Data Compression and Transmission in Wireless Sensor Networks

Ashkan Heshmati

A Thesis
in
The Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science (Electrical Engineering) at
Concordia University
Montreal, Quebec, Canada

March 2007

Your file Votre référence
ISBN: 978-0-494-28915-0
Our file Notre référence
ISBN: 978-0-494-28915-0

ABSTRACT

Data Compression and Transmission in Wireless Sensor Networks

Ashkan Heshmati

Wireless Sensor Networks (WSNs) have emerged as a new research area in communication systems in recent years. Theoretically, sensor networks can be modeled by either Chief Executive Officer (CEO) problem or Multi-terminal communication problem. A Wireless Sensor Network consists of a set of battery-powered nodes which are limited in energy; and network's lifetime depends on the energy consumption of the nodes. Therefore, one of the main issues in designing WSNs is reducing the energy consumption. In sensor networks we deal with a set of correlated observations. We can exploit this correlation and compress the data which is going to be transmitted from different nodes. Moreover, as all nodes are equipped with antennas, we can take advantage of having several antennas and apply advanced energy-efficient communication methods such as Multiple-Input Multiple-Output (MIMO) technique.

The contributions of this thesis are presented in two parts: In the first part we consider the Chief Executive Officer (CEO) problem for binary sources. We model the source by an i.i.d. unbiased sequence which is connected to each sensor via a Binary Symmetric Channel (BSC). We analyze the behavior of the system and show that sensors can compress their observations according to Slepian-Wolf rate bound. We consider the conditional entropy of the source given a set of observations as a function of number of sensors as well as cross-over probability of the BSC. We derive a closed form for this function and prove that it converges to zero as the number of sensors tends to infinity. Substituting this function in *Fano's inequality*, we determine the minimum number of

sensors required to achieve a desired probability of error. We also derive the estimation rule for a Maximum Likelihood estimator.

In the second part we consider cooperative communication where sensors are able to communicate in order to jointly compress their correlated information and apply MIMO transmission techniques. We assume two scenarios. In the first scenario we consider the Gaussian CEO problem where sensors observe a common Gaussian source and report noisy versions of this source to the CEO. We propose an energy-efficient cooperative algorithm for data estimation exploiting virtual MIMO technique. In the second scenario we extend the problem to Multi-terminal communication where all nodes wish to transfer their individual correlated information to the Fusion Center which is interested in all observations. We propose two different cooperative data compression algorithms. In the first algorithm we transform the correlated data into parallel independent Gaussian sources. We derive mathematical closed form equations for the transform matrices and optimum rate allocation. In the second algorithm we apply Vector Quantization. We simulate the proposed algorithms and compare their performance.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my graduate supervisor, Professor M. Reza Soleymani, for his guidance and support during the course of my graduate studies. I appreciate the considerable amount of time and effort he invested in helping me with my research and thesis. I would also thank my fellow colleagues who have provided helpful suggestions along the way. This thesis could not have been completed without the endless encouragement and love from my family, I am deeply indebted to them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

θ	Gaussian source to be estimated
n_i	Gaussian additive noise
X_i	Sensor i's observation
$X_{i,j}$	Observation of sensor i belonging to cluster j
ε_j	Differential data of sensor j in each cluster
m	Number of members in each cluster
L	Number of sensors
l	Length of binary message
y_j	Sub-average of members in each cluster
R	Compression (Quantization) rate
Y	Binary source to be estimated
E_i	Binary additive error
p_e	Cross-over probability in Binary Symmetric Channel
p_r	Desired average probability of bit error
$H(X)$	Entropy of X
σ_x^2	Variance of Gaussian source to be estimated
σ_n^2	Variance of Gaussian noise
σ_{ij}	Covariance of two random variables X_i and X_j
C_{XX}	Covariance Matrix
U	Transform Matrix for eigenvalue decomposition
V_i	Eigenvectors
U_i	Normalized eigenvectors
Λ	Diagonal covariance matrix
λ_i	Eigenvalues
Λ_L	Likelihood Ratio

D	Distortion
Q_i	Vector Quantization point
$f_X(x)$	Probability distortion function of X
N_T	Number of transmit antennas
N_R	Number of receive antennas
P_C	Circuit power consumption
P_{ADC}	Analog to Digital Converter power consumption
P_{DAC}	Digital to Analog Converter power consumption
P_{mix}	Mixer power consumption
P_{synth}	Frequency Synthesizer power consumption
P_{ct}	Transmitter circuit power consumption
P_{cr}	Receiver circuit power consumption
P_{IFA}	Intermediate Frequency Amplifier power consumption
P_{LNA}	Low Noise Amplifier power consumption
P_{filt}	Power consumption of active filters of Transmitter
P_{filr}	Power consumption of active filters of Receiver
P_{PA}	Power consumption of power amplifier
P_{out}	Transmission power consumption
$\overline{P}_b$	Average probability of bit error
f_c	Center frequency
T_{on}	Transmission time for each symbol
M_l	Link margin
N_f	Noise figure
G_t	Transmitter antenna gain
G_r	Receiver antenna gain
λ	Wavelength
$\overline{E}_b$	Average energy per bit
R_b	Bit rate
d	Distance
κ	Exponent of path loss

N_r	Power spectral density of total noise at the receiver
N_0	Power spectral density of noise at room temperature
ξ	Peak-to-average ratio
η	Drain efficiency
WSN	Wireless Sensor Network
LDPC	Low-Density Parity Check
CEO	Chief Executive Officer
BER	Bit Error Rate
MIMO	Multiple-Input Multiple-Output
SISO	Single-Input Single-Output

CHAPTER ONE

INTRODUCTION

1.1 Motivation

Sensor networks have become known as a new discipline in communication systems in recent years. In many engineering problems we deal with sensing an environment and making estimates based on the phenomena sensed. The numerous applications of sensors in life and environment together with advances in wireless communications, microelectronics and digital signal processing have motivated researchers to develop the idea of Wireless Sensor Networks (WSNs). As a general definition, a Wireless Sensor Network consists of a set of small, battery-powered and low-cost nodes which are usually equipped with three major components: a sensing device, a processor and a communication module. They can communicate with each other, process their observations and cooperatively transmit their information to the Fusion Center (FC) [1].

Wireless sensor networks have many applications in imaging the environment and habitat monitoring. Imaging implies spatially distributed sampling or measuring of a field (e.g., moisture content in soil or air temperature) which provides us with more precision, diversity and tolerance against failure. The fact that nodes are connected wirelessly and operate on batteries let us sense remote or inaccessible areas. For instance they can be used for detection of fire in forests, floods and earthquakes. They also have health, industrial, commercial and home applications which include monitoring the patients or industrial settings and home automation systems [2].

Sensor networks have wide variety of applications. Depending on the application, design constraints vary from network to network. We categorize sensor networks into different classes. For instance in one class of WSNs, all nodes observe one phenomenon and the Fusion Center estimates the phenomenon based on all observations. In this system exact observations of individual nodes are not of interest and only the final estimation is desired (e.g. measuring the moisture content in soil). On the other hand, there also exist other networks which require all individual observations at the Center (e.g. fire detection networks). We consider both system models in our work and throughout the thesis we study Energy-Constrained wireless sensor networks.

One of the main characteristics of sensor networks is that in most cases we deal with an ensemble of correlated observations. Specially, in distributed sensing systems where all agents are imaging one phenomenon, observations are highly correlated. The problem of distributed sensing and encoding correlated observations was first introduced by Flynn and Gray [3] in 1987. They considered a distributed sensing system wherein several sensors observe a phenomenon, separately encode their observations and communicate to an estimator over a channel of limited capacity. The objective is to estimate the phenomenon with minimum distortion. They analyzed the achievable rate and distortion for the case of two sensors under certain conditions and proposed two algorithms for quantizer design in order to achieve good performance.

Distributed sensing and wireless sensor networks can also be regarded as an example of the classical Chief Executive Officer (CEO) problem [4], defined in information theory. The problem states that the CEO (also referred to as Central Estimation Officer) is interested in estimating a data sequence which cannot be observed by him directly. Therefore, CEO deploys a team of agents who observe independently corrupted versions of this data and report it to him. This problem was introduced by Berger and Zhang [5] in 1994 and was developed in [6] with a Gaussian source model. They considered a continuous source with an i.i.d. sequence of zero mean Gaussian random variables $N(0,\sigma_x^2)$ which are corrupted by identical independent memoryless noises $N(0,\sigma_n^2)$. This model is called "Quadratic Gaussian CEO Problem". In [7] the rate-distortion function for this source is studied. We also use this data model for our 'Continuous' sources throughout this thesis. In addition, we extend the CEO problem to discrete model and consider 'Binary' correlated sources and analyze the binary source modeled in [10].

The problem of lossless encoding of discrete correlated sources was introduced by Slepian and Wolf [8] in 1973. Consider correlated discrete sources X and Y with joint entropy $H(X,Y)$. The two sources are encoded separately and decoded jointly. Slepian and Wolf proved that if X is encoded to rate $H(X)$, Y could be encoded to rate $H(Y \mid X)$ and decoded using X, with no loss of information. The joint decoder exploits X and the side information received from Y in order to decode Y. They presented a theoretical limit for achievable rate region. After them, Wyner and Ziv [9] considered the extension of this problem for lossy compression.

In recent years, several algorithms have been proposed for encoding correlated binary sources using error correcting codes. Proposed algorithms are mostly based on Turbo Codes [10]–[14] or Low-Density Parity Check (LDPC) codes [15]–[17], [22]. The results have improved as sequentially presented and in comparison, it seems that LDPC codes have better performance than Turbo codes. They have achieved rates very close to the Slepian–Wolf limit. Distributed Source Coding (or source coding with side information) has application in systems where sources do not communicate with each other and want to send their information to a common decoder. Therefore, Distributed Source Coding is a good candidate for Wireless Sensor Networks [19]–[22].

A group of the Wireless Sensor Networks, which is of our interest, can be regarded as an implementation of CEO problem with energy constraint for the nodes. Considering the energy constraint together with correlation of data, we can compress the information and reduce the number of bits to be transmitted. As in wireless communications most of the energy consumption is due to transmission; reducing the number of bits to be transmitted helps us save considerable amount of energy. On the other hand, it is also possible to apply advanced wireless transmission techniques (such as multiple antenna communication technique) which requires less energy compared to traditional systems.

A lot of research has been done in order to take advantage of the correlation among sensors' data and reduce the number of bits to be transmitted. Some of the works are based on distributed source coding theorem [19]–[22] and some are based on decentralized estimation technique [23]–[27]. In [19], authors proposed DIstributed Source Coding Using Syndromes (DISCUS) technique for sensor networks. In DISCUS [18], codewords are divided into Cosets. One source sends uncoded data and other sources send the side information which is the index of the Coset containing their codeword. Extension of this idea with an adaptive signal processing approach is proposed in [20]. In this work, in addition to the concept of DISCUS, an adaptive filtering framework is used at the Data Gathering Center to continuously track the correlation structure among sensors' data. The data gathering center first asks nodes for uncoded data for K rounds in order to learn the correlation structure of data and estimate the correlation parameters. After initialization, it asks for coded (compressed) data and using the already set correlation parameters estimates data of each sensor and updates the correlation parameters for the next round. They showed that their algorithm saves considerable energy (from 10%–65%). Other Distributed Source Coding techniques have also been proposed for wireless sensor networks. In [21] compression via powerful channel codes (such as Turbo and LDPC codes) for sensor networks is presented. The LDPC-based coding scheme presented in [22] achieves any arbitrary rate on the Slepian–Wolf region.

The problem of decentralized estimation in sensor networks has been studied under different constraints in [23]–[25]. In these algorithms, sensors perform a local quantization on their data considering the fact that their observations are correlated with those of other sensors. They produce a binary message and send it to the FC. Fusion Center combines these messages based on the quantization rules used at the sensor nodes and estimates the unknown parameter. Optimal local quantization and final fusion rules are investigated in these works. Decentralized Estimation problem is extended in [26] and [27], considering energy constraint and power scheduling for the nodes. In these papers, algorithms are designed assuming that sensor observations and additive noises are bounded. In our thesis, we always consider the unbounded noise case and Gaussian probability density function for sensor observations and noises.

As an alternative approach, some research has been done on energy-efficient transmission techniques such as cooperative/virtual Multiple-Input Multiple-Output (MIMO) and its application in sensor networks [28]–[33]. In these works, as each sensor is equipped with one antenna, nodes are able to form a virtual MIMO system. If nodes communicate with one another and get knowledge of the information of each other, they can be regarded as one system and resemble a virtual MIMO. In [28] the application of MIMO techniques in sensor networks based on Alamouti [34] space-time block codes was introduced. In this paper, authors compared the energy-efficiency of MIMO systems with SISO and showed that in some cases, where transmission distance is short, SISO transmission can be more energy-efficient than cooperative MIMO. Note that in sensor networks total energy consumption of nodes should be considered. As cooperative MIMO has the overhead of communication among nodes, considering both the Circuit and Transmission energy consumptions makes SISO outperform cooperative MIMO in some cases. The problem of cooperative transmission in Wireless Sensor Networks is extended and analyzed in more details in [30]–[33]. Energy-efficiency of MIMO techniques has been explored analytically in [30] and [31]. Also a combination of the distributed and adaptive signal processing algorithm for source coding presented in [20] and cooperative MIMO technique is proposed in [32].

