{i-1}}{r_i - r_{i-1}} + \frac{1}{2} \right)$$

holds for the magnitude cell $[r_{i-1}, r_i)$, then the magnitude refinement is selected, otherwise the phase refinement is chosen. Furthermore, in the case when the magnitude is refined, the new magnitude threshold $r_{i-1} + \gamma$ is determined using an iterative method in the spirit of Max-Lloyd's algorithm [41]. In our implementation of the practical scheme of [12], the optimal value of $\gamma$ is obtained by applying a linear search over the interval $[0, 6]$ with step size $0.001$. Further, we verify that the new threshold $r_{i-1} + \gamma$ satisfies the iterative equations given in [12].

Note that in the sequel the distortion is represented in dB. The distortion pairs of the proposed scheme and of the scheme of [12] are denoted by $(D_1, D_2)^{\phi}$ and $(D_1, D_2)^{[12]}$, respectively. Additionally, the proposed FR-SRUPQ design is also compared against the optimal single-resolution FRUPQ [17], where the notations $(D_1^{[17]}, D_2^{[17]})$ are used ($D_i^{[17]}$ represents the distortion of the optimal FRUPQ [17] with $N_i$ levels, $i = 1, 2$).

We first discuss the results when $N_1 = 2$. In this case, for each $N_2$ we obtain the same pair of distortions $(D_1, D_2)^{\phi}$ for all values of $\phi$ considered. Moreover, when $N_1 = 2$ the coarse UPQ is the same in all three compared scenarios, and consists of only one magnitude cell and two phases in the phase quantizer. Thus, its distortion is $D_1 = -1.664$ dB. The corresponding performance at the refinement stage is illustrated in Figure 4, which plots $D_2$ versus $R_2 = \frac{1}{2} \log_2 N_2$. It can be observed that when $N_2 \geq 8$, our scheme always outperforms the method of [12] with a peak improvement of $0.831$ dB, achieved when $N_2 = 32$. On the other hand, when $N_2 = 4$ both schemes have the same fine UPQ, corresponding to $\mathbf{r} = (0, +\infty)$ and $\mathbf{P} = (4)$, which is also the optimal FRUPQ. Additionally, an important observation is that the refined UPQ of the proposed scheme has performance very close to the optimal single description FRUPQ for all values of $N_2$, with a distortion gap $D_2^{\phi} - D_2^{[17]}$ in the range of $[0, 0.165]$ dB.

Let us consider now the case when $N_1 = 4$. In this case, the coarse UPQ is the same in all three scenarios, corresponding to $\mathbf{r} = (0, +\infty)$, $\mathbf{P} = (4)$ and $D_1 = -4.396$ dB, except for our design at $N_2 = 8$ and $\phi = 0.1$. Moreover, when $N_1 = 4$ and $N_2 \geq 16$, the pair of values $(D_1, D_2)^{\phi}$ does not change with $\phi$. Figure 5 plots the value of $D_2$ versus $R_2 = \frac{1}{2} \log_2 N_2$ for $N_1 = 4$. It can be seen that the proposed design is always superior to the approach of [12] when $N_2 \geq 16$, with an improvement reaching up to $0.723$ dB at $N_2 = 32$. The distortion gap $D_2^{\phi} - D_2^{[17]}$ is in the range of $[0.038, 0.87]$ dB.

As indicated in Figure 5, when $(N_1, N_2) = (4, 8)$ and $\phi = 0.1$, the distortion pair $(D_1, D_2)^{\phi=0.1}$ is $(-3.761, -6.837)$ dB. It is then noted that when $\phi = 0.1$, the coarse UPQ performs worse, while the fine UPQ has better performance, than for the cases of $\phi = 0.5, 0.9$. This is expected, as a small value of $\phi = 0.1$ means that more emphasis is placed on minimizing the distortion of the fine UPQ. Additionally, for the same case, the proposed coarse UPQ performs $0.635$ dB worse than the scheme of [12], while the fine UPQ outperforms the counterpart of [12] by up to $0.794$ dB. Note that the weighted distortion $\phi D_1 + (1 - \phi) D_2$ of the proposed design (which is $-6.411$ dB) is $0.564$ dB smaller than that of [12] (which equals $-5.847$ dB).

Table II illustrates the performance comparison with the scheme of [12] and the FRUPQ of [17], for $N_1 \geq 8$. In this table, the improvement over the scheme of [12] is provided, i.e., $\Delta_i^{\phi} = D_i^{[12]} - D_i^{\phi}$ for $i = 1, 2$. It can be noted that the proposed design outperforms the scheme of [12] for

TABLE II: Performance comparison of the proposed FR-SRUPQ with the design of [12] and with FRUPQ [17], for $N_1 \geq 8$.

