SCALABLE DISTRIBUTED SOURCE CODING

Ankur Saxena and Kenneth Rose

Department of Electrical & Computer Engineering
University of California, Santa Barbara, CA-93106, USA
Email: {ankur,rose}@ece.ucsb.edu

ABSTRACT

This paper considers the problem of *scalable* distributed coding of correlated sources that are communicated to a central unit. The general setting is typically encountered in sensor networks. The conditions of communication channels between the source encoders and fusion center may be time-varying and it is often desirable to guarantee a base layer of coarse information during channel fades. In addition the desired system should be *robust* to various scenarios of channel failure and should utilize all the available information to attain the best possible compression efficiency. Although a standard 'Lloyd-style' distributed coder design algorithm can be generalized to scalable distributed source coding, the resulting algorithm depends heavily on initialization and will virtually always converge to a poor local minimum on the distortion-cost surface. We propose an efficient initialization scheme for such a system, which employs a properly designed multi-stage distributed coder. In our prior work, we highlighted the fundamental conflict that arises when multi-stage coding is directly combined with distributed quantization. Here we use the multi-stage distributed coding system to initialize a scalable distributed coder and propose an iterative algorithm for joint design of all system components once the structural constraint is removed. Simulation results show considerable gains over randomly initialized scalable distributed coder design.

Index Terms— Distributed coding, scalable coding, sensor networks.

1. INTRODUCTION

Distributed source coding (DSC) [7, 10] has witnessed a significant revival of interest with sensor networks as a prominent application area. Review papers, e.g., [3] and [11] describe a variety of strategies for distributed coding system design. The basic setting in DSC involves multiple correlated sources (e.g., spatially distributed sensors) transmitting information to a fusion center without communicating amongst themselves. The main objective in DSC is to exploit inter-source correlations despite the fact that each sensor source is encoded without access to other sources. The only information available to a source encoder about the other sources is via joint statistics (typically extracted from training data).

The communication channels in a sensor field may vary in capacity due to the presence of obstacles or other phenomena such as fading. In such a scenario, it will be beneficial to convey a minimal amount of information even when the channel deteriorates. This motivates the problem of scalable distributed source coding (S-DSC) or successive refinement of distributed correlated sources (see Fig. 1), which generalizes the traditional problem of scalable coding of single source [1]. Successive refinement for Wyner-Ziv coding (side information at the decoder) was proposed in [8], and has been studied in [8, 9] from the information-theoretic perspective of characterizing achievable rate-distortion regions. In this paper, we derive practical iterative algorithms for the design of successive-refinable system within the multi-terminal (distributed) setting.

The work was supported in part by the NSF under grants IIS-0329267 and CCF-0728986, the University of California MICRO program, Applied Signal Technology Inc., Cisco Systems Inc., Dolby Laboratories Inc., Qualcomm Inc., and Sony Ericsson, Inc.

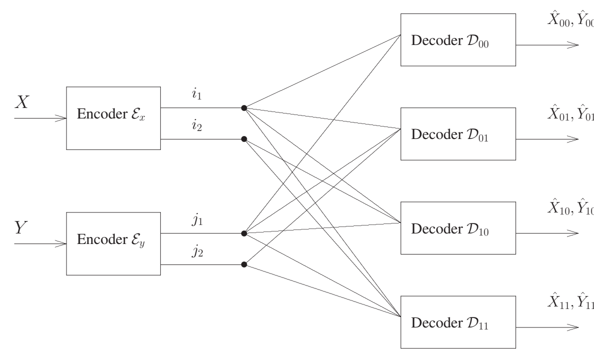


Fig. 1. Scalable distributed source coding

In our previous work on multi-stage distributed source coding (MS-DSC) [6], which is a special constrained case of the S-DSC problem, we showed that there exists a fundamental conflict between the objectives of multi-stage coding and distributed quantization. While there is no such direct conflict in unconstrained S-DSC, and a Lloyd-style DSC design algorithm [4, 5] can be extended to S-DSC, the resulting algorithm depends heavily on initialization. This paper considers efficient solutions and proposes MS-DSC design as effective initialization for S-DSC.

It is also desirable that the S-DSC system be *robust* to fades or failures of various communication channels, and utilize all received information to attain maximal efficiency. We incorporate system robustness objectives by adopting techniques for robust distributed coder design [5].