1.2 Contributions

Our contributions in this thesis are divided in two parts. In the first part, we extend the CEO problem to Correlated Binary Sources. We define a correlation model for a set of binary observers (sensors) which together with the CEO want to estimate a common source. We analyze the estimation problem based on entropy. We calculate the conditional entropy of the unknown parameter given the set of observations, prove its convergence as the number of observers tends to infinity and propose a mathematical method to find the minimum required number of sensors for estimation with given probability of error. We also derive the estimation rule according to Maximum Likelihood estimation method. This problem can be regarded as a special case of sensor networks, when observations are assumed to be 'Binary'.

In the second part, we focus on energy-constrained Wireless Sensor Networks and Correlated Gaussian Sources. First, we propose a data estimation algorithm for Quadratic Gaussian CEO Problem, which exploits Cooperative MIMO for transmission. In our scenario, nodes communicate with each other to process their data in order to make it more compact and become able to transmit it using MIMO techniques. We then consider the Multi-terminal communication problem for sensor networks where the center is interested in receiving all individual observations. We propose and compare two cooperative algorithms for data compression and transmission.

1.3 Organization of Chapters

The thesis is organized as follows: in Chapter 2 we present a background on the concepts and terms which are defined in the literature and used in our thesis. We present a review on the classical CEO problem and its relations to Wireless Sensor Networks. It is followed by Distributed Source Coding and its applications in WSNs. Moreover, we review the Cooperative Source Coding for correlated information and finally present the background on Cooperative Transmission in WSNs and Space-Time Coding.

In Chapter 3 we consider the problem of estimation in correlated binary sources. A number of agents are observing a phenomenon and wish to report their observations to a common center (CEO). We define a discrete model for data and their correlation and prove that it is possible to detect the phenomenon with any arbitrarily low distortion (probability of error).

In Chapter 4 we present the energy-efficiency of Multiple-Input Multiple-Output (MIMO) systems and its application on Wireless Sensor Networks. Firstly, we consider the Gaussian CEO problem in Wireless Sensor Networks where all nodes are deployed in a field in order to observe a phenomenon and report it to the CEO. In this scenario, CEO or Fusion Center is interested in estimating the information about the environment by processing observations of all nodes. We propose an energy-efficient cooperative algorithm for data estimation in CEO-based wireless sensor networks exploiting cooperative (virtual) MIMO technique.

Afterwards, we extend the problem to Multi-Terminal communication scenario and consider the general sensor network case where all nodes wish to transfer their individual information to a center. We propose two communication algorithms for the network and compare their performance. The proposed algorithms are based on joint compression of correlated Gaussian sources by transforming into independent parallel Gaussian sources and Vector Quantization. Finally, chapter 5 concludes the thesis.

CHAPTER TWO

BACKGROUND

This chapter presents a background on the main concepts and terms which are defined in the literature and used throughout the thesis. In Section 2.1 we present a review on the classical CEO problem and its relations to Wireless Sensor Networks. Section 2.2 is devoted to Distributed Source Coding and its application in WSNs. Slepian-Wolf Theorem is discussed in general and a simple example is given for clarification. We review the Cooperative Source Coding for correlated Gaussian information and Vector Quantization in Section 2.3. Finally, Section 2.4 presents the background on Cooperative Transmission in WSNs and Space-Time Coding.

2.1 CEO Problem and Distributed Sensing

In 1994, Berger and Zhang [5] introduced the Chief Executive Officer (CEO) problem. The problem states that a CEO (also referred to as Central Estimation Officer) is interested in estimating a sequence of data which cannot be observed by him directly. Hence, he employs a number of agents to observe independently corrupted (noisy) versions of the phenomenon and report to him for final estimation. The general model for CEO problem can be illustrated as Fig. 2.1.

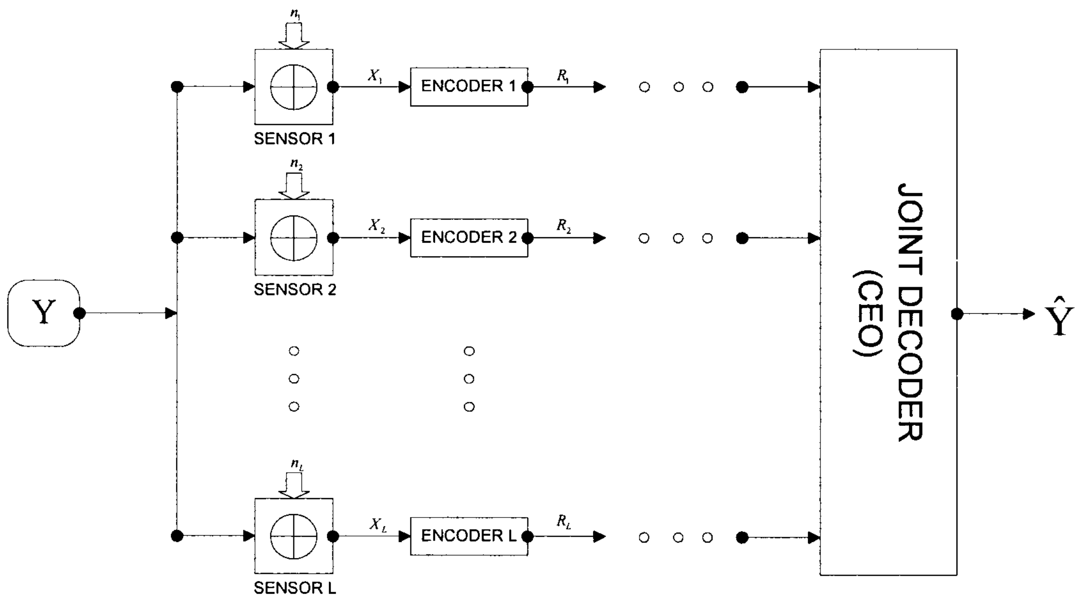

Figure 2.1 – CEO Problem

In the above figure, Y is the actual phenomenon to be sensed and estimated by the CEO and X_i's are the noisy observations of the agents. Additive noises have identical distribution but are independent from each other. That is, observations are conditionally independent given the source.

$$p(Y, X_1, X_2, \ldots X_L) = p(Y)\prod_{i=1}^{L} p(X_i \mid Y) \tag{2.1}$$

In 1995, Viswanathan and Berger [6] considered the Quadratic Gaussian model for CEO problem. They assumed that the source is a continuous Gaussian random variable $\theta \sim N(0,\sigma_x^2)$ and additive noises have also Gaussian distribution $n_i \sim N(0,\sigma_n^2)$. Hence,

$$X_i = \theta + n_i \quad , \quad X_i \sim N(0, \sigma_x^2 + \sigma_n^2) \tag{2.2}$$

The CEO is interested in reconstructing θ from the set of X_i's with minimum mean square distortion. Berger et al [4] studied the asymptotic behavior of minimum achievable distortion as the number of agents and total data rate of agents in limit tend to infinity.

In the next chapter, we study the CEO problem for a discrete binary source Y, which is observed by a number of agents who have an average probability of error (p_e) in bit detection. We assume that observations are conditionally independent given the source. In the following problem, we are interested in determining the asymptotic behavior of uncertainty about Y in limit as the number of observers tends to infinity and how many agents are required in order to detect Y with any arbitrary average number of errors.

Wireless Sensor Networks (WSNs) have recently become a growing interest. Distributed sensing of a phenomenon to gain a more precise, robust and fault tolerant observation has always been of interest for scientists. Technological advances in manufacturing small, low-cost and low-power integrated systems to perform all tasks of sensing an environment, processing the data and transmitting information have now let the engineers develop this idea [1] and implement WSNs. Wireless sensor networks can be regarded as an example of classical CEO problem. The main difference is that in CEO problem, there is no energy constraint for the agents, whereas, in a typical WSN energy consumption of nodes is the main issue. In Chapter 4, we propose the ways of reducing energy consumption by applying network source compression and cooperative transmission.

2.2 Distributed Source Coding in Sensor Networks

2.2.1 Entropy

One of the basic definitions in information theory is the concept of *Entropy*, which is a measure of uncertainty of a random variable. Suppose X is a discrete random variable with probability mass function (pmf) $p(x) = \Pr\{X = x\}, x \in \mathrm{X}$. The Entropy $H(X)$ of X is defined by [39]

$$H(X) = -\sum_{x \in \mathrm{X}} p(x) \log p(x) \tag{2.3}$$

Let $X = \begin{cases} 1 & \text{with probability, } p \\ 0 & \text{with probability, } 1-p \end{cases}$

then,

$$H(X) = -p \log p - (1-p)\log(1-p) \overset{def}{=} H(p) \tag{2.4}$$

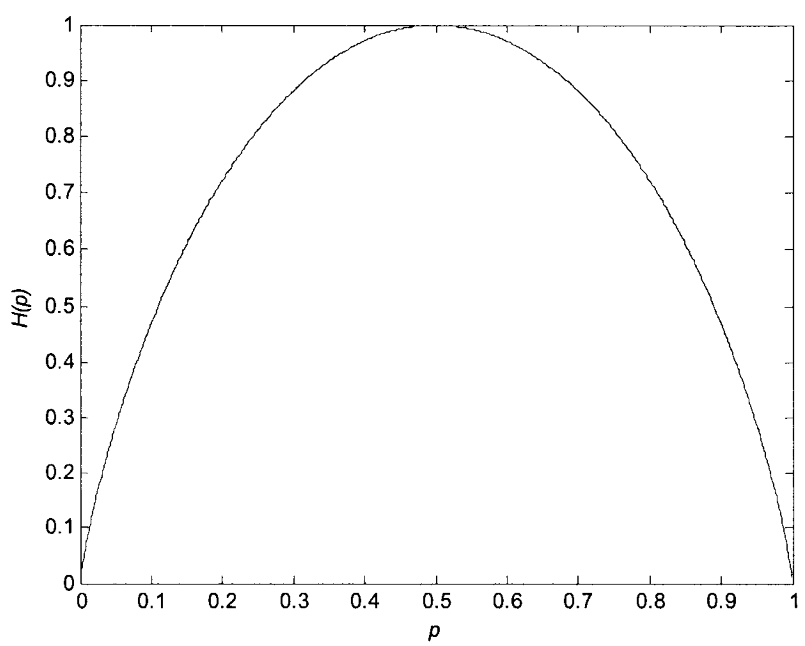

Figure 2.2 – $H(p)$ versus p

2.2.2 Slepian-Wolf Theorem

In 1973, Slepian and Wolf [8] presented a theorem that became the basis of Distributed Source Coding. They proved that correlated sources can be encoded separately (i.e. without conferring with each other) with a total rate as low as their joint entropy and decoded jointly without any loss of information. The achievable rate region is shown in Fig. 2.3.

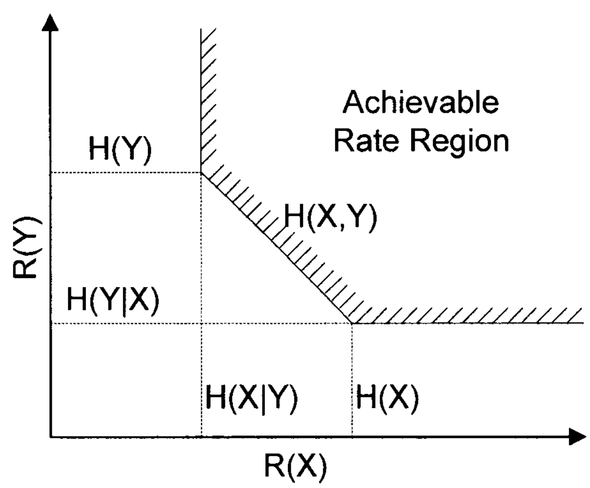

Figure 2.3 – Achievable rate region for Slepian-Wolf theorem

They proved their theorem with a Syndrome-Based idea. Consider two correlated sources X and Y with joint entropy $H(X,Y)$. The theorem states that if the first source encodes its data with rate $R_1 = H(X)$ and the second source encodes its data with rate $R_2 = H(Y \mid X)$, it is possible to decode X and Y with a joint decoder and without any loss of information. The first message sequence is transmitted with rate $R_1 = H(X)$. The second encoder is equipped with different random codebooks, in which it searches in all codebooks for a codeword identical to the message sequence to encode the second message. It then transmits the address (Syndrome) of that codebook. The decoder uses information received from Y as '*side information*' and with the help of X, decodes Y. It searches in the addressed codebook for a codeword which is maximally likely to the first message. To present this idea more clearly consider the following example:

We have two binary sources which have 16 outputs. We assume that the sources are correlated and their outputs are very close to each other. All possible codewords are listed and shown on the real axis (Fig. 2.4).

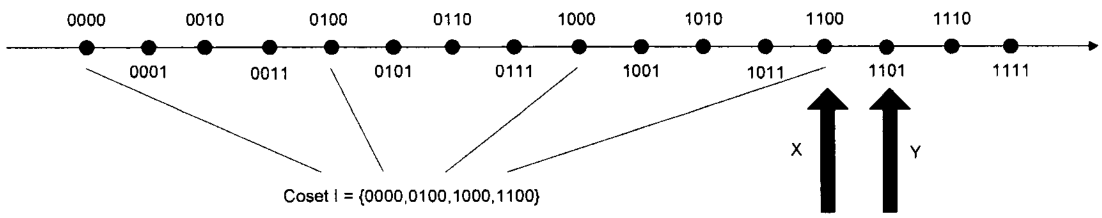

Figure 2.4 – Example of Distributed Source Coding

We divide them into Cosets (codebooks) and address each Coset with two bits

$$\text{Coset I} = \{0000, 0100, 1000, 1100\} \equiv 01$$

$$\text{Coset II} = \{0001, 0101, 1001, 1101\} \equiv 10$$

$$\text{Coset III} = \{0010, 0110, 1010, 1110\} \equiv 11$$

$$\text{Coset IV} = \{0011, 0111, 1011, 1111\} \equiv 00$$

One of the nodes (X) encodes itself into $H(X) = 4$ bits (i.e. sends its actual codeword) and the other node (Y) sends its Coset address (syndrome) as side information. The decoder looks at the members of the received coset and chooses the closest codeword to X and declares it as Y.