| $(N_1,N_2)$ | $(D_1, D_2)^{[12]}$ | $(\Delta_1, \Delta_2)^{\phi=0.1}$ | $(\Delta_1, \Delta_2)^{\phi=0.5}$ | $(\Delta_1, \Delta_2)^{\phi=0.9}$ | $(D_1^{[17]}, D_2^{[17]})$ |
|---|---|---|---|---|---|
| $(8, 16)$ | $(-6.043, -8.882)$ | $(0.513, 0.554)$ | $(0.854, 0.404)$ | $(0.869, 0.341)$ | $(-6.913, -9.614)$ |
| $(8, 32)$ | $(-6.043, -11.430)$ | $(0.759, 0.826)$ | $(0.866, 0.616)$ | $(0.869, 0.604)$ | $(-6.913, -12.340)$ |
| $(8, 64)$ | $(-6.043, -14.783)$ | $(0.865, 0.228)$ | $(0.869, 0.218)$ | $(0.869, 0.218)$ | $(-6.913, -15.150)$ |
| $(16, 32)$ | $(-8.882, -11.430)$ | $(0.349, 0.833)$ | $(0.627, 0.631)$ | $(0.732, 0.257)$ | $(-9.614, -12.340)$ |
| $(16, 64)$ | $(-8.882, -14.783)$ | $(0.721, 0.267)$ | $(0.729, 0.259)$ | $(0.732, 0.247)$ | $(-9.614, -15.150)$ |
| $(32, 64)$ | $(-11.430, -14.783)$ | $(0.428, 0.323)$ | $(0.801, 0.037)$ | $(0.906, -0.297)$ | $(-12.340, -15.150)$ |

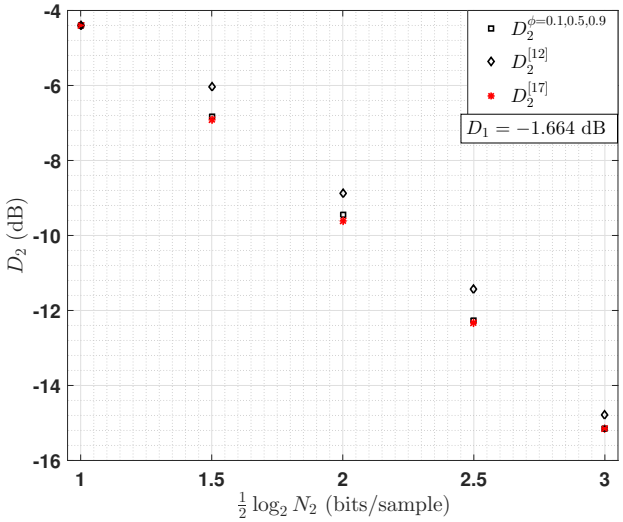



Fig. 4: Distortion $D_2$ versus $\frac{1}{2}\log_2 N_2$ when $N_1 = 2$, in the fixed-rate case.

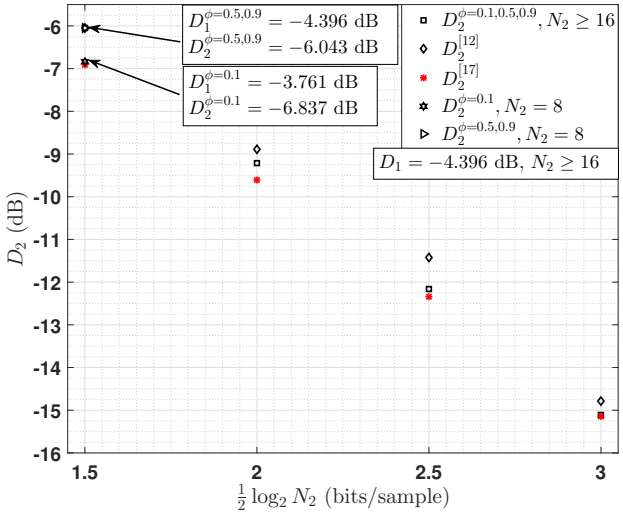
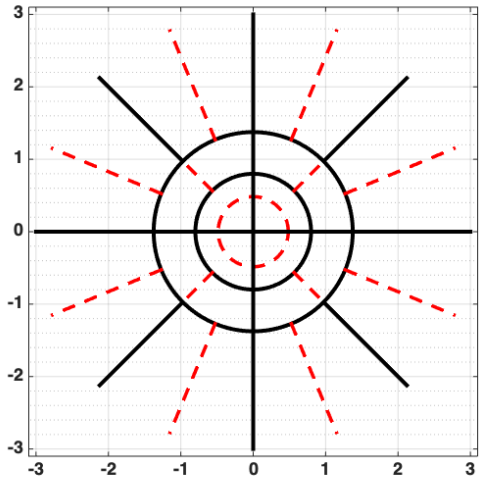



Fig. 5: Distortion $D_2$ versus $\frac{1}{2}\log_2 N_2$ when $N_1 = 4$, in the fixed-rate case.

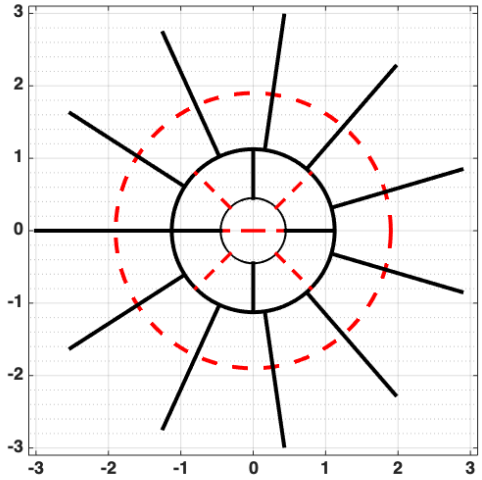
both coarse and fine UPQs in all cases, except for the pair $(N_1, N_2) = (32, 64)$ at $\phi = 0.9$. In most cases, $\Delta_1^\phi \geq 0.5$ dB and $\Delta_2^\phi \geq 0.2$ dB, while the peak improvements reach over $0.8$ dB. As Table II shows, the impact of $\phi$ on the performance of the proposed design is more noticeable for $N_1 \geq 8$. As expected, as $\phi$ increases from $0.1$ to $0.9$, the value of $D_1^\phi$ is non-increasing, while the value of $D_2^\phi$ is non-decreasing. Actually, for all pairs $(N_1, N_2)$ the distortion $D_1^{\phi=0.9}$ is very close to the optimal performance of the $N_1$-level FRUPQ [17].