The rest of the paper is organized as follows. In Sec. 2, we state the problem formally and introduce notation. In Sec. 3, we specify the components of the S-DSC system and an iterative design algorithm. Sec. 4 reviews the MS-DSC system while Sec. 5 describes the clever initialization scheme for S-DSC. Simulation results are summarized in Sec. 6, followed by conclusions in Sec. 7.

2. PROBLEM STATEMENT

Consider the S-DSC scenario in Fig. 1. For brevity, we restrict the analysis to the case of two sources and to two layers, but without

978-1-4244-2354-5/09/$25.00 ©2009 IEEE

loss of generality as the model is trivially extendible to an arbitrary number of sources or layers. Here (X, Y) are two continuous amplitude, i.i.d., correlated (scalar or vector) sources. The encoder $\mathcal{E}_x$ for source X compresses the data and transmits an index pair $\{i_1, i_2\}$ where $i_1 \in \{1..2^{R_{1x}}\}$ and $i_2 \in \{1..2^{R_{2x}}\}$. Similarly the encoder $\mathcal{E}_y$ for Y has an index pair $\{j_1, j_2\}$ as output where $j_1 \in \{1..2^{R_{1y}}\}$ and $j_2 \in \{1..2^{R_{2y}}\}$. R_{1x} and R_{1y} correspond to the first (base) layer rates while R_{2x} and R_{2y} denote the incremental second (enhancement) layer rates.

We assume that the fusion center obtains full information from the base layer while data from the enhancement layers for sources X and Y is lost independently with probabilities p_x and p_y, respectively. Depending on the subset of information received from the enhancement layers, the fusion center uses decoder $\mathcal{D}_{00}$, $\mathcal{D}_{01}$, $\mathcal{D}_{10}$ or $\mathcal{D}_{11}$ to reconstruct X as $\hat{X}_{00}$, $\hat{X}_{01}$, $\hat{X}_{10}$ or $\hat{X}_{11}$, respectively and similarly for Y (see Fig. 1). The subscripts in a decoder indicate whether the enhancement layer for source X and Y have been received, e.g., decoder $\mathcal{D}_{10}$ is used when enhancement layer from only X is received. Thus the decoders $\mathcal{D}_{00}$, $\mathcal{D}_{01}$, $\mathcal{D}_{10}$ and $\mathcal{D}_{11}$ are used with probabilities $p_{00} = p_x p_y$, $p_{01} = p_x(1-p_y)$, $p_{10} = (1-p_x)p_y$ and $p_{11} = (1-p_x)(1-p_y)$, respectively.

The distortion incurred when decoder $\mathcal{D}_{00}$ is used for reconstructing sources will be:

$$E[\alpha d(X, \hat{X}_{00}) + (1-\alpha) d(Y, \hat{Y}_{00})], \tag{1}$$

where $d(\cdot, \cdot)$ is an appropriately defined distortion measure and $\alpha \in [0, 1]$ is a weighting factor which governs the relative importance of the sources X and Y at the fusion center. The other distortion terms when decoders $\mathcal{D}_{01}$, $\mathcal{D}_{10}$ or $\mathcal{D}_{11}$ are used are similarly defined. Note that we use uppercase letters for a random variable and lowercase letters to denote its particular realization.

We use the following streamlined notation to denote the distortion terms for a data point (x,y) when different decoders are used:

$$\begin{aligned}
D_{00}(x, y) &= \alpha d(x, \hat{x}_{00}(i_1, j_1)) + (1-\alpha) d(y, \hat{y}_{00}(i_1, j_1)) \\
D_{01}(x, y) &= \alpha d(x, \hat{x}_{01}(i_1, j_1, j_2)) + (1-\alpha) d(y, \hat{y}_{01}(i_1, j_1, j_2)) \\
D_{10}(x, y) &= \alpha d(x, \hat{x}_{10}(i_1, j_1, i_2)) + (1-\alpha) d(y, \hat{y}_{10}(i_1, j_1, i_2)) \\
D_{11}(x, y) &= \alpha d(x, \hat{x}_{11}(i_1, j_1, i_2, j_2)) + \\
&\quad (1-\alpha) d(y, \hat{y}_{11}(i_1, j_1, i_2, j_2)),
\end{aligned} \tag{2}$$

where the index pairs are determined by the source values, $\mathcal{E}_x(x) = \{i_1, i_2\}$ and $\mathcal{E}_y(y) = \{j_1, j_2\}$. D_{00} denotes to the distortion when decoder $\mathcal{D}_{00}$ is used and so on.