Ex) $X = 1100$, $Y = 1101$

X sends $M_1 = 1100$ and as Y belongs to *Coset* II , it sends $M_2 = 10$

Decoder searches in *Coset* II for the closest codeword to M_1 which is 1101

This idea has been examined in details as DIstributed Source Coding Using Syndromes (DISCUS) [18]. In this paper algebraic trellis codes are used to compress the message. Practical implementation of Slepian-Wolf model has been proposed using different types of channel codes [10]-[22]. Distributed Source Coding using Turbo codes is originally proposed in [10]. Both sources are considered binary i.i.d. with $\Pr\{X=1\}=\frac{1}{2}$. The correlation is assumed to be in the form of $\Pr\{x_i=y_i\}=1-p$. Each encoder generates two parity sequences which are punctured before transmission. Half of the systematic bits are transmitted along with parities. The decoders use the usual iterative Turbo decoding to decode X and Y. The only difference is that decoders iterate an additional extrinsic information between each other after performing each round of iteration.

Compression of binary correlated sources using LDPC codes is first considered in [15]. The idea is presented for asymmetric case where the first source, is transmitted uncompressed and the second source is compressed using the parity check matrix of an LDPC code. Consider sources X and Y with probability distributions described above. The message sequence X is transmitted uncompressed and the message sequence Y is multiplied by the Parity Check matrix to form a syndrome. This syndrome is transmitted to the decoder. The decoder uses a modified version of Belief Propagation algorithm [15] to decode Y from X using this syndrome.

2.3 Cooperative Source Coding in Sensor Networks

2.3.1 Compression of correlated Gaussian sources

Consider a set of correlated Gaussian sources ($X_i \sim N(0,\sigma_i^2)$, $i = 1,2,...,m$) which are available at a joint encoder for compression. The Rate-Distortion function for this multivariate normal vector can be obtained by reverse water-filling on the eigenvalues [39].

The covariance matrix of these sources is defined as

$$C_{XX} = [\sigma_{ij}]_{m\times m} \tag{2.5}$$

$$\sigma_{ij} = E[X_i X_j] \quad , \quad i,j = 1,2,...,m \tag{2.6}$$

Since the sources are correlated, the non-diagonal elements of the covariance matrix are non-zero. Therefore, by applying *eigenvalue decomposition* method, we transform it into a diagonal matrix to achieve parallel (independent) Gaussian sources.

$$C_{XX} = U\Lambda U^T \tag{2.7}$$

Where,

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_m \end{bmatrix}_{m\times m} \quad \text{and} \quad UU^T = U^T U = I \tag{2.8}$$

U is the transform matrix which maps correlated sources $X_i \sim N(0,\sigma_i^2)$ to independent sources $\hat{X}_i \sim N(0,\lambda_i)$.

$$\hat{X} = U^T X \qquad , \qquad C_{\hat{X}\hat{X}} = \Lambda \tag{2.9}$$

Columns of U are normalized eigenvectors of C_{XX}.

In the new covariance matrix Λ, diagonal elements are the eigenvalues of the covariance matrix C_{XX}, and can be regarded as variances of transformed sources. Now, we can apply the Rate-Distortion theorem for parallel Gaussian sources [39].

***Theorem* [39] (*Rate-Distortion function for parallel Gaussian sources*)**

Let $\hat{X}_i \sim N(0, \lambda_i)$, $i = 1,2,...,m$ be independent Gaussian random variables and let the distortion measure be mean squared error. Then the rate distortion function is given by

$$R(D) = \sum_{i=1}^{m} \frac{1}{2} \log \frac{\lambda_i}{D_i} \quad (2.10)$$

where

$$D_i = \begin{cases} \delta & , \quad \delta < \lambda_i \\ \lambda_i & , \quad \delta \geq \lambda_i \end{cases} \quad (2.11)$$

and δ is chosen so that $\sum_{i=1}^{m} D_i = D$. This method is known as *Reverse Water-Filling* [39].

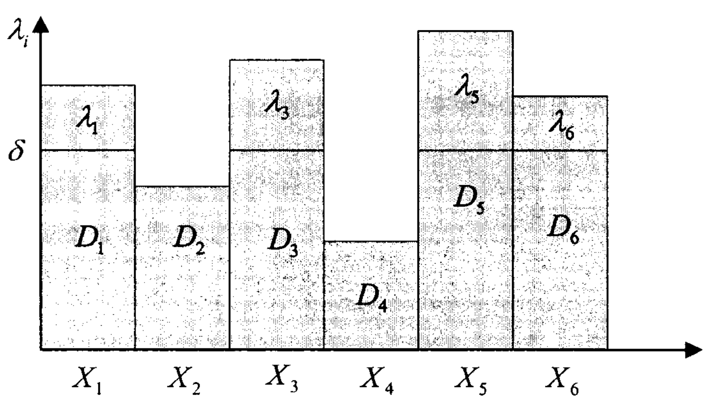

Figure 2.5 – Reverse Water-Filling for independent Gaussian random variables

Once transformed, the sources will be compressed according to the above rates and transmitted over the channel. At the receiver, they will be converted to X as $X = U\hat{X}$. We apply the above algorithm for joint compression of sensor observations in cooperative Wireless Sensor Networks.

2.3.2 Vector Quantization

Vector Quantization [36] (VQ) is a data compression method. In VQ, instead of quantizing each symbol separately, we apply the quantization rule to a vector of symbols. It can be used in many applications such as image compression, voice compression and pattern recognition. A Vector Quantizer maps a *k-dimensional* vector V^k in the vector space $\Re^k$ into a finite set of codewords $\{Q_i^k = (q_{i_1}, q_{i_2}, ..., q_{i_k}) \in \Re^k : i = 1,2,...,N\}$. The quantizer maps the vector to the associated codeword of its neighbor region. The neighbor region of Q_i^k is defined as

$$S_i = \left\{x \in \Re^k : \left\|x - Q_i^k\right\| \le \left\|x - Q_j^k\right\|, \forall j \ne i\right\} \tag{2.12}$$

where

$$\left\|x - Q_i^k\right\| = \sqrt{\sum_{j=1}^{k} (x_j - q_{i_j})^2} \tag{2.13}$$

Therefore, the vector space $\Re^k$ is partitioned into N neighbor regions $\{S_i : i = 1,2,...,N\}$ where, $\bigcup_{i=1}^{N} S_i = \Re^k$ and $\bigcap_{i=1}^{N} S_i = \varphi$.

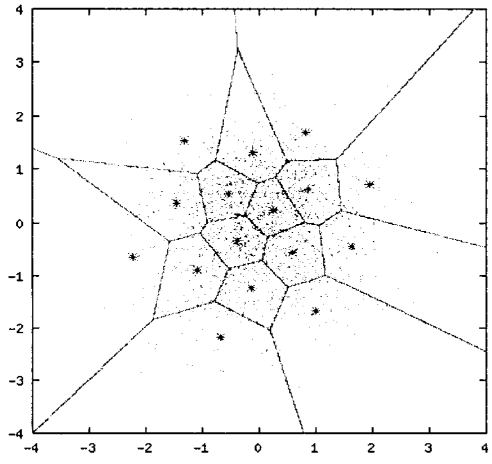

Figure 2.6 – Partitioning Example for R^2

Vector Quantizer takes the input vector, searches among all codewords to find the closest one to the input vector (which produces the least distortion) and outputs the index (address) of that codeword. At the receiver side, decoder takes the index and decodes the quantized vector. Vector quantization offers better performance than scalar quantization of the same rate, specially for correlated information [36]. The main problem with vector quantization is the increased complexity. As the number of codewords increases, the search complexity for optimum codeword increases exponentially [36]. Another challenge in VQ is the codebook design. Here, we present an algorithm which produces the best quantizer with Least Mean Square distortion.

Codebook Design

The distribution of data to be quantized is given. We generate a large number of random sample vectors with the given distribution and design the codebook for them. The procedure, known as Lloyd-Max [37]-[38] algorithm, is described bellow:

1. Determine the number of codewords (N)
2. Choose N random vectors from the set of generated samples as the initial codewords.
3. Calculate the Euclidean distance of all vectors from the codewords and put all of the vectors in the neighbor region of each codeword in one group.
4. To each group, assign a new codeword $\hat{Q}_i$ by taking the average over all sample vectors belonging to that group.
5. Continue steps 3 and 4 until the variation of distortion D is less than a defined threshold. That is, $0 < \frac{D_{old} - D_{new}}{D_{old}} < \varepsilon$.

2.4 Transmission in Cooperative Sensor Networks

In this section we study the problem of transmission in energy-constrained wireless sensor networks. In classical systems, nodes communicate to the center on a Single-Input Single-Output (SISO) basis where each node transmits its information directly to the center. Here, we present the cooperative transmission technique which exploits Space-Time coding for energy efficient transmission.

2.4.1 Cooperative (virtual) MIMO in sensor networks

In Wireless Sensor Networks, every single node is equipped with an antenna and they all wish to report their observations to a common center. This implies that if we consider the system in a Transmitter–Receiver context, on the transmission side we have multiple antennas. The advantage of having multiple antennas has motivated researchers in recent years to propose and develop the problem of Cooperative Communication in Wireless Sensor Networks [28]–[33]. In a typical sensor network nodes are located close to one another and their distance from each other is relatively shorter than their distance to the Fusion Center. Recently, a lot of advances have been achieved in energy-efficient multiple antenna communication systems known as Multiple-Input Multiple-Output (MIMO) technique, where information is encoded over space and time and is transmitted by multiple antennas. Wireless medium is affected by channel impairments such as fading and noise. MIMO techniques provide diversity gain as well as coding gain. Therefore, they have better performance (Bit Error Rate) than SISO systems. That is, MIMO systems offer the same performance as a SISO system by spending less power for transmission. The problem with MIMO systems is that they require more hardware and have higher complexity. Therefore, their circuit energy consumption is not negligible.

In [28], Cui et al presented an energy-efficient cooperative technique for sensor networks. In order to be able to apply MIMO techniques for transmission, nodes need to have knowledge about each others' information. Therefore, authors in [28] proposed local data exchange among sensor nodes. They studied cooperation for two nodes and applied 'Alamouti' Space-Time coding scheme. The network is divided into groups of two nodes and in each group, nodes confer with each other. Having exchanged their data, they encode it into Alamouti codewords and transmit the codewords with their antennas. The performance is the same as a system with two antennas but we have the overhead of transmission and reception of data in local exchange which requires additional circuitry and increased processing complexity (the communication module needs to be a transmitter/receiver rather than a transmitter). In [28] authors showed that if the distance between nodes and Fusion Center (long-haul transmission) is long enough, cooperative MIMO technique can be more energy-efficient than SISO. As in wireless sensor networks total energy consumption of the nodes is critical, we have to take into account both the transmission energy and consumed energy of the circuit. As analyzed in [28] and shown in Figures 2.7 and 2.8, Transmission energy of SISO is always higher than MIMO but when we consider Total energy consumption, for short distances, SISO outperforms cooperative MIMO.