Figures 6a and 6b illustrate the structures of the UPQ partitions for the scheme of [12] and for the proposed design, respectively, for $(N_1, N_2) = (16, 32)$ and $\phi = 0.1$. In this case, our scheme outperforms the scheme of [12] with improvements of $0.349$ dB and $0.833$ dB for the coarse and fine UPQ, respectively. Note that the partition of the coarse UPQ is represented using solid lines, while dashed lines represent the refinement. Recall that the design of [12] constructs a sequence of successively refinement UPQs in a series of stages, starting from the optimal 2-cell UPQ and applying a one-bit refinement at each stage. Thus, the construction of the $2^k$-level UPQ in this sequence is heavily constrained by the previous UPQs, fact which contributes to limiting its performance. On the other hand, our proposed algorithm does not impose preexisting constraints on the coarse UPQ and thus provides more freedom in the optimization. For instance, we observe in Figure 6b that the cell in the center of the coarse UPQ (solid lines) is a disc, while the course UPQ of [12] does not have such a cell. Additionally, even if the fine UPQ of [12] is obtained by applying the best one-bit refinement (asymptotically) to the coarse UPQ, the preexisting structural constraints severely limit the possible configurations, thus explaining the degradation in performance in comparison with our design.

Further, Figures 7a and 7b depict the partitions of the FR-SRUPQ of [12] and the proposed design, respectively, for the pair $(N_1, N_2) = (32, 64)$ when $\phi = 0.9$. This is the only case in Table II where the performance of the fine UPQ for our scheme is worse than that of [12]. In order to understand why this happens it is instructive to compare first the partitions of the optimal 32-level and 64-level FRUPQs. The optimal FRUPQ [17] for $N = 32$ has $\mathbf{r} = (0, 0.363, 1.031, 1.846, \infty)$ and $\mathbf{P} = (1, 7, 12, 12)$, while for $N = 64$, $\mathbf{r} = (0, 0.536, 0.998, 1.534, 2.234, \infty)$ and $\mathbf{P} = (5, 10, 15, 18, 16)$. We notice that there is a large difference between the tuples of thresholds of the magnitude partitions of the two FRUPQs. We further observe that the partition of the coarse UPQ shown in Figure 7b is very close to the partition of the optimal 32-level FRUPQ. This is because

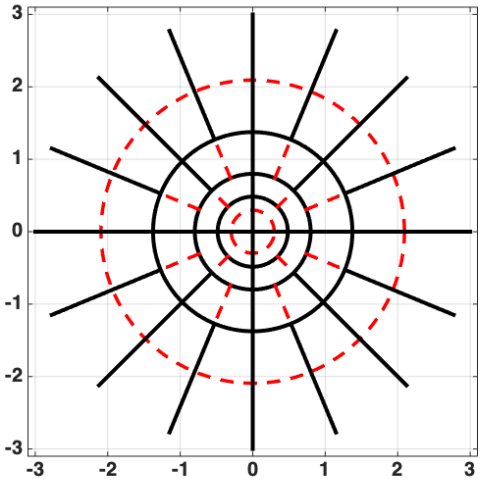


(a) The scheme of [12].

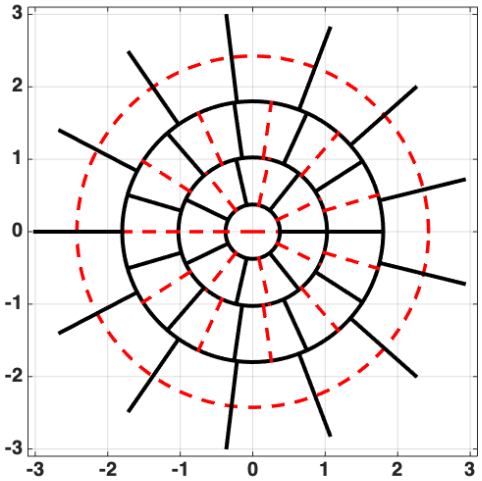


(b) The proposed scheme.

Fig. 6: The partitions of the FR-SRUPQ of [12] (a) and of the proposed scheme (b), for $N_1 = 16$ and $N_2 = 32$. The solid lines represent the partition of coarse UPQ, while the dashed lines represent the refinement. The parameters of the SRUPQs are: (a) $\mathbf{r} = (0, 0.799, 1.375, \infty)$, $\mathbf{P} = (4, 4, 8)$, $\bar{\mathbf{s}} = (0, 0.484, 0.799, 1.375, \infty)$, $\bar{\mathbf{P}} = (4, 4, 8, 16)$; (b) $\mathbf{r} = (0, 0.450, 1.125, \infty)$, $\mathbf{P} = (1, 4, 11)$, $\bar{\mathbf{s}} = (0, 0.450, 1.125, 1.900, \infty)$, $\bar{\mathbf{P}} = (2, 8, 11, 11)$.