Next we define the net average distortion incurred for a source pair {x,y} as:

$$\begin{aligned}
D_{net}(x, y) &= p_{00} D_{00}(x, y) + p_{01} D_{01}(x, y) \\
&+ p_{10} D_{10}(x, y) + p_{11} D_{11}(x, y).
\end{aligned} \tag{3}$$

The S-DSC design objective is to minimize the following distortion cost given rate allocations R_{1x}, R_{2x}, R_{1y} and R_{2y}; and enhancement layer loss probabilities p_x and p_y:

$$E[D_{net}(X, Y)]. \tag{4}$$

3. SCALABLE-DSC MODULES

3.1. Encoder

The S-DSC encoder $\mathcal{E}_x$ for source X is shown in Fig. 2. The high rate quantizer maps source X to an index k representing Voronoi region C^x_k. Next, a *lossy* (many to one) mapping which we refer to as Wyner-Ziv (WZ) mapping is employed (the name loosely

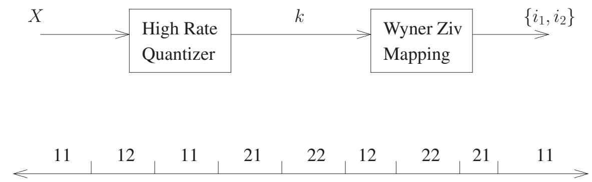


Fig. 2. S-DSC encoder for source X and an example of Wyner Ziv mapping from Voronoi regions to index pair $\{i_1, i_2\}$. Here the first region is mapped to index pair $\{i_1, i_2\} = \{1, 1\}$ etc.

accounts for the fact that the scenario involves lossy coding with side information whose asymptotic performance bound was given in [10]). The WZ mapping block takes in k and outputs an index pair $\{i_1, i_2\} = v(k)$ representing region $R^x_{i_1,i_2} = \bigcup_{k:v(k)=\{i_1,i_2\}} C^x_k$, to be transmitted over the channel. The idea is that using the information from correlated source Y, different regions C^x_k which are mapped to the same WZ index pair can be distinguished. An example of WZ mapping for a scalar source with $\mathcal{K} = 9$, $\mathcal{I}_1 = 2$ and $\mathcal{I}_2 = 2$, is also given in Fig. 2. The encoder $\mathcal{E}_{1y}$ for source Y is defined similarly.

3.2. Decoders

Each of the decoders $\mathcal{D}_{00}$, $\mathcal{D}_{01}$, $\mathcal{D}_{10}$ and $\mathcal{D}_{11}$ employs two codebooks (one per source). As mentioned in Sec. 2, the input indices for $\mathcal{D}_{00}$ are i_1 and j_1; $\mathcal{D}_{01}$ are i_1, j_1 and j_2, etc., to account for various channel conditions.

3.3. Components to Optimize

The design algorithm for S-DSC needs to optimize the high rate quantizers, WZ mappings (or encoder) and decoder codebooks. We restrict the scope here to the design of decoder codebooks and WZ mappings. (For simplicity, we assume that high rate quantizers are independently designed using Lloyd's algorithm [2]. Additional gains due to their joint optimization with the rest of the system are expected to be small.)

3.4. Iterative Design Algorithm

Herein we assume squared error distortion measure for simplicity. To minimize the cost in (4), the Wyner-Ziv mappings and the various decoder codebooks (with some initialization) are optimized iteratively using the following update rules obtained from the necessary conditions for optimality:

1. **Wyner Ziv Mappings** For $k = 1 : \mathcal{K}$, assign k to index pair $\{i_1, i_2\}$ according to

   $$v(k) = \arg\min_{i_1, i_2} \sum_{x \in C_k} D_{net}(x, y). \tag{5}$$

   The dependence of $D_{net}(x, y)$ on the index pair is specified by (2) and (3).

2. **Decoder Codebook: Reconstruction Values**

   $$\begin{aligned}
   \hat{x}_{00}(i_1, j_1) &= E[X | X \in R^x_{i_1}, Y \in R^y_{j_1}] \\
   \hat{x}_{01}(i_1, j_1, j_2) &= E[X | X \in R^x_{i_1}, Y \in R^y_{j_1,j_2}] \\
   \hat{x}_{10}(i_1, j_1, i_2) &= E[X | X \in R^x_{i_1,i_2}, Y \in R^y_{j_1}] \\
   \hat{x}_{11}(i_1, j_1, i_2, j_2) &= E[X | X \in R^x_{i_1,i_2}, Y \in R^y_{j_1,j_2}]
   \end{aligned} \tag{6}$$

Similar rules for source Y can be trivially obtained and are omitted here. In the above update rules, $R^x_{i_1} = \bigcup_{i_2} R^x_{i_1,i_2}$ and $R^y_{j_1} = \bigcup_{j_2} R^y_{j_1,j_2}$.