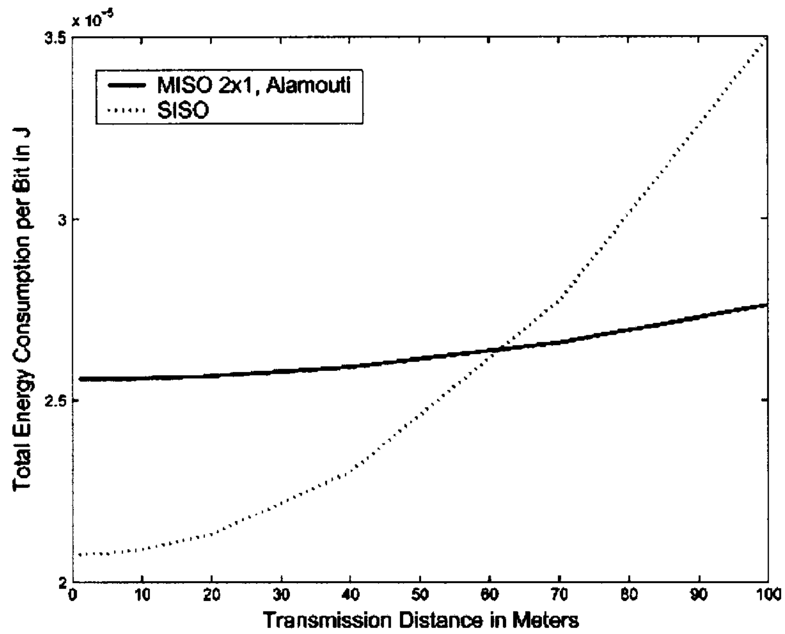

Figure 2.7 – Total energy consumption per bit over *d*, MISO vs. SISO [28]

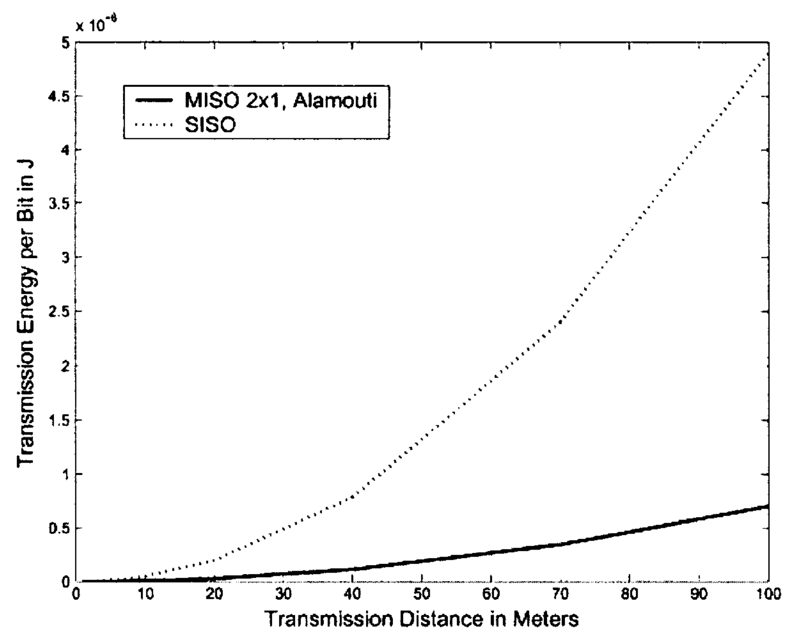

Figure 2.8 – Transmission energy consumption per bit over *d*, MISO vs. SISO [28]

The idea of Virtual (Cooperative) MIMO, proposed in [28], has been extended and analyzed in more details in [30] and [31]. In [28] authors assumed that perfect

Channel State Information (CSI) is known at the receiver while in [31] the training overhead required for channel estimation is also taken into account. The system model and parameters used in all these works are the same. The analysis is as follows:

Consider the system as a narrow-band, flat Rayleigh fading communication link with multiple antennas. The system has N_T Transmit antennas and N_R receive antennas. The energy consumption of baseband signal processing blocks is omitted and the system is assumed to be uncoded. The total power consumption is divided into two main components: power consumption of power amplifiers P_{PA} and circuit blocks P_C. Assuming that power consumption of amplifiers is linearly dependent on the transmit power P_{out}, it can be approximated as [31]

$$P_{PA} = (1+\alpha)P_{out} \tag{2.14}$$

where

$$\alpha = \frac{\xi}{\eta} - 1 \tag{2.15}$$

In the above equation, η is the drain efficiency of RF power amplifier and ξ is the peak-to-average ratio which depends on the modulation scheme and constellation size. For M-QAM modulation we have

$$\xi = 3\frac{M - 2\sqrt{M} + 1}{M - 1} \tag{2.16}$$

The transmit power P_{out} can be calculated as

$$P_{out} = \frac{(4\pi)^2 d^\kappa M_l N_f}{G_t G_r \lambda^2} \overline{E}_b R_b \tag{2.17}$$

where d is the transmission distance and κ is the channel path loss exponent which usually lies between 2–4 for wireless channels (κ=2 corresponds to free space propagation). G_t and G_r are the antenna gains for transmitter and receiver and λ is the wavelength. M_l is defined as the link margin to compensate the hardware process variations and other additive background noise or interference. $N_f = \frac{N_r}{N_o}$ is the receiver noise figure where N_r is the power spectral density (PSD) of the total effective noise at the receiver input and N_o is the single-sided thermal noise PSD at room temperature. $\overline{E}_b$ is the average energy per bit required for a given BER and R_b is the system bit rate. The block diagram of Transmitter and Receiver circuits is shown in Fig. 2.9 and Fig. 2.10.

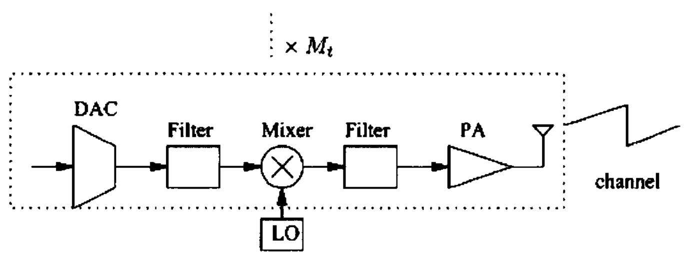

Figure 2.9 – Transmitter circuit blocks [28]

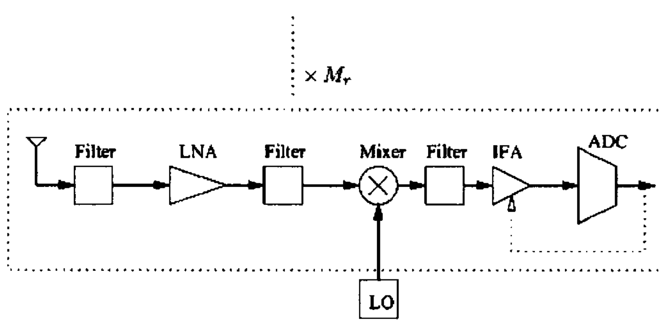

Figure 2.10 – Receiver circuit blocks [28]

As assumed in [28], [30] and [31],we consider that the frequency synthesizer is shared among all the antennas in MIMO system. We can estimate the total power consumption of the circuit as

$$P_C \approx N_T(P_{DAC} + P_{mix} + P_{filt}) + 2P_{synth} + N_R(P_{LNA} + P_{mix} + P_{IFA} + P_{filr} + P_{ADC}) \quad (2.18)$$

where P_{DAC} , P_{mix} , P_{filt} , P_{synth} , P_{LNA} , P_{IFA} , P_{filr} and P_{ADC} are the power consumption values of Digital to Analog (D/A) converter, mixer, active filters of transmitter, frequency synthesizer, Low Noise Amplifier, Intermediate Frequency Amplifier, active filters of receiver and Analog to Digital (A/D) converter respectively. The assumed values for these parameters are listed in Table 2.1.

P_{DAC} = 15.7 mW	P_{ADC} = 6.7 mW
P_{filr} = P_{filt} = 2.5 mW	f_c = 2.5 GHz
P_{mix} = 30.3 mW	P_{syn} = 50 mW
P_{LNA} = 20 mW	P_{IFA} = 3 mW
$G_t G_r$ = 5 dBi	N_0 = – 171 dBm/Hz
M_l = 40 dB	N_f = 10 dB

Table 2.1 – System Parameters

Total energy per bit for a fixed rate system can be estimated as

$$E_{bt} = \frac{P_{PA} + P_C}{R_b} \tag{2.19}$$

The bit error rate of an M-QAM MIMO system ($M = 2^b$) with a square constellation (b is even) is given by [31]

$$\overline{P}_b = \frac{4}{b}\left(1 - \frac{1}{2^{b/2}}\right)\frac{1}{2^{N_T N_R}}\left(1 - \frac{1}{\sqrt{1 + \frac{1}{\overline{E}_b / 2N_o}}}\right)^{N_T N_R} \sum_{k=0}^{N_T N_R - 1} \frac{1}{2^k}\binom{N_T N_R - 1 + k}{k}\left(1 + \frac{1}{\sqrt{1 + \frac{1}{\overline{E}_b / 2N_o}}}\right)^k \tag{2.20}$$

When b is odd, we may use (2.20) as an upper bound for the BER after dropping the term $\left(1 - \frac{1}{2^{b/2}}\right)$. By inverting (2.20) we can obtain the required $\overline{E}_b$ for any BER value $\overline{P}_b$.

If we substitute for the above values in equations, we will have the relationship between total energy consumption per bit and distance. As noted in [28] optimizing the constellation size also minimizes energy consumption. As simulated in [31], following figures illustrate the Total Energy consumption versus constellation size and distance.

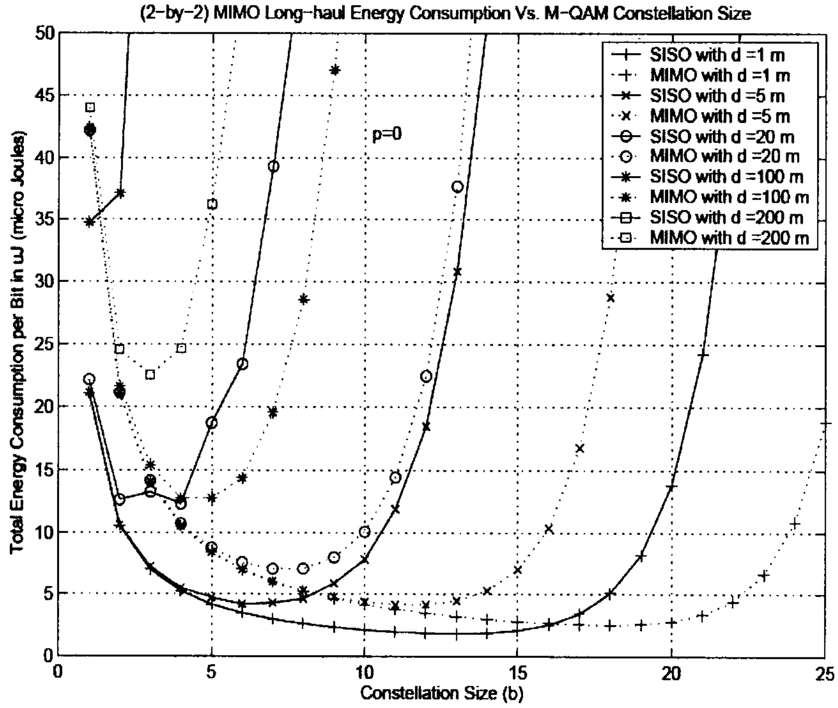

Figure 2.11 – Total Energy consumption per bit vs. Constellation size [31]

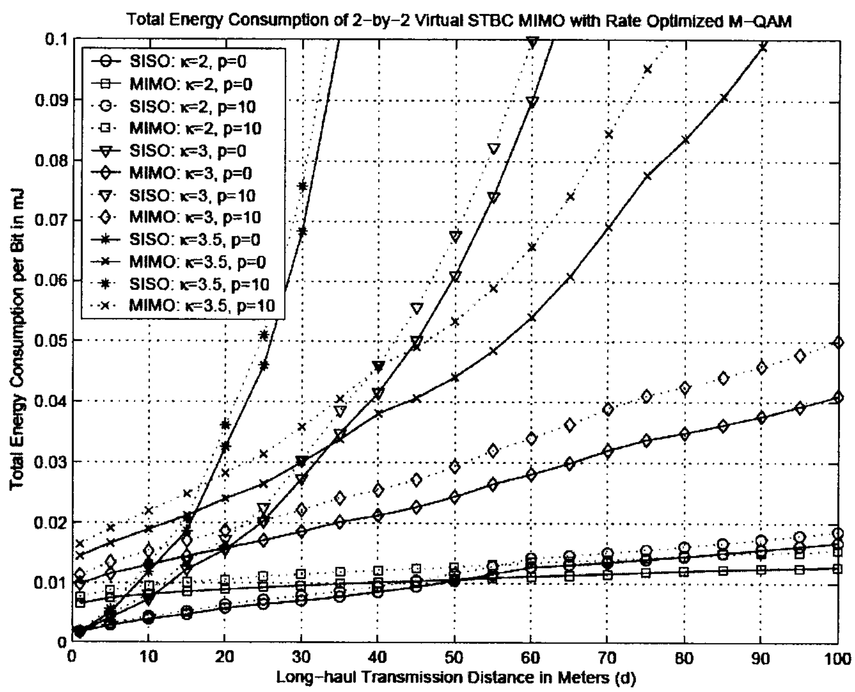

Figure 2.12 – Total Energy consumption per bit vs. Long-haul transmission distance [31]

2.4.2 Space-Time block codes

One of the fundamental characteristics of the wireless channel is multipath fading. That is, transmitted signal is severely attenuated after passing through wireless environment. Fading is a random process that can be modeled as the channel gain. Since the value of fading is random, it is better to have multiple replicas of the transmitted signal at the receiver. Having multiple copies of the signal, we can combine them and detect the signal. This can be done by having diversity in time (like using error correcting codes) or space (using multiple antennas). Recently there have been many advances in Space-Time Coding.

In Space-Time coding signal is encoded over time and space and is transmitted through multiple antennas at the same time. A simple transmit diversity scheme has been proposed by Alamouti [34] in 1998. His proposed scheme is as follows:

As shown in Fig. 2.13, the scheme has two transmit antennas and one receive antenna. At a given symbol period, two signals are transmitted simultaneously by transmit antennas. Suppose that s_0 and s_1 are two consecutive symbols.