(a) The scheme of [12].



(b) The proposed scheme.

Fig. 7: The partitions of the FR-SRUPQ of [12] (a) and of the proposed scheme (b) with $\phi = 0.9$, for $N_1 = 32$ and $N_2 = 64$. The parameters of the SRUPQs are: (a) $\mathbf{r} = (0, 0.484, 0.799, 1.375, \infty)$, $\mathbf{P} = (4, 4, 8, 16)$, $\bar{\mathbf{s}} = (0, 0.297, 0.484, 0.799, 1.375, 2.093, \infty)$, $\bar{\mathbf{P}} = (4, 4, 8, 16, 16, 16)$; (b) $\mathbf{r} = (0, 0.375, 1.025, 1.800, \infty)$, $\mathbf{P} = (1, 7, 11, 13)$, $\bar{\mathbf{s}} = (0, 0.375, 1.025, 1.800, 2.425, \infty)$, $\bar{\mathbf{P}} = (2, 14, 22, 13, 13)$.

$\phi$ is high, thus more emphasis is placed on minimizing the distortion of the coarse UPQ. This implies that the fine UPQ will have magnitude thresholds which are far from those of the optimal 64-level FRUPQ, explaining its poorer performance.

## VI. Conclusion

This paper presents algorithms for the globally optimal design of successively refinable unrestricted polar quantizers for bivariate circularly symmetric sources, for both the entropy-constrained and fixed-rate cases. The global optimality holds when the magnitude quantizers thresholds are confined to some finite sets. For the entropy-constrained case, the cost to be minimized is a weighted sum of distortions and entropies, and the proposed algorithm involves a series of stages including solving the minimum-weight path problem for multiple node pairs in certain weighted directed acyclic graphs. In the fixed-rate case, the proposed solution is based on tackling with a series of dynamic programming problems. The experimental results performed on a bivariate circularly symmetric Gaussian source demonstrate the excellent performance in practice of

the proposed entropy-constrained scheme. For the fixed-rate case, significant improvement can be achieved over the prior practical scheme, which was based on asymptotic analysis.

## REFERENCES


[1] N. C. Gallagher, Jr., "Quantizing schemes for the discrete Fourier transform of a random time-series," *IEEE Trans. Inform. Theory*, vol. IT-24, no. 2, pp. 156-163, Mar. 1978.

[2] W. A. Pearlman and R. M. Gray, "Source coding of the discrete Fourier transform", *IEEE Trans. Inform. Theory,* vol. IT-24, no. 6, pp. 683-692, Nov. 1978.

[3] W. A. Pearlman, "Polar quantization of a complex Gaussian random variable", *IEEE Trans. Commun.,* vol. COM-27, no. 6, pp. 892-899, Jun. 1979.

[4] S. G. Wilson, "Magnitude/phase quantization of independent Gaussian variates", *IEEE Trans. Commun.,* vol. COM-28, no. 11, pp. 1924-1929, Nov. 1980.

[5] P. F. Swaszek and T.W. Ku, "Asymptotic performance of unrestricted polar quantizers," *IEEE Trans. Inform. Theory,* vol. IT-32, no. 2, pp. 330-333, Mar. 1986.

[6] G. H. Senge, "Quantization of image transforms with minimum distortion", *Technical Report No. ECE-77-8,* Dept. of Elec. and Comp. Eng., University of Wisconsin, Madison, WI, Jun. 1997.

[7] D. L. Neuhoff, "Polar quantization revisited," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT 1997),* pp. 60, Ulm, Germany, Jun. 1997.

[8] P. W. Moo and D. L. Neuhoff, "Uniform Polar Quantization Revisited", in *Proc. IEEE Int. Symp. Inform. Theory (ISIT 1998),* pp. 100, Cambridge, MA, USA, Aug. 1998.

[9] A. M. Bruckstein, R. J. Holt and A. N. Netravali, "Holographic representations of images", *IEEE Trans. Image Process.,* vol. 7, no. 11, pp. 1583-1597, Nov. 1998.

[10] N. Kingsbury and T. Reeves, "Redundant representation with complex wavelets: How to achieve sparsity", in *Proc. Int. Conf. Image Process.(ICIP 2003),* Barcelona, Spain, Sep. 2003, pp. 45-48.

[11] R. Vafin and W. B. Kleijn, "Entropy-constrained polar quantization and its application to audio coding", *IEEE Trans. Speech and Audio Process.,* vol. 13, no. 2, pp. 220-232, Mar. 2005.

[12] E. Ravelli and L. Daudet, "Embedded polar quantization", *IEEE Signal Process. Lett.,* vol. 14, no. 10, pp. 657-660, Oct. 2007.

[13] Z. Peric and J. Nikolic, "Design of asymptotically optimal unrestricted polar quantizer for Gaussian source", *IEEE Signal Process. Lett.,* vol. 20, no. 10, pp. 980-983, Oct. 2013.

[14] P. Nazari, B-K. Chun, F. Tzeng and P. Heydari, "Polar quantizer for wireless receivers: theory, analysis, and CMOS implementation", *IEEE Trans. Circuits and Systems-I: Regular Papers,* vol. 61, no. 3, pp. 877-887, Mar. 2014.