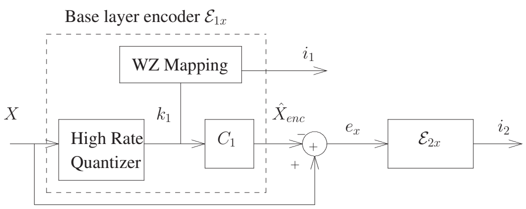


Fig. 3. MS-DSC encoder

The index-assignment problem implicit in WZ mapping is a discrete optimization problem. In the case of S-DSC, we need to map K different regions to one of the $I_1 * I_2$ indices for source X. Random initialization of the above iterative algorithm generally leads to a poor local minimum. While we do not attempt a global solution to this index assignment problem, we propose to use an optimized MS-DSC system as a "clever" initialization for S-DSC design. Simulation results confirm that the proposed initialization obtains considerable gains over uninformed, randomly initialized solutions.

4. MULTI-STAGE DSC: OVERVIEW

4.1. Encoder

The MS-DSC encoder for source X comprises of base layer encoder $\mathcal{E}_{1x}$ and enhancement layer encoder $\mathcal{E}_{2x}$ (Fig. 3). The WZ mapping block for $\mathcal{E}_{1x}$ takes k_1 as input and outputs index $i_1 = v_1(k_1)$ representing region $R^x_{i_1} = \bigcup_{k_1:v_1(k_1)=i_1} C^x_{k_1}$. The *encoder* codebook C_1 also takes k_1 as input and outputs $\hat{X}_{enc}$ which is used to compute the residual $e_x = X - \hat{X}_{enc}$. Base layer encoder $\mathcal{E}_{1y}$ for source Y is defined similarly. The error residuals e_x and e_y obtained by the first encoding stage are correlated and a second layer distributed coder is designed to exploit inter-source correlations between the residuals. The second stage encoders $\mathcal{E}_{2x}$ and $\mathcal{E}_{2y}$ similarly consist of a high rate quantizer followed by WZ mapping.

4.2. Decoder

Similar to the S-DSC system, the MS-DSC system consists of four decoders: $\mathcal{D}_{00}$, $\mathcal{D}_{01}$, $\mathcal{D}_{10}$ and $\mathcal{D}_{11}$. However, in MS-DSC the decoding is performed in an additive fashion. For example, for source X, decoder $\mathcal{D}_{10}$ in MS-DSC (Fig. 4) comprises two codebooks which have the base layer and enhancement layer indices as input respectively.

4.3. MS-DSC Design

The most crucial observation is of the underlying objectives conflict in MS-DSC design and that one should allow different codebooks for constructing $\hat{X}_{enc}$ and $\hat{X}_{10,dec}$, etc. At the decoder, both indices i_1 and j_1 can be utilized to construct $\hat{X}_{10,dec}$. However, the encoder for source X only has access to index i_1 to construct $\hat{X}_{enc}$ and does not know j_1. Obviously, there will be a mismatch between $\hat{X}_{enc}$ and $\hat{X}_{10,dec}$. A possible way to match $\hat{X}_{10,dec}$ with $\hat{X}_{enc}$ is to make $\hat{X}_{10,dec}$ a function of i_1 alone. Such a strategy is referred to as the naive scheme in MS-DSC and is analyzed in [6]. Unfortunately, it defeats the main objective of efficiently exploiting inter-source correlation in the base layer of MS-DSC.

The idea, is therefore to allow for some mismatch between the first layer estimates in MS-DSC at the encoder and decoder, and jointly optimize so that efficient distributed coding will more than compensate for any allowed mismatch. Another crucial point to note

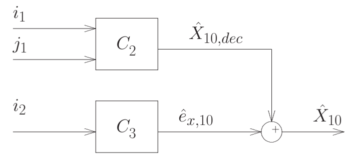


Fig. 4. MS-DSC decoder $\mathcal{D}_{10}$

is that the source encoder has complete knowledge of the source itself, while the decoder has additional knowledge from the correlated source Y, in the form of index j_1 . This implies that there may exist some (elusive) overlapping information at both ends that could be exploited, if an appropriate means were devised.