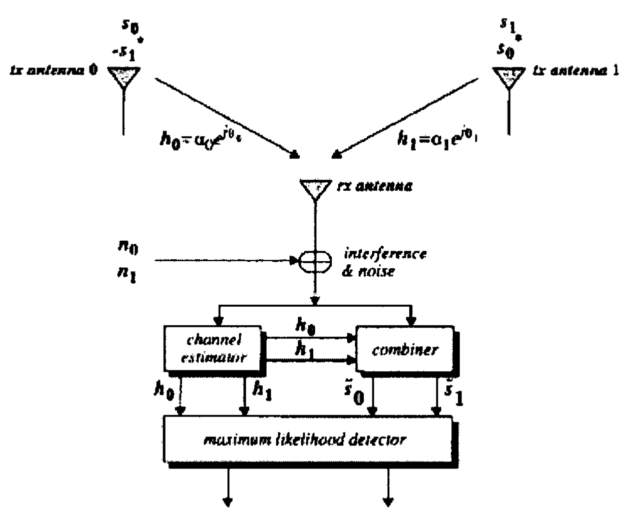

Figure 2.13 Two transmit diversity scheme with one receiver [34]

For the first symbol period, s_0 is transmitted over antenna 0 and s_1 is transmitted over antenna 1. During the next symbol period, $(-s_1^*)$ will be transmitted by antenna 0 and s_0^* will be transmitted by antenna 1, where (*) denotes complex conjugate operation.

The channel gains are modeled as multiplicative distortion and we assume that they are constant during two consecutive symbol periods. As indicated in Fig 2.13, channel coefficient is denoted as h_0 for transmit antenna 0 and h_1 for transmit antenna 1. Received signals for two consecutive symbol periods are:

$$r_o = h_0 s_0 + h_1 s_1 + n_0 \tag{2.21}$$

$$r_1 = -h_0 s_1^* + h_1 s_0^* + n_1 \tag{2.22}$$

After channel estimation (h_0 and h_1) the received signals will be combined at the receiver.

$$\tilde{s}_0 = h_0^* r_0 + h_1 r_1^* = (h_0^2 + h_1^2) s_0 + h_0^* n_0 + h_1 n_1^* \tag{2.23}$$

$$\tilde{s}_1 = h_1^* r_0 - h_0 r_1^* = (h_0^2 + h_1^2) s_1 - h_0 n_1^* + h_1^* n_0 \tag{2.24}$$

These combined signals are then sent to the maximum likelihood detector and s_0 and s_1 will be detected. Alamouti coding scheme is based on orthogonal design and that makes it easy and simple to decode. Orthogonality means that inner product of columns in the coding matrix is equal to zero.

$$\begin{bmatrix} s_0 & -s_1^* \\ s_1 & s_0^* \end{bmatrix} \begin{matrix} \leftarrow \textit{Transmit Antenna}\, 1 \\ \leftarrow \textit{Transmit Antenna}\, 2 \end{matrix}$$

$$\text{Inner product} = s_0.(-s_1^*)^* + s_1.(s_0^*)^* = -s_0.s_1 + s_1.s_0 = 0$$

Alamouti scheme has been extended for more than two antennas in [35] and is referred to as *Space-Time Block Coding (STBC)*. In STBC, data is encoded using a Space-Time block code and is split into m streams which is simultaneously transmitted using m antennas. The received signal at each receive antenna is linear superposition of attenuated transmitted signals, impaired by channel fading. Using the orthogonal structure of Space-Time Block Codes, signal streams can be decoupled and detected by maximum likelihood decoding. Orthogonal designs offer maximum achievable diversity order for any given number of Transmit antennas. It is possible to construct orthogonal STBC for complex and real constellations. For real constellations (such as PAM), maximum possible transmission rate can be achieved and for any complex constellation (such as PSK or QAM) we can always design a code of rate $1/2$ for any number of transmit antennas. For special cases of two, three and four transmit antennas, Space-Time Block Codes are designed that achieve rates 1, $3/4$ and $3/4$, respectively. The designed codes are:

Two transmit antennas; Rate One (Alamouti):

$$\begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix} \tag{2.25}$$

Three transmit antennas; Rate $3/4$:

$$\begin{bmatrix} s_1 & -s_2^* & \frac{s_3^*}{\sqrt{2}} & \frac{s_3}{\sqrt{2}} \\ s_2 & s_1^* & \frac{s_3^*}{\sqrt{2}} & -\frac{s_3}{\sqrt{2}} \\ \frac{s_3}{\sqrt{2}} & \frac{s_3}{\sqrt{2}} & \frac{-s_1 - s_1^* + s_2 - s_2^*}{2} & \frac{s_2 + s_2^* + s_1 - s_1^*}{2} \end{bmatrix} \tag{2.26}$$

Four transmit antennas; Rate $\frac{3}{4}$:

$$\begin{bmatrix} s_1 & -s_2^* & \frac{s_3^*}{\sqrt{2}} & \frac{s_3^*}{\sqrt{2}} \\ s_2 & s_1^* & \frac{s_3^*}{\sqrt{2}} & -\frac{s_3^*}{\sqrt{2}} \\ \frac{s_3}{\sqrt{2}} & \frac{s_3}{\sqrt{2}} & \frac{-s_1 - s_1^* + s_2 - s_2^*}{2} & \frac{s_2 + s_2^* + s_1 - s_1^*}{2} \\ \frac{s_3}{\sqrt{2}} & -\frac{s_3}{\sqrt{2}} & \frac{-s_2 - s_2^* + s_1 - s_1^*}{2} & -\frac{s_1 + s_1^* + s_2 - s_2^*}{2} \end{bmatrix} \tag{2.27}$$

For any complex constellation with at least four transmit antennas we can design an STBC with rate $\frac{1}{2}$. For instance,

$$\begin{bmatrix} s_1 & -s_2 & -s_3 & -s_4 & s_1^* & -s_2^* & -s_3^* & -s_4^* \\ s_2 & s_1 & s_4 & -s_3 & s_2^* & s_1^* & s_4^* & -s_3^* \\ s_3 & -s_4 & s_1 & s_2 & s_3^* & -s_4^* & s_1^* & s_2^* \\ s_4 & s_3 & -s_2 & s_1 & s_4^* & s_3^* & -s_2^* & s_1^* \end{bmatrix} \tag{2.28}$$

For any real constellation we can design an orthogonal STBC of full rate, for instance:

$$\begin{bmatrix} s_1 & -s_2 & -s_3 & -s_4 & -s_5 & -s_6 & -s_7 & -s_8 \\ s_2 & s_1 & -s_4 & s_3 & -s_6 & s_5 & s_8 & -s_7 \\ s_3 & s_4 & s_1 & -s_2 & -s_7 & -s_8 & s_5 & s_6 \\ s_4 & -s_3 & s_2 & s_1 & -s_8 & s_7 & -s_6 & s_5 \\ s_5 & s_6 & s_7 & s_8 & s_1 & -s_2 & -s_3 & -s_4 \\ s_6 & -s_5 & s_8 & -s_7 & s_2 & s_1 & s_4 & -s_3 \\ s_7 & -s_8 & -s_5 & s_6 & s_3 & -s_4 & s_1 & s_2 \\ s_8 & s_7 & -s_6 & -s_5 & s_4 & s_3 & -s_2 & s_1 \end{bmatrix} \tag{2.29}$$

CHAPTER THREE

CEO PROBLEM FOR CORRELATED BINARY SOURCES

Consider the problem of estimating a binary sequence from a set of "Correlated Binary Sources". We have a set of binary sources whose data are correlated and we want to compress their data and send them to the Fusion Center. The goal of the fusion center is to extract the overall information from the set of data received from the sources and estimate an unknown phenomenon. This problem can be regarded as 'Binary' Chief Executive Officer (CEO) problem [5], where a CEO employs a team of agents to observe independently corrupted versions of a phenomenon and report the observations to him. We consider that the phenomenon to be observed is a binary source and all agents observe binary data which are independently corrupted by errors. For example, sources can be digital video camera arrays.

As the sources' data are correlated, according to Slepian-Wolf theorem [8], they can be encoded separately and decoded jointly with a side information decoder. In source coding with side information one of the sources compresses its data according to its own entropy and other sources compress their data according to the conditional entropy. That is, if X_1 and X_2 denote the correlated binary sources, node 1 encodes X_1 into $H(X_1)$ bits and node 2 encodes its data into $H(X_2 \mid X_1)$ bits. Decoder uses this information together with information received from X_1 (side information) to decode X_2.

We define the "Binary CEO Problem" in sensor networks as following: There are a number of sensors (agents) with a certain precision in sensing and each of them observes a binary source with an average probability of bit error, ' p_e '. The CEO is interested in having a more precise observation on the phenomenon (i.e. average probability of bit error less than p_e). Therefore, it deploys a number of these sensors and by processing all observations, detects the phenomenon with any arbitrary average probability of error.

First, we define the correlation and present the system model. We then analyze the systems' behaviour considering entropy as the measure of uncertainty and prove the convergence of estimation analytically. Using the *Fano's inequality* [39] we determine the theoretical (minimum) number of agents required to observe the phenomenon for any given distortion. Finally, we find the estimation rule for extracting the information from the set of observations.

3.1 System Model

We denote the unknown phenomenon to be estimated as Y, the number of sensors as n, the sensor observations as (X_i , i = 1, 2, …, n) and the final estimation as $\hat{Y}$. Sensor observations (X_i) are corrupted versions of Y. Meaning that, X_i 's are modulo-2 addition of Y with a random error (E_i , i = 1, 2, …, n). E_i is a 'Bernoulli' random variable with probability of being 1 equal to ' p_e '.

$$X_i = Y + E_i \tag{3.1}$$

$$\Pr\ \{E_i = 1\} = p_e \tag{3.2}$$

$$\Pr\ \{E_i = 0\} = 1 - p_e \tag{3.3}$$

Observations (X_i's) are conditionally independent given the source (Y).

As Y is a binary source, its entropy is less than or equal to 1,

$$H(Y) \leq H(\frac{1}{2}) = 1 \tag{3.4}$$

Without loss of generality we assume that source Y is unbiased and we have

$$\Pr\{\ Y = 0\ \} = \Pr\{\ Y = 1\ \} = \frac{1}{2} \tag{3.5}$$

$$H(Y) = 1 \tag{3.6}$$

We denote the received vector at the Fusion Center by $\underline{X} = (X_1, X_2, ..., X_n)$ and the desired average probability of error by ' p_r '.

$$p_r = \Pr\{\hat{Y} \neq Y\} \tag{3.7}$$

3.2 Theoretical analysis for required number of agents

In this system we observe a random vector $\underline{X}$ and wish to estimate the value of the random variable Y which is highly correlated to the entries of $\underline{X}$ with a low probability of error $p_r = \Pr\{\hat{Y} \neq Y\}$. This can be done if the conditional entropy $H(Y \mid \underline{X})$ is small. We calculate a function $\hat{Y} = f(\underline{X})$ which is an estimate of Y and wish to bound the probability that $\hat{Y} \neq Y$. We observe that $Y \to \underline{X} = (X_1, X_2, ..., X_n) \to \hat{Y}$ form a Markov chain. Therefore, based on *Fano's inequality* [39] we have

$$H(p_r) + p_r \log(|Y| - 1) \geq H(Y \mid \underline{X}) \tag{3.8}$$

As we consider Y to be a binary source, its cardinality is $|Y| = 2$ and the inequality reduces to

$$H(Y \mid \underline{X}) \leq H(p_r) \tag{3.9}$$

We have to calculate the conditional entropy $H(Y \mid \underline{X}) = H(Y \mid X_1, X_2, ..., X_n)$ as a function of (n, p_e) and relate it to the desired probability of error (p_r).

From the chain rule in joint entropy we have

$$\begin{aligned} H(Y, X_1, X_2, ..., X_n) &= H(Y) + H(X_1, X_2, ..., X_n \mid Y) \\ &= H(X_1, X_2, ..., X_n) + H(Y \mid X_1, X_2, ..., X_n) \end{aligned} \tag{3.10}$$

Since the additive errors in observations (E_i's) are independent of each other,

$$\begin{aligned} H(X_1, X_2, ..., X_n \mid Y) &= H(X_1 \oplus Y, X_2 \oplus Y, ..., X_n \oplus Y \mid Y) = H(E_1, E_2, ..., E_n \mid Y) \\ &= \sum_{i=1}^{n} H(E_i \mid Y) = \sum_{i=1}^{n} H(E_i) = nH(p_e) \end{aligned} \tag{3.11}$$

Substituting (3.11) and the fact that $H(Y) = 1$, into (3.10), we will have

$$H(Y \mid X_1, X_2, ..., X_n) = 1 + nH(p_e) - H(X_1, X_2, ..., X_n) \tag{3.12}$$

From the definition of joint entropy, we have

$$H(X_1, X_2, ..., X_n) = -\sum_i p_i(X_1, X_2, ..., X_n) \log p_i(X_1, X_2, ..., X_n)$$

$$= -\sum_i [p_i(X_1, X_2, ..., X_n \mid Y = 0) \Pr(Y = 0) + p_i(X_1, X_2, ..., X_n \mid Y = 1) \Pr(Y = 1)]$$

$$\times \log[p_i(X_1, X_2, ..., X_n \mid Y = 0) \Pr(Y = 0) + p_i(X_1, X_2, ..., X_n \mid Y = 1) \Pr(Y = 1)] \quad (3.13)$$

Where $\sum_i$ means summation over all possible probabilities of $\underline{X} = (X_1, X_2, ..., X_n)$.