[15] A. Z. Jovanovic, Z. H. Peric, J. R. Nikolic and M. R. Dincic, "Asymptotic analysis and design of restricted uniform polar quantizer for Gaussian sources", *Digital Signal Process.,* vol. 49, pp. 24-32, Feb. 2016.

[16] H. Wu and S. Dumitrescu, "Design of optimal entropy-constrained unrestricted polar quantizer for bivariate circularly symmetric sources", *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2169-2180, May. 2018.

[17] H. Wu and S. Dumitrescu, "Design of optimal fixed-rate unrestricted polar quantizer for bivariate circularly symmetric sources", *IEEE Signal Process. Lett.,* vol. 25, no. 5, pp. 715-719, May. 2018.

[18] W. Equitz and T. Cover, "Successive refinement of information," *IEEE Trans. Inform. Theory,* vol. 37, no. 2, pp. 269275, Mar. 1991.

[19] B. Rimoldi, "Successive refinement of information: characterization of achievable rates," *IEEE Trans. Inform. Theory,* vol. 40, no. 1, pp. 253-259, Jan. 1994.

[20] H. Brunk and N. Farvardin, "Fixed-rate successively refinable scalar quantizers," in *Proc. Data Compression Conf. (DCC),* Snowbird, Utah, Mar. 1996, pp. 250-259.

[21] H. Jafarkhani and V. Tarokh, "Design of successively refinable trellis-coded quantizers," *IEEE Trans. Inform. Theory,* vol. 45, no. 5, pp. 1490-1497, Jul. 1999.

[22] D. Muresan and M. Effros, "Quantization as histogram segmentation: globally optimal scalar quantizer design in network systems," in *Proc. Data Compress. Conf. (DCC)*, Snowbird, UT, Apr. 2002, pp. 302-311.

[23] X. Wu and S. Dumitrescu, "On optimal multi-resolution scalar quantization", in *Proc. Data Compress. Conf. (DCC)*, Snowbird, UT, Apr. 2002, pp. 322-331.

[24] S. Dumitrescu and X. Wu, "Optimal multiresolution quantization for scalable multimedia coding," in *Proc. IEEE Infomation Theory Workshop (ITW 2002),* Bangalore, India, Oct. 2002, pp. 139-142.

[25] S. Dumitrescu and X. Wu, "Algorithms for optimal multi-resolution quantization," *J. Algorithms,* vol. 50, no. 1, pp. 1-22, Jan. 2004.

[26] M. Effros and D. Dugatkin, "Multiresolution vector quantization," *IEEE Trans. Inform. Theory,* vol. 50, no. 12, pp. 3130-3145, Dec. 2004.

[27] D. Muresan and M. Effros, "Quantization as histogram segmentation: optimal scalar quantizer design in network systems," *IEEE Trans. Inform. Theory,* vol. 54, no. 1, pp. 344-366, Jan. 2008.

[28] J. Chen, S. Dumitrescu, Y. Zhang and J. Wang, "Robust multiresolution coding," *IEEE Trans. Commun,* vol. 58, no. 11, pp. 3186-3195, Nov. 2010.

[29] C-Y. Wang and M. Gastpar, "On distributed successive refinement with lossless recovery," *IEEE Int. Symp. Inform. Theory (ISIT),* Honolulu, HI, Jun. 2014, pp. 2669-2673.

[30] A. No, A. Ingber and T. Weissman, "Strong Successive Refinability and Rate-Distortion-Complexity Tradeoff," *IEEE Trans. Inform. Theory,* vol. 62, no. 6, pp. 3618-3635, Jun. 2016.

[31] L. Zhou, V. Y. Tan and M. Motani, "Second-order and moderate deviations asymptotics for successive refinement," *IEEE Trans. Inform. Theory,* vol. 63, no. 5, pp. 2896-2921, May. 2017.

[32] V. Kostina and E. Tuncel, "The rate-distortion function for successive refinement of abstract sources," *IEEE Int. Symp. Inform. Theory (ISIT),* Aachen, Germany, Jun. 2017, pp. 1923-1927.

[33] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Process. Magazine,* vol. 18, no. 5, pp. 36-58, Sep. 2001.

[34] D. Taubman and M. Marcellin, *JPEG2000 image compression fundamentals, standards and practice*, Springer, 2012.

[35] H. Wu and S. Dumitrescu, "Design of optimal entropy-constrained successively refinable unrestricted polar quantizer for bivariate circularly symmetric sources", in *Proc. 29th Biennial Symposium on Communications (BSC 2018),* Toronto, Canada, Jun. 2018.

[36] P. Vogel, "Source coding by classification-91-715," *IEEE Trans. Commun.,* vol. 43, no. 11, pp. 2821-2832, Nov. 1995.

[37] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Entropy-constrained vector quantization," *IEEE Trans. Acoust., Speech, Signal Process.,* vol. 37, no. 1, pp. 31-42, Jan. 1989.

[38] H. Everett III, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Operat. Res.,* vol. 11, no. 3, pp. 399-417, Jun. 1963.

[39] D. G. Luenberger, *Optimization by Vector Space Methods*. New York: Wiley, 1969.

[40] A. V. Trushkin, "Sufficient conditions for uniqueness of a locally optimal quantizer for a class of convex error weighting functions", *IEEE Trans. Inform. Theory,* vol. 28, no. 2, pp. 187-198, Mar. 1982.