Therefore different codebooks are used for calculating $\hat{X}_{10,dec}$ and $\hat{X}_{enc}$ at the decoder versus encoder in the MS-DSC system. The encoder codebook (C_1) has k_1 as input, and the *decoder helper* codebook C_2 has inputs i_1 and j_1. This flexibility enables optimization of the tradeoff between better exploitation of inter-source correlations, and the cost of some mismatch in the system. Appropriate design of encoder and decoder codebooks (as well as WZ mappings) optimizes the precise overall performance while accounting for the mismatch. More details about MS-DSC coder design are given in [6].

5. EFFECTIVE INITIALIZATION FOR S-DSC DESIGN

MS-DSC is a special case of S-DSC under additive encoding/decoding constraints. The proposed scheme for S-DSC takes the optimal MS-DSC system as an effective initialization and then removes the structural constraints to apply the iterative algorithm in Sec. 3.4.

In the MS-DSC scheme, the source space for X (in base layer) is divided into K_1 different regions. These K_1 different regions are mapped to one of the I_1 regions via the base-layer WZ mapping. The residual $e_x = X - \hat{X}_{enc}$ is then quantized by a high-rate quantizer having K_2 different output cells (regions), which are mapped to one of the I_2 different regions via enhancement layer WZ mapping. Hence during design, all the training point samples for source X are associated with an index $\{i_1, i_2\}$.

Now consider a sample X corresponding to some high-rate quantizer region $k_1 \in \{1..K_1\}$ and WZ mapping region i_1. This sample is associated with an index i_2 for the enhancement layer (through e_x). We define C^x_k ($k \in \{1..K\}$ and $K = K_1 * I_2$) as the set of all source points X that lie in the region $C^x_{k_1}$ (corresponding to high rate quantizer output index k_1) and $R^x_{i_2}$ (corresponding to index i_2 of the enhancement layer WZ mapping), i.e.,

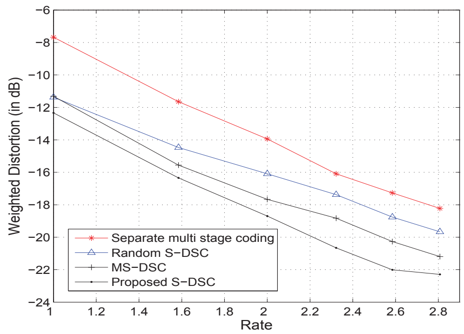
$$C^x_k = C^x_{k_1} \bigcap R^x_{i_2}. \quad (7)$$

Each of the regions C^x_k is associated with an index pair $\{i_1, i_2\}$. So we effectively view the X source space as divided into K different regions, each of which is mapped to one of the index pair $\{i_1, i_2\}$ via an *implicit S-DSC* WZ mapping $v(k) = \{i_1, i_2\}$. We can use these K regions and WZ mappings as an initial solution for the S-DSC algorithm in Sec. 3.4. A similar construction of the different regions and WZ mapping is performed for source Y.

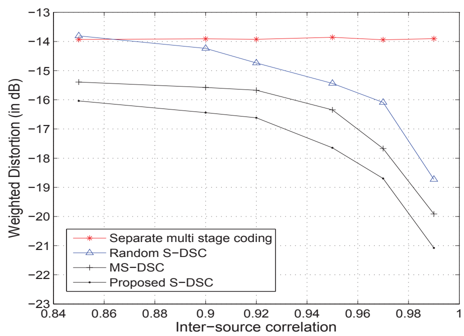
5.1. Encoding during S-DSC operation

Note that the region $R^x_{i_2}$ in (7) corresponding to index i_2 (outcome of enhancement layer WZ mapping) is a union of different possibly, non-contiguous regions. Hence the region C^x_k in (7) is also a union of different possibly, non-contiguous regions.

The encoding for a sample X will still be performed in a similar fashion to encoding in MS-DSC, i.e., for X, find the high rate



(a)



(b)

Fig. 5. Performance comparison of randomly initialized scalable DSC, multi-stage DSC, and proposed scalable DSC which is initialized with MS-DSC. (a) Performance versus source rate. (b) Performance versus inter-source correlation.

quantizer region and the corresponding index k_1 and $\hat{X}_{enc}$. Using k_1, find the base layer WZ mapping index i_1. Calculate the residual $e_x = X - \hat{X}_{enc}$ and find the corresponding enhancement layer WZ index i_2. The resulting indices i_1 and i_2 are then transmitted as base and enhancement layer information, respectively.