The joint probability distribution function $p_i(X_1, X_2, ..., X_n \mid Y = 0)$ has Binomial distribution. That is, for any $0 \le k \le n$, there are $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ combinations of $p_i(X_1, X_2, ..., X_n \mid Y = 0)$ with probability $p_e^{\,k}(1 - p_e)^{n-k}$ that have k ones and $(n - k)$ zeros and with similar analysis, there are $\binom{n}{k}$ combinations of $p_i(X_1, X_2, ..., X_n \mid Y = 1)$ with probability $(1 - p_e)^k p_e^{\,n-k}$ that have k ones and $(n - k)$ zeros. Therefore, if we substitute these values in (3.13) and take the summation over all possible combinations, we will have

$$H(X_1, X_2, ..., X_n) = -\sum_{k=0}^{n} \binom{n}{k} \left[\frac{1}{2} p_e^{\,k}(1 - p_e)^{n-k} + \frac{1}{2}(1 - p_e)^k p_e^{n-k}\right]$$

$$\times \log\left[\frac{1}{2} p_e^{\,k}(1 - p_e)^{n-k} + \frac{1}{2}(1 - p_e)^k p_e^{n-k}\right]$$

$$= -\sum_{k=0}^{n} \frac{1}{2}\binom{n}{k} p_e^{\,k}(1 - p_e)^{n-k} \log\left[\frac{p_e^{\,k}(1 - p_e)^{n-k} + (1 - p_e)^k p_e^{\,n-k}}{2}\right]$$

$$-\sum_{k=0}^{n} \frac{1}{2}\binom{n}{k} (1 - p_e)^k p_e^{\,n-k} \log\left[\frac{p_e^{\,k}(1 - p_e)^{n-k} + (1 - p_e)^k p_e^{\,n-k}}{2}\right] \quad (3.14)$$

In the last summation we define the variable $l = n - k$ and substitute it for k.

$$-\sum_{k=0}^{n}\frac{1}{2}\binom{n}{k}(1-p_e)^k p_e^{\,n-k}\log\left[\frac{p_e^{\,k}(1-p_e)^{n-k}+(1-p_e)^k p_e^{\,n-k}}{2}\right]$$

$$=-\sum_{l=0}^{n}\frac{1}{2}\binom{n}{n-l}(1-p_e)^{n-l} p_e^{\,l}\log\left[\frac{p_e^{\,n-l}(1-p_e)^{l}+(1-p_e)^{n-l} p_e^{\,l}}{2}\right]$$

$$=-\sum_{l=0}^{n}\frac{1}{2}\binom{n}{l}(1-p_e)^{n-l} p_e^{\,l}\log\left[\frac{p_e^{\,n-l}(1-p_e)^{l}+(1-p_e)^{n-l} p_e^{\,l}}{2}\right] \qquad (3.15)$$

Now if we change the name of auxiliary variable l with k, we will have

$$H(X_1,X_2,\ldots,X_n)=-\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[\frac{p_e^{\,k}(1-p_e)^{n-k}+(1-p_e)^k p_e^{\,n-k}}{2}\right]$$

$$=1-\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[p_e^{\,k}(1-p_e)^{n-k}+(1-p_e)^k p_e^{\,n-k}\right] \quad (3.16)$$

From independence of E_1 , E_2 , … , E_n we know that $H(E_1,E_2,\ldots,E_n)=nH(p_e)$.

On the other hand,

$$H(E_1,E_2,\ldots,E_n)=-\sum_i p_i(E_1,E_2,\ldots,E_n)\log p_i(E_1,E_2,\ldots,E_n)$$

$$=-\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[p_e^{\,k}(1-p_e)^{n-k}\right] \qquad (3.17)$$

Thus,

$$H(Y\mid X_1,X_2,\ldots,X_n)=1+nH(p_e)-H(X_1,X_2,\ldots,X_n)$$

$$=1-\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[p_e^{\,k}(1-p_e)^{n-k}\right]$$

$$-\left(1-\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[p_e^{\,k}(1-p_e)^{n-k}+(1-p_e)^k p_e^{\,n-k}\right]\right)$$

$$=\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[p_e^{\,k}(1-p_e)^{n-k}+(1-p_e)^k p_e^{\,n-k}\right]$$

$$-\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[p_e^{\,k}(1-p_e)^{n-k}\right] \qquad (3.18)$$

Finally, the closed form equation for entropy of Y given $\underline{X} = (X_1, X_2, ..., X_n)$ will be

$$H(Y \mid X_1, X_2, ..., X_n) = \sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[1 + \left(\frac{p_e}{1-p_e}\right)^{n-2k}\right]. \tag{3.19}$$

According to *DeMoivre-Laplace Theorem* [40], the asymptotic form of Binomial distribution with parameters p_e and $1-p_e$, when $n \to \infty$, for k in the $\sqrt{np_e(1-p_e)}$ neighbourhood of np_e can be approximated by

$$\binom{n}{k} p_e^k (1-p_e)^{n-k} \cong \frac{1}{\sqrt{2\pi np_e(1-p_e)}} e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}} \tag{3.20}$$

The normal function is a close approximation for binomial distribution and in the limit when n tends to infinity, the approximation can be stated as equality. Therefore, for n very large, we can approximate $\binom{n}{k} p_e^k (1-p_e)^{n-k}$ by an upper bound which is the *Normal* function with mean np_e and standard deviation $\sqrt{np_e(1-p_e)}$. The mean grows linearly with n while the standard deviation grows with $\sqrt{n}$.

Going back to (3.18), by adding and subtracting $\sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[\binom{n}{k}\right]$ we have

$$\begin{aligned} H(Y \mid X_1, X_2, ..., X_n) = &\sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[p_e^{\,k} (1-p_e)^{n-k} + (1-p_e)^k p_e^{\,n-k}\right] \\ &+ \sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[\binom{n}{k}\right] - \sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[\binom{n}{k}\right] \\ &- \sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[p_e^{\,k} (1-p_e)^{n-k}\right] \\ &= \sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[\binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} + \binom{n}{k} (1-p_e)^k p_e^{\,n-k}\right] \end{aligned}$$

$$-\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\right]$$

$$=\sum_{k=0}^{n}\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}\log\left[1+\frac{\binom{n}{k}(1-p_e)^k p_e^{\,n-k}}{\binom{n}{k}p_e^{\,k}(1-p_e)^{n-k}}\right] \qquad (3.21)$$

We substitute (3.20) in (3.21),

$$\lim_{n\to\infty}H(Y\mid X_1,X_2,\ldots,X_n)=\lim_{n\to\infty}\sum_{k=0}^{n}\frac{1}{\sqrt{2\pi np_e(1-p_e)}}e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}}\log(1+e^{-\frac{[k-n(1-p_e)]^2-(k-np_e)^2}{2np_e(1-p_e)}})$$

$$=\lim_{n\to\infty}\sum_{k=0}^{n}\frac{1}{\sqrt{2\pi np_e(1-p_e)}}e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}}\log(1+e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}}) \qquad (3.22)$$

We will prove that the above summation has a limit and converges to zero.

We find the Maximum of the sequence $S(k)=\frac{1}{\sqrt{2\pi np_e(1-p_e)}}e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}}\log(1+e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}})$

by taking the first derivative and setting it equal to zero.

$$\frac{d}{dk}S(k)=\frac{d}{dk}\left[\frac{1}{\sqrt{2\pi np_e(1-p_e)}}e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}}\log(1+e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}})\right]$$

$$=\frac{1}{\sqrt{2\pi np_e(1-p_e)}}\left(\frac{np_e-k}{np_e(1-p_e)}e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}}\log(1+e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}})+\frac{\log(e)(1-2p_e)e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}}}{p_e(1-p_e)\left[1+e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}}\right]}e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}}\right)$$

$$=\frac{e^{-\frac{(k-np_e)^2}{2np_e(1-p_e)}}}{\sqrt{2\pi n}\left[p_e(1-p_e)\right]^{\frac{3}{2}}}\left((p_e-\frac{k}{n})\log(1+e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}})+\log(e)(1-2p_e)\frac{e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}}}{1+e^{\frac{(2k-n)(1-2p_e)}{2p_e(1-p_e)}}}\right)=0 \quad (3.23)$$

We denote the value of k that maximizes $S(k)$ by k_m.

In order to make the notations simpler, we define new variables

$$\varepsilon = 2k_m - n \quad , \quad A = \frac{1-2p_e}{2p_e(1-p_e)} \tag{3.24}$$

$$k_m = \frac{n+\varepsilon}{2} \quad , \quad \frac{(2k_m - n)(1-2p_e)}{2p_e(1-p_e)} = A\varepsilon \tag{3.25}$$

Equation (3.23) reduces to

$$(p_e - \frac{k_m}{n})\log(1+e^{A\varepsilon}) + \log(e)(1-2p_e)\frac{e^{A\varepsilon}}{1+e^{A\varepsilon}} = 0 \tag{3.26}$$

We substitute for k_m

$$(\frac{1}{2} + \frac{\varepsilon}{2n} - p_e)\ln(1+e^{A\varepsilon}) = (1-2p_e)\frac{e^{A\varepsilon}}{1+e^{A\varepsilon}} \tag{3.27}$$

$$\left[\frac{\varepsilon}{n} + (1-2p_e)\right]\ln(1+e^{A\varepsilon}) = (1-2p_e)\frac{2e^{A\varepsilon}}{1+e^{A\varepsilon}} \tag{3.28}$$

We claim that $\lim\limits_{n\to\infty}\frac{\varepsilon}{n} = 0$. As defined earlier, $\varepsilon = 2(k_m - \frac{n}{2})$.

Since $0 \le k_m \le n$ from (3.24) we have $-n \le \varepsilon \le n$ and $-1 \le \lim\limits_{n\to\infty}\frac{\varepsilon}{n} \le 1$.

We calculate the limit of (3.28) when $n \to \infty$

$$\lim_{n\to\infty}\left\{\left[\frac{\varepsilon}{n} + (1-2p_e)\right]\ln(1+e^{A\varepsilon})\right\} = \lim_{n\to\infty}\left\{(1-2p_e)\frac{2e^{A\varepsilon}}{1+e^{A\varepsilon}}\right\} \tag{3.29}$$

We assume that $\lim\limits_{n\to\infty}\frac{\varepsilon}{n} = L$, $-1 \le L \le 1$

If $L > 0$ we have $\varepsilon = nL$, $\varepsilon \to \infty$

$$\lim_{n\to\infty}\{\ln(1+e^{A\varepsilon})\} = A\varepsilon \tag{3.30}$$

$$\lim_{n\to\infty}\{\frac{2e^{A\varepsilon}}{1+e^{A\varepsilon}}\} = 2 \tag{3.31}$$

$$\lim_{n\to\infty}\{[L+(1-2p_e)]A\varepsilon\}=2(1-2p_e) \tag{3.32}$$

$$\lim_{n\to\infty}\varepsilon=\frac{4p_e(1-2p_e)}{L+1-2p_e} \tag{3.33}$$

Equation (3.33) implies that $\lim_{n\to\infty}\varepsilon$ is bounded, and $L=\lim_{n\to\infty}\frac{\varepsilon}{n}=0$ which contradicts with the assumption that $L>0$.

If $L<0$ we have $\varepsilon=nL$, $\varepsilon\to-\infty$

$$\lim_{n\to\infty}\{\ln(1+e^{A\varepsilon})\}=e^{A\varepsilon} \tag{3.34}$$

$$\lim_{n\to\infty}\{\frac{2e^{A\varepsilon}}{1+e^{A\varepsilon}}\}=2e^{A\varepsilon} \tag{3.35}$$

$$\lim_{n\to\infty}\{[L+(1-2p_e)]e^{A\varepsilon}\}=(1-2p_e)2e^{A\varepsilon} \tag{3.36}$$

$$L=1-2p_e>0 \tag{3.37}$$

Equation (3.37) contradicts with the assumption that $L<0$. Therefore, the only solution is that $L = 0$.

Going back to (3.29), $\lim_{n\to\infty}\left\{\left[\frac{\varepsilon}{n}+(1-2p_e)\right]\ln(1+e^{A\varepsilon})\right\}=\lim_{n\to\infty}\left\{(1-2p_e)\frac{2e^{A\varepsilon}}{1+e^{A\varepsilon}}\right\}$.

Substituting $L = 0$ in (3.29), we have

$$\ln(1+e^{A\varepsilon})=\frac{2e^{A\varepsilon}}{1+e^{A\varepsilon}} \tag{3.38}$$

We solve the above equation and find $A\varepsilon=1.3665$ and from (3.25) we will have

$$\varepsilon=\frac{2p_e(1-p_e)}{1-2p_e}(1.3665)=\frac{2.733p_e(1-p_e)}{1-2p_e} \tag{3.39}$$

Equation (3.39) states that for large values of n , ε is only a function of p_e and for n sufficiently large, from (3.24) we have

$$k_m = \frac{n+\varepsilon}{2} = \frac{n}{2} + \frac{1.3665 p_e(1-p_e)}{1-2p_e} \tag{3.40}$$

Substituting k_m into $S(k)$

$$\lim_{n\to\infty} S(k_m) = \lim_{n\to\infty} \frac{1}{\sqrt{2\pi n p_e(1-p_e)}} e^{-\frac{(k_m - np_e)^2}{2np_e(1-p_e)}} \log(1 + e^{\frac{(2k_m-n)(1-2p_e)}{2p_e(1-p_e)}})$$

$$= \lim_{n\to\infty} \frac{1}{\sqrt{2\pi n p_e(1-p_e)}} e^{-\frac{(n+\varepsilon-2np_e)^2}{8np_e(1-p_e)}} \log(1 + e^{A\varepsilon})$$

$$= \lim_{n\to\infty} \frac{2.2991}{\sqrt{2\pi n p_e(1-p_e)}} e^{-\frac{n(1-2p_e)^2}{8p_e(1-p_e)}} = \lim_{n\to\infty} \frac{e^{-n}}{\sqrt{n}} \tag{3.41}$$

Going back to (3.22) and substituting (3.41)

$$\lim_{n\to\infty} H(Y \mid X_1, X_2, \ldots, X_n) = \lim_{n\to\infty} \sum_{k=0}^{n} S(k) < \lim_{n\to\infty} \sum_{k=0}^{n} S(k_m) = \lim_{n\to\infty} (n+1) \frac{e^{-n}}{\sqrt{n}} = \lim_{n\to\infty} \sqrt{n} e^{-n} = 0 \tag{3.42}$$

In order to illustrate the behaviour of system when the number of observers increases, we sketch $H(Y \mid X_1, X_2, \ldots, X_n)$ for different values of p_e.

$$H(Y \mid X_1, X_2, \ldots, X_n) = \sum_{k=0}^{n} \binom{n}{k} p_e^{\,k} (1-p_e)^{n-k} \log\left[1 + \left(\frac{p_e}{1-p_e}\right)^{n-2k}\right]$$

As shown in Fig. 3.1, when the number of observers increases, the uncertainty about Y decreases and tends to zero. For smaller values of p_e the convergence is faster and less number of observers is needed.