[41] J. Max, "Quantizing for minimum distortion", *IRE Trans. Inform. Theory*, vol. IT-6, no. 1, pp. 7-12, Mar. 1960.


## APPENDIX A
## PROOFS

In order to prove Proposition 4, we need the following lemmas.

*Lemma 1:* $\lim_{P'\to\infty} rslope_P(P') = 0$.

*Proof:* According to Proposition 2, if $P' \geq 3$ then $P'$ and $P'+1$ are both in $\hat{\mathcal{P}}_P$. Thus, $rslope_P(P')$ is the slope of the line connecting $S(PP')$ and $S(P(P'+1))$. Further, we obtain

$$\begin{aligned}
\lim_{P'\to\infty} rslope_P(P') &= \lim_{P'\to\infty} \frac{f(PP'+P) - f(PP')}{h(PP'+P) - h(PP')} \\
&= \lim_{P'\to\infty} \frac{(\mathrm{sinc}(\frac{1}{PP'}) + \mathrm{sinc}(\frac{1}{PP'+P}))(\mathrm{sinc}(\frac{1}{PP'}) - \mathrm{sinc}(\frac{1}{PP'+P}))}{\ln(P'+1) - \ln P'} \\
&= -2 \lim_{P'\to\infty} \frac{\mathrm{sinc}(\frac{1}{PP'+P}) - \mathrm{sinc}(\frac{1}{PP'})}{\ln(P'+1) - \ln P'},
\end{aligned}$$

where the last equality is based on the fact that $\lim_{P'\to\infty}(\mathrm{sinc}(\frac{1}{PP'}) + \mathrm{sinc}(\frac{1}{PP'+P})) = 2$. Further, in light of Cauchy's mean value theorem, since functions $\mathrm{sinc}(\cdot)$ and $\ln(\cdot)$ are both continuous on $[PP', PP'+P]$, and differentiable

on the open interval $(PP', PP'+P)$, there exists some $Q \in (PP', PP'+P)$ such that

$$\frac{\text{sinc}(\frac{1}{PP'+P}) - \text{sinc}(\frac{1}{PP'})}{\ln(P'+1) - \ln P'} = \frac{(\text{sinc}(\frac{1}{Q}))'}{(\ln Q)'}.$$

Therefore, it is sufficient to prove that $\lim_{Q\to\infty} \frac{(\text{sinc}(\frac{1}{Q}))'}{(\ln Q)'} = 0$. For this, note that we have the following sequence of relations

$$\begin{aligned}\lim_{Q\to\infty} \frac{(\text{sinc}(\frac{1}{Q}))'}{(\ln Q)'} &= \lim_{Q\to\infty} \frac{\frac{1}{\pi}\sin(\frac{\pi}{Q}) - \frac{1}{Q}\cos(\frac{\pi}{Q})}{\frac{1}{Q}} \\ &= \lim_{Q\to\infty} \left(\frac{\sin(\frac{\pi}{Q})}{\frac{\pi}{Q}} - \cos(\frac{\pi}{Q})\right) \\ &= 0.\end{aligned}$$

Thus, the proof is completed. ∎

*Lemma 2:* For any two points $A$ and $B$ in the plane we use the notation $slope(AB)$ for the slope of the line connecting $A$ and $B$. Let $P_i \in \hat{\mathcal{P}}_1$ for $1 \le i \le 4$, such that $P_1 < P_2$, $P_3 < P_4$, $P_1 \le P_3$ and $P_2 \le P_4$. Then one has $slope(S(P_1)S(P_2)) \le slope(S(P_3)S(P_4))$.

*Proof:* Consider the function $t : [0,\infty) \to \mathbb{R}$ such that for each $x \ge 0$ the pair $(x, t(x))$ is the unique point with abscissa $x$ situated on the lower convex hull of $\mathcal{U}_1$. Then, clearly, function $t$ is convex. For each $1 \le i \le 4$, let $x_i = h(P_i)$. Then $f(P_i) = t(x_i)$ and the claim follows in virtue of the following lemma. ∎

*Lemma 3:* Let $t : \mathbb{R} \to \mathbb{R}$ be a convex function and $x_1 < x_2$, $x_3 < x_4$, $x_1 \le x_3$ and $x_2 \le x_4$. Then the following holds

$$\frac{t(x_2) - t(x_1)}{x_2 - x_1} \le \frac{t(x_4) - t(x_3)}{x_4 - x_3}. \tag{28}$$

*Proof:* Let us assume that $x_2 \ne x_3$. In order to prove (28), we will prove the following two inequalities

$$\frac{t(x_2) - t(x_1)}{x_2 - x_1} \le \frac{t(x_3) - t(x_2)}{x_3 - x_2}, \tag{29}$$

$$\frac{t(x_3) - t(x_2)}{x_3 - x_2} \le \frac{t(x_4) - t(x_3)}{x_4 - x_3}. \tag{30}$$

To prove (29) we will consider separately the cases 1) $x_2 < x_3$ and 2) $x_3 < x_2$. First note that by performing some algebraic manipulations, (29) becomes

$$\frac{t(x_2)(x_3 - x_1) - t(x_1)(x_3 - x_2) - t(x_3)(x_2 - x_1)}{(x_2 - x_1)(x_3 - x_2)} \le 0. \tag{31}$$