6. SIMULATION RESULTS

We give two examples to demonstrate the gains of S-DSC with effective initialization as proposed, over randomly initialized S-DSC. In the simulations, sources X and Y are jointly Gaussian with zero means, unit variances and correlation coefficient ρ. The weighting coefficient α is set to 0.5 to give equal importance to both the sources at the decoder. A training set of 10000 scalars is generated. The number of prototypes is 60 for the high rate quantizers which are designed using Lloyd's algorithm [2]. Simulation results are depicted in Fig. 5.

In the first experiment, the same transmission rate is allocated to each layer of each source, i.e., $R_{1x} = R_{2x} = R_{1y} = R_{2y} = R$. The probability of enhancement layer loss for both sources $p_x = p_y = 0.2$ and $\rho = 0.97$. The weighted distortion (4) at the decoder is plotted versus the rate R. We compare: (i) separate multi-stage coding (in which inter-source correlation is not utilized) (ii) randomly initialized S-DSC system ('Random S-DSC') (iii) structurally constrained MS-DSC system ('MS-DSC'), and (iv) proposed S-DSC system which is initialized by MS-DSC ('Proposed S-DSC'). For fair comparison and to eliminate atypically poor results, the initialization for both the Random S-DSC and MS-DSC algorithms was done 20 times and the best results are reported. The proposed S-DSC scheme leads to substantial gains over Randomly initialized S-DSC, for example $\sim$ 3.3 dB at $R = log_2(5)$ bits/sample. It is noteworthy that the performance of Random S-DSC is almost always worse than MS-DSC despite MS-DSC's limiting structural constraint. This illustrates the severity of the local minima problem and the critical importance of effective initialization in S-DSC design.

In the second simulation, we vary the inter-source correlation ρ while keeping all the rates fixed at 2 bits/sample. Again the proposed S-DSC scheme outperforms Random S-DSC and gains of up to $\sim$ 2.6 dB are achieved (e.g., at $\rho = 0.97$). We should also mention here that the proposed S-DSC scheme takes about 4-5 times longer running time than the randomly initialized S-DSC during design.

7. CONCLUSIONS

In this paper, we have considered the design of scalable distributed source coders. The proposed S-DSC system is robust to failure of a subset of the channels and utilizes all the available information to attain the best possible compression efficiency. The performance of a Lloyd-style S-DSC algorithm is heavily dependent on initialization and may even underperform a proposed structurally constrained (multi-stage) DSC algorithm. The MS-DSC algorithm solution is used as a "clever" initialization for the S-DSC design algorithm. Simulation results show that the resulting S-DSC scheme consistently outperforms the randomly initialized S-DSC approach. Finally, we note that the proposed methods are extendible to incorporate entropy coding, but such extension is omitted for brevity.

8. REFERENCES

[1] W. Equitz and T. M. Cover, "Successive refinement of information ," *IEEE Trans. on Information Theory*, vol. 37, no. 2, pp. 269–275, Nov 1991.

[2] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. on Information Theory*, vol. 28, no. 2, pp. 129–137, Mar 1982.

[3] S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense microsensor network," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 51–60, Mar 2002.

[4] D. Rebollo-Monedero, R. Zhang, and B. Girod, "Design of optimal quantizers for distributed source coding," in *IEEE Data Compression Conference*, Mar 2003, pp. 13–22.

[5] A. Saxena, J. Nayak, and K. Rose, "On efficient quantizer design for robust distributed source coding," in *IEEE Data Compression Conference*, Mar 2006, pp. 63–72.

[6] A. Saxena and K. Rose, "Distributed multi-stage coding of correlated sources," in *IEEE Data Compression Conference*, Mar 2008, pp. 312–321.

[7] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. on Information Theory*, vol. 19, no. 4, pp. 471–480, Jul 1973.

[8] Y. Steinberg and N. Merhav, "On successive refinement for the Wyner-Ziv problem," *IEEE Trans. on Information Theory*, vol. 50, no. 8, pp. 1636–1654, Aug 2004.

[9] C. Tian and S. Diggavi, "Multistage successive refinement for Wyner-Ziv source coding with degraded side informations," in *IEEE ISIT*, July 2006, pp. 1594–1598.

[10] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side-information at the decoder," *IEEE Trans. on Information Theory*, vol. 22, pp. 1–10, Jan 1976.

[11] Z. Xiong, A. Liveris, and S.Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 80–94, Sep 2004.