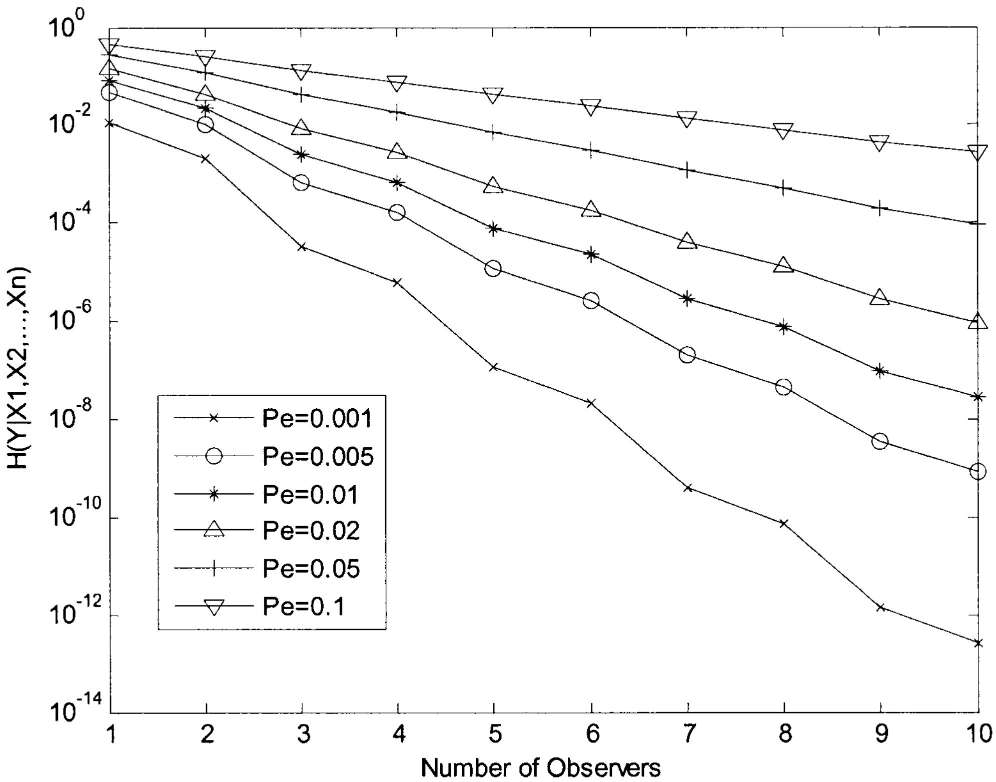

Figure 3.1 – Comparison of speed of convergence for different values of p_e

3.3 Estimation Rule

We calculate the likelihood ratio and derive the Maximum Likelihood estimation rule for *Y*.

We denote the *Likelihood Ratio* by Λ_L.

$$\Lambda_L = \log \frac{\Pr(Y=0 \mid X_1, X_2, ..., X_n)}{\Pr(Y=1 \mid X_1, X_2, ..., X_n)} \tag{3.43}$$

If $\Lambda_L > 0$, we estimate $Y = 0$ and if $\Lambda_L < 0$, we estimate $Y = 1$.

Based on Bayes' Theorem [40] we have

$$\Pr(Y \mid X_1, X_2, ..., X_n) = \frac{\Pr(X_1, X_2, ..., X_n \mid Y)}{\Pr(X_1, X_2, ..., X_n)} \Pr(Y) \tag{3.44}$$

Therefore,

$$\Lambda_L = \log\frac{\Pr(Y=0 \mid X_1, X_2, \ldots, X_n)}{\Pr(Y=1 \mid X_1, X_2, \ldots, X_n)} = \log\frac{\dfrac{\Pr(X_1, X_2, \ldots, X_n \mid Y=0)}{\Pr(X_1, X_2, \ldots, X_n)}\Pr(Y=0)}{\dfrac{\Pr(X_1, X_2, \ldots, X_n \mid Y=1)}{\Pr(X_1, X_2, \ldots, X_n)}\Pr(Y=1)}$$

$$= \log\frac{\Pr(Y=0)}{\Pr(Y=1)} + \log\frac{\Pr(X_1, X_2, \ldots, X_n \mid Y=0)}{\Pr(X_1, X_2, \ldots, X_n \mid Y=1)} \tag{3.45}$$

As we assumed $\Pr(Y = 0) = \Pr(Y = 1) = \frac{1}{2}$, the first term of Λ_L will be cancelled.

Suppose that $(X_1, X_2, \ldots, X_n)$ has k ones and $(n-k)$ zeros,

$$\Pr(X_1, X_2, \ldots, X_n \mid Y=0) = p_e^k (1-p_e)^{n-k} \tag{3.46}$$

$$\Pr(X_1, X_2, \ldots, X_n \mid Y=1) = p_e^{n-k} (1-p_e)^k \tag{3.47}$$

$$\log\frac{\Pr(X_1, X_2, \ldots, X_n \mid Y=0)}{\Pr(X_1, X_2, \ldots, X_n \mid Y=1)} = \log\frac{p_e^k (1-p_e)^{n-k}}{p_e^{n-k} (1-p_e)^k} = (n-2k)\log\left(\frac{1-p_e}{p_e}\right) \tag{3.48}$$

Since $0 < p_e < \frac{1}{2}$, $\log\left(\frac{1-p_e}{p_e}\right)$ is always positive. Therefore, if $k < \frac{n}{2}$, $\Lambda_L > 0$ and Y is estimated to be zero and if $k > \frac{n}{2}$, $\Lambda_L < 0$ and Y is estimated to be one.

In summary, Maximum Likelihood estimation in this problem, results in 'Majority Decision'. That is, since sources are unbiased and probability of $Y = 0$ is equal to $Y = 1$, we look at the set of observations and count the number of zeros and ones, whichever is higher, we consider as Y. For general cases (biased sources) we should calculate the likelihood ratio as in (3.45) and find the estimation rule.

3.4 Coding Structure

Our analysis proved that the following coding scheme can be applied to this problem. We determine the number of required observers (n) and follow the algorithm:

a) Node 1 encodes its data into $H(X_1)$ bits and sends its message to the receiver.

b) Node 2 encodes its data into $H(X_2 \mid X_1)$ bits and sends its message to the receiver.

c) Similarly, nodes $i = 3,4,\ldots, n$ encode their data into $H(X_i \mid X_{i-1},\ldots,X_2,X_1)$ bits and send their message to the receiver.

The receiver sequentially decodes X_1, $X_2,\ldots$, X_n using the previously decoded data. That is, the joint decoder at the receiver, uses $X_1, X_2,\ldots, X_{i-1}$ and the message received from node i, which was encoded to $H(X_i \mid X_{i-1},\ldots,X_2,X_1)$ bits, in order to decode X_i. After all data is detected at the receiver, it applies the Maximum Likelihood estimation rule to estimate Y. The decoding structure at the receiver is illustrated in Fig. 3.2.

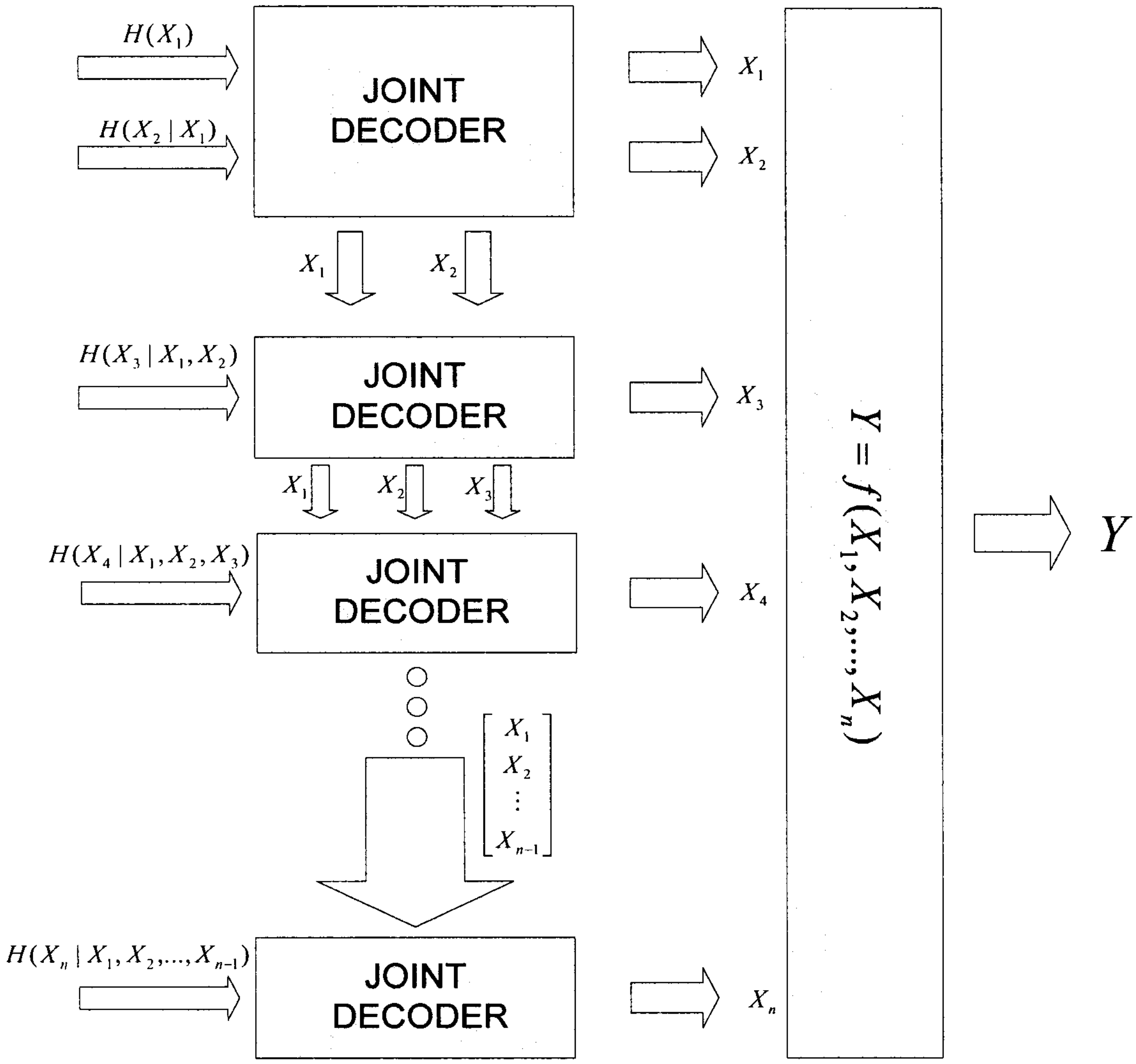

Figure 3.2 – Receiver structure for joint decoding

3.4 Supplementary Analysis

We can also model this problem as encoding a data sequence using repetition code, passing it through a Binary Symmetric Channel (BSC) with cross-over probability p_e and decoding the noisy sequence. The block diagram of this model is shown in Fig.3.3

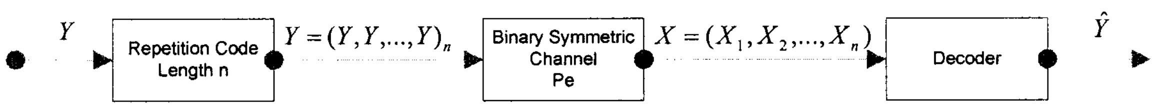

Figure 3.3 – Repetition code model

As $\underline{X} = (X_1, X_2, ..., X_n)$ is a function of Y and the final estimation $\hat{Y}$ is a function of $\underline{X}$, we have $Y \to \underline{X} = (X_1, X_2, ..., X_n) \to \hat{Y}$ forming a Markov chain. Therefore,

$$H(Y \mid \underline{X}) \leq H(Y \mid \hat{Y}) \tag{3.49}$$

The uncertainty about Y given $\hat{Y}$, depends on the probability of decoding error:

$$\Pr\{error\} = \Pr\{Y \neq \hat{Y}\} \tag{3.50}$$

As derived in Section 3.3, we will have an estimation error whenever the weight of the error vector $\underline{E} = (E_1, E_2, ..., E_n)$ is greater than $\frac{n}{2}$. Hence,

$$\Pr\{Y \neq \hat{Y}\} = \Pr\left\{weight(error) \geq \frac{n}{2}\right\} = \sum_{i=\frac{n}{2}}^{n} \binom{n}{i} p_e^i (1-p_e)^{n-i} \cong Q\left(\frac{\frac{n}{2} - np_e}{\sqrt{np_e(1-p_e)}}\right)$$

$$= Q\left(\sqrt{n}\frac{\frac{1}{2} - p_e}{\sqrt{p_e(1-p_e)}}\right) \tag{3.51}$$