In case 1) one has $x_1 < x_2 < x_3$. Thus $(x_2 - x_1)(x_3 - x_2) > 0$ and (31) becomes equivalent to $t(x_2)(x_3 - x_1) - t(x_1)(x_3 - x_2) - t(x_3)(x_2 - x_1) \le 0$, which is further equivalent to

$$t(x_2) \le t(x_1)\frac{x_3 - x_2}{x_3 - x_1} + t(x_3)\frac{x_2 - x_1}{x_3 - x_1}. \tag{32}$$

Denote $\rho = \frac{x_3 - x_2}{x_3 - x_1}$. Then $0 < \rho < 1$, $\frac{x_2 - x_1}{x_3 - x_1} = 1 - \rho$ and $x_2 = \rho x_1 + (1-\rho)x_3$. Thus, (32) is equivalent to $t(\rho x_1 + (1-\rho)x_3) \le \rho t(x_1) + (1-\rho)t(x_3)$, which is true in virtue of the convexity of function $t$.

Let us consider now case 2). Then $x_1 \le x_3 < x_2$. If $x_1 = x_3$ then (29) holds trivially with equality. Assume now that $x_1 < x_3$. Then $(x_2 - x_1)(x_3 - x_2) < 0$ and (31) becomes equivalent to $t(x_2)(x_3 - x_1) + t(x_1)(x_2 - x_3) - t(x_3)(x_2 - x_1) \ge 0$, which is equivalent to

$$t(x_3) \le t(x_1)\frac{x_2 - x_3}{x_2 - x_1} + t(x_2)\frac{x_3 - x_1}{x_2 - x_1}. \tag{33}$$

If we let $\rho = \frac{x_2 - x_3}{x_2 - x_1}$ then $0 < \rho < 1$ and inequality (33) is equivalent to $t(\rho x_1 + (1-\rho)x_2) \le \rho t(x_1) + (1-\rho)t(x_2)$, which holds since $t$ is convex. With this observation the proof of (29) is complete. The proof of (30) follows along similar lines. Clearly, (29) and (30) further imply (28). In the case when $x_2 = x_3$ the proof of (28) is analogous to the proof of (29) in case 1). These considerations complete the proof of the lemma. ∎

*Lemma 4:* $slope(S(1)S(3)) \le slope(S(2)S(4))$.

*Proof:* By using the definition of $S(P)$, after some algebraic manipulations we obtain that the above inequality is equivalent to $\frac{-27}{4\pi^2 \ln 3} \le \frac{-4}{\pi^2 \ln 2}$. This is further equivalent to $27 \ln 2 \ge 16 \ln 3$. By applying the exponential function this becomes equivalent to $2^{27} \ge 3^{16}$. The latter relation is true since $2^{27} = (2^5)^5 \times 2^2$, $3^{16} = (3^3)^5 \times 3$, while $2^5 > 3^3$ and $2^2 > 3$. ∎

*Proof of Proposition 4:* Let $\delta = \frac{\lambda_2}{(1-\phi)x([b_{K_2}, b_{K_2+1}))^2 \ln 2}$. Let $P^*$ denote $P'_{1,max}$ and $P^*_P$ denote $P'_{P,max}$. Assume first that $P^* \ge 3$. Then, according to Proposition 2, $P^* + 1 \in \hat{\mathcal{P}}_1$ and based on relation (17), the following holds

$$-\delta \le slope(S(P^*)S(P^*+1)). \tag{34}$$

Note that $P^* \le P\lceil \frac{P^*}{P} \rceil$ and $P^* + 1 \le P\lceil \frac{P^*}{P} \rceil + P$ and, based on Proposition 2, $P^*, P\lceil \frac{P^*}{P} \rceil, P^*+1$ and $P\lceil \frac{P^*}{P} \rceil + P$ are in $\hat{\mathcal{P}}_1$. Thus, we can apply Lemma 2 with $P_1 = P^*$, $P_2 = P^*+1$, $P_3 = P\lceil \frac{P^*}{P} \rceil$ and $P_4 = P\lceil \frac{P^*}{P} \rceil + P$ and obtain

$$slope(S(P^*)S(P^*+1)) \le slope(S(P\left\lceil \frac{P^*}{P} \right\rceil)S(P\left\lceil \frac{P^*}{P} \right\rceil + P)).$$

The above equation together with (34) implies that

$$-\delta \le slope(S(P\left\lceil \frac{P^*}{P} \right\rceil)S(P\left\lceil \frac{P^*}{P} \right\rceil + P)). \tag{35}$$

Recall that $P^*_P$ is the smallest integer in $\hat{\mathcal{P}}_P$ such that

$$-\delta \le rslope_P(P^*_P) \le slope(S(PP^*_P)S(PP^*_P + P)). \tag{36}$$

Corroborating the above observation with relation (35) and with the fact that $\lceil \frac{P^*}{P} \rceil \in \hat{\mathcal{P}}_P$ (since $\hat{\mathcal{P}}_P = \mathbb{Z}_+$ by Proposition 2) and that the slopes of the convex hull of $\mathcal{U}_P$ increase from left to right, we conclude that $P^*_P \le \lceil \frac{P^*}{P} \rceil < \frac{P^*}{P} + 1$.

It remains to consider now the case when $P^* = 1$. Then relation (34) has to be replaced by $-\delta \le slope(S(1)S(3))$. Assume now that $P \ge 3$. We can apply Lemma 2 with $P_1 = 1$, $P_2 = 3$, $P_3 = P$ and $P_4 = 2P$ and obtain that $slope(S(1)S(3)) \le slope(S(P)S(2P))$. This implies that $-\delta \le slope(S(P\cdot 1)S(P\cdot 2))$. Using further (36) we conclude that $P^*_P \le 1 < \frac{P^*}{P} + 1$. Consider now $P = 2$. Since $P \notin \hat{\mathcal{P}}_1$ we can no longer apply Lemma 2 as above. However, we still obtain $slope(S(1)S(3)) \le slope(S(P\cdot 1)S(P\cdot 2))$ according to Lemma 4. Then we conclude as above that $P^*_P = 1 < \frac{P^*}{P} + 1$. Thus, the proof is complete. ∎

In order to prove Proposition 5, we need the following lemma.

*Lemma 5:* Consider $P \in \mathbb{Z}_+$ and let $P^*$ denote the solution to problem (16). Then for any $P_1, P_2 \in \hat{\mathcal{P}}_P$ such that $P^* \leq P_1 < P_2$ one has

$$f(PP_1) + \delta h(PP_1) \leq f(PP_2) + \delta h(PP_2).$$

*Proof:* Note that since $h(PP_1) < h(PP_2)$, the above inequality is equivalent (after some algebraic manipulations) to

$$-\delta \leq slope(S(PP_1)S(PP_2)). \tag{37}$$

Since $P^*, P_1, P_2 \in \hat{\mathcal{P}}_P$ and $P^* \leq P_1 < P_2$, an argument similar to the proof of Lemma 2 implies that $rslope_P(P^*) \leq slope(S(PP_1)S(PP_2))$. The definition of $P^*$ leads that $-\delta \leq rslope_P(P^*)$. Combining the last two inequalities proves relation (37). This completes the proof. ∎

*Proof of Proposition 5:* It is sufficient to prove that, if an EC-SRUPQ has $P_i > P_{max}$ for some $i$, then by replacing $P_i$ by $P_{max}$ the cost defined in (9) does not increase. Note that the portion of the cost affected by $P_i$ is $c(C_i, P_i) = \alpha(C_i, P_i) + \sum_{j=1}^{M_{2,i}} \beta(C_{i,j}, P_i, P_{i,j})$, where

$$\alpha(C_i, P_i) = q(C_i)\phi x^2(C_i)\left(f(P_i) + \frac{\lambda_1}{\phi x^2(C_i)\ln 2} h(P_i)\right),$$

$$\beta(C_{i,j}, P_i, P_{i,j}) = q(C_{i,j})(1-\phi)x^2(C_{i,j})\left(f(P_iP_{i,j}) + \frac{\lambda_2}{(1-\phi)x^2(C_{i,j})\ln 2} h(P_iP_{i,j})\right).$$

Let $P_c^*$ denote the solution to problem (16) for $P = 1$ and $\delta = \frac{\lambda_1}{\phi x^2(C_i)\ln 2}$. According to [16, Proposition 2], one has $P_c^* \leq P''$. Thus, $P_c^* \leq P_{max}$. By applying further Lemma 5 and the fact that $q(C_i)\phi x^2(C_i) > 0$, one obtains that $\alpha(C_i, P_i) \geq \alpha(C_i, P_{max})$.

Further, let $P_f^*$ denote the solution to problem (16) for $P = P_{i,j}$ and $\delta = \frac{\lambda_2}{(1-\phi)x^2(C_{i,j})\ln 2}$. According to Proposition 3, one has $P_f^* \leq P'_{P_{i,j},max}$. Using further Proposition 4 one obtains $P'_{P_{i,j},max} \leq \frac{P'_{1,max}}{P_{i,j}} + 1 \leq P'_{1,max} + 1$. Since $P_{max} \geq P'_{1,max} + 1$, one concludes that $P_f^* \leq P_{max}$. By applying Lemma 5 leads to $\beta(C_{i,j}, P_i, P_{i,j}) \geq \beta(C_{i,j}, P_{max}, P_{i,j})$, which concludes the proof. ∎

PLACE PHOTO HERE

**Huihui Wu** (S'14) received the B.Sc. degree in communication engineering from Southwest University for Nationalities, Chengdu, China, in 2011, and the M.S. degree in communication engineering from Xiamen University, Xiamen, China, in 2014. He received the Ph.D. degree in electrical and computer engineering from McMaster University, Hamilton, Canada. His research interests include channel coding, joint source and channel coding, multiple description coding, and signal quantization.

PLACE PHOTO HERE

**Sorina Dumitrescu** (M'05-SM'13) received the B.Sc. and Ph.D. degrees in mathematics from the University of Bucharest, Romania, in 1990 and 1997, respectively. From 2000 to 2002 she was a Postdoctoral Fellow in the Department of Computer Science at the University of Western Ontario, London, Canada. Since 2002 she has been with the Department of Electrical and Computer Engineering at McMaster University, Hamilton, Canada, where she held a Postdoctoral and a Research Associate position, and where she is currently an Associate Professor. Her current research interests include multimedia coding and communications, network-aware data compression, joint source-channel coding, signal quantization. Her earlier research interests were in formal languages and automata theory. She was a recipient of the NSERC University Faculty Award during 2007-2012.