The approximation in (3.51) holds for large values of n. As n tends to infinity, we have

$$\lim_{n\to\infty} \Pr\{Y \neq \hat{Y}\} = \lim_{n\to\infty} Q\left(\sqrt{n}\frac{\frac{1}{2} - p_e}{\sqrt{p_e(1-p_e)}}\right) = 0 \tag{3.52}$$

Therefore,

$$\lim_{n\to\infty} H(Y \mid \hat{Y}) = 0 \tag{3.53}$$

and from (3.49) we conclude that

$$\lim_{n\to\infty} H(Y \mid \underline{X}) = 0 \tag{3.54}$$

CHAPTER FOUR

COOPERATIVE SENSOR NETWORKS: COMPRESSION AND TRANSMISSION

In Wireless Sensor Networks (WSNs) nodes operate on batteries. Therefore, they have limited energy. One of the main objectives in designing WSNs is to reduce energy consumption as much as possible, so that network lifetime increases. As explained earlier in Chapter 2, we can apply cooperative communication techniques for compression and transmission of information. We assume that nodes are able to communicate with each other and base our transmission on Virtual (Cooperative) MIMO technique. Energy efficiency of Virtual MIMO has already been studied in [28], [30] and [31]. Considering their results, in this chapter we propose techniques for processing and compressing the information before transmission with the aim of saving more energy. In order to become a Virtual MIMO system, nodes confer with each other. Since sensors' data are correlated, they can apply distributed source coding techniques to send their information to their neighbors and as we have their information available in multiple nodes, this information can be compressed using a joint encoder. Two classes of sensor networks are studied. First, we consider CEO problem in wireless sensor networks where CEO's objective is to estimate one phenomenon $\theta = f(X_1, X_2, ..., X_L)$ from a set of observations and after that, we consider sensor networks in the context of Multi-Terminal communication problem where the center is interested in all observations and networks' objective is to have individual observations $\underline{X} = (X_1, X_2, ..., X_L)$ at the Center.

4.1 CEO Problem in Wireless Sensor Networks

4.1.1 Assumptions

A) System Model

The network consists of L distributed sensor nodes and a Fusion Center (also called CEO). Sensors are deployed uniformly in the field, close to one another, each taking observations on an unknown parameter θ. The CEO is located far from the nodes. All nodes observe same phenomenon but with noisy observations. These nodes together with the Fusion Center are supposed to estimate the value of the unknown parameter. Nodes send binary messages to the Fusion Center. FC processes the received messages and estimates the phenomenon.

B) Data Model

Consider the Quadratic Gaussian CEO problem [6] in wireless sensor networks. The phenomenon to be estimated has Gaussian distribution $\theta \sim N(0, \sigma_x^2)$ and the observations are corrupted by additive noises $n_i \sim N(0, \sigma_n^2)$. That is,

$$X_i = \theta + n_i \tag{4.1}$$

$$X_i \sim N(0, \sigma_x^2 + \sigma_n^2) \tag{4.2}$$

We estimate θ as $\hat{\theta}$ where,

$$\hat{\theta} = \frac{1}{L}\sum_{i=1}^{L} X_i \,. \tag{4.3}$$

Note that since additive noises are zero-mean, we will have $E[\hat{\theta}] = E[\theta]$ and the estimation is unbiased.

C) Reference System Model

Our reference system consists of L conventional Single-Input Single-Output (SISO) wireless links, each connecting one of the sensor nodes to the FC. For the reference system we do not consider any communication or cooperation among the sensors. Therefore each sensor quantizes its observation by an l–bit scalar quantizer designed for distribution of θ, generates a message of length l and transmits it directly to the FC. Fusion Center receives all messages and performs the processing, which is calculation of the numerical average of these messages.

4.1.2 Cooperative Data Processing Algorithm

Sensor observations are analog quantities. Therefore, each sensor has to quantize (compress) its data before transmission. For data compression we use l–bit scalar quantizer [37],[38]. In our algorithm, network is divided into *clusters*, each cluster having a fixed number of members m. Members of each cluster are supposed to cooperate with one another in two ways:

1. Share, Process and Compress their data
2. Cooperatively transmit their processed data using virtual MIMO

Figure 4.1 illustrates the cooperative data processing block diagram for CEO problem in Wireless Sensor Networks.

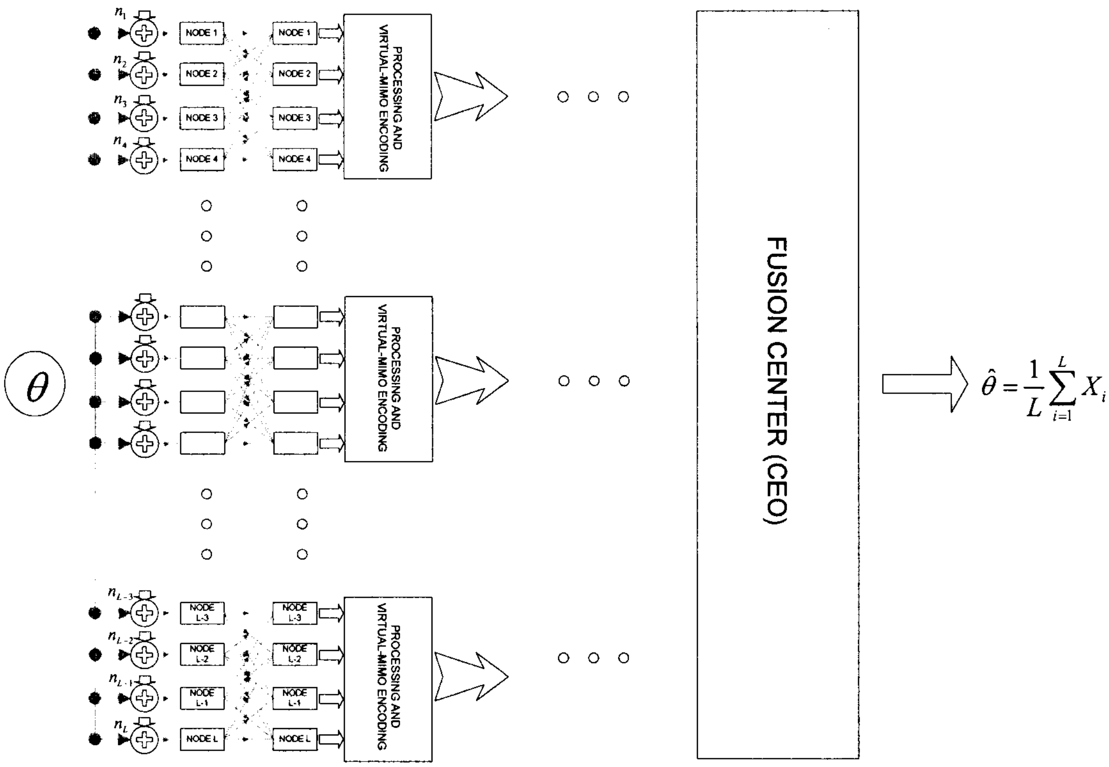

Figure 4.1 – CEO problem in Wireless Sensor Networks

A) Phase 1: Cooperative Processing and Compression

In the first phase, nodes inside each cluster confer with one another and send their data to their neighbors so that all members of each cluster will be aware of observations of their co-cluster neighbors and will have the same set of data as others. We can model the channel between nodes inside a cluster as AWGN with κ^{th}–order path loss ($\kappa = 3.5$). This part of transmission can be done with very small probability of error (error-free).

At this time, each node decodes the ' m ' received messages from its neighbors, takes their numerical average and again quantizes the sub-average into l bits. As they all have the same set of data, they will reach the same result. This message is the information of the corresponding cluster. As each sensor has an l–bit message, in this phase by computing the average, we have compressed ml bits into l bits. That is, the compression ratio is m.

The first phase of our algorithm not only compresses the message length by factor of m, but also lets us consider each cluster as a virtual Multiple-Input Multiple-Output (MIMO) system with m antennas.

B) Phase 2: Cooperative Transmission by Virtual-MIMO

In the second phase, sensors split this message into m symbols and transmit it using an orthogonal space-time block code (STBC) [35] of order m. As orthogonal STBC codes have lower Bit Error Rate (BER) compared to SISO, we can meet the required BER or distortion by consuming less energy than a SISO system.

Depending on the size of clusters, we can use different STBCs. For $m = 2$, we can use Alamouti code [34], which is full rate. For $m = 3$ and $m = 4$, we have an orthogonal code of rate $3/4$ and for $m \geq 4$ we can construct an orthogonal code of rate $1/2$ [35].

Processing and cooperation block diagram of members in each cluster is shown in Fig. 4.2.

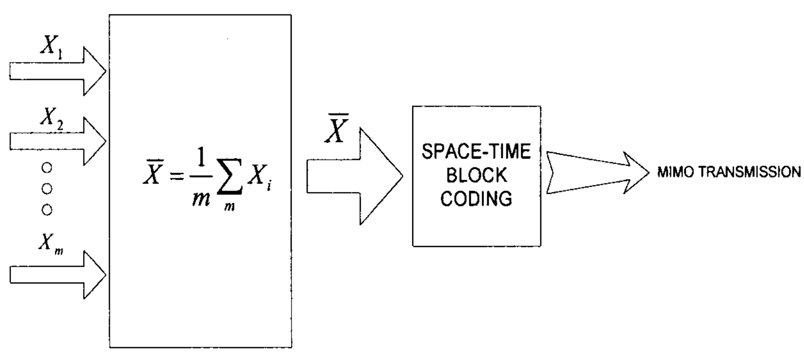

Figure 4.2 – Block diagram of Cooperation (Processing and Virtual-MIMO)

We consider the channel connecting clusters to the FC as a *Rayleigh* fading channel with square-law path loss ($\kappa = 2$). Once all clusters have transmitted their data to the FC, the Fusion Center decodes these binary packets into analog numbers and takes the numerical average of these messages and considers it as the network's measurement.

4.1.3 Analysis

The performance metric considered in our analysis is the total distortion due to compression and errors occurred during transmission. The first distortion is due to finite length quantizer, used in each sensor to represent the analog number by l bits. This distortion depends on the design of quantizer. We consider a Gaussian scalar quantizer which is designed over 10^6 randomly generated samples. The second distortion is due to errors occurred during transmission through the channel. In our system, this distortion is proportional to the probability of bit error. Since the probability of bit error p_e is a function of transmission energy per bit E_b (assuming fixed power for noise), the transmission distortion will be a function of E_b. In this section we characterize the transmission and total consumed energy of sensors and find the relationship between distortion and probability of bit error. Finally, we sketch the Energy–Distortion diagrams of V-MIMO and SISO.

A) Estimation Procedure

Estimation of the unknown parameter (calculation of average) is done in two stages. The first stage is done among m sensors inside each cluster and the second level is done by taking the average of the results of the first level. We denote the data belonging to sensor i and cluster j by $X_{i,j}$. The outcome of each cluster by $y_j, (j = 1,2,..., \frac{L}{m})$ and the final estimation by $\hat{\theta}$.

$$y_j = \frac{1}{m}\sum_{i=1}^{m} X_{i,j} \quad , \quad j = 1,2,\ldots,\frac{L}{m} \tag{4.4}$$

$$\hat{\theta} = \frac{m}{L}\sum_{j=1}^{\frac{L}{m}} y_j = \frac{m}{L}\sum_{j=i}^{\frac{L}{m}} \frac{1}{m}\sum_{i=1}^{m} X_{i,j} = \frac{1}{L}\sum_{j=1}^{\frac{L}{m}}\sum_{i=1}^{m} X_{i,j} = \frac{1}{L}\sum_{all\ data} X_{i,j} \tag{4.5}$$

B) Energy Consumption

In order to compare the proposed algorithm with the reference conventional system (SISO), we have to consider total energy consumption of sensors which is the energy consumed in baseband and RF circuitry plus the energy consumed in power amplifiers in order to satisfy certain SNR at the receiver. In addition, we have to make sure that the total delay (transmission rate) of the system is the same for both schemes.

The difference in total energy consumption is due to the fact that in the proposed V-MIMO scheme, each sensor in the "*Cooperation Phase*" is involved in one transmission and $(m - 1)$ receptions, so that if we assume that each transceiver circuit consumes P_{ct} (Watts) during transmission and P_{cr} (Watts) during reception, the overhead of "Cooperation Phase" for each sensor will be

$$E_{Ph1} = P_{ct} \times T_{on} + (m-1)P_{cr} \times T_{on} \tag{4.6}$$

Where T_{on} is the time that each sensor spends to transmit l bits to its neighbors.

The communication system model for transmission and reception is the same as [28], [30] and [31]. That is,

$$P_{ct} = P_{DAC} + P_{filt} + P_{mix} + P_{synth} \tag{4.7}$$

$$P_{cr} = P_{ADC} + P_{filr} + P_{mix} + P_{synth} + P_{LNA} + P_{IFA} \tag{4.8}$$

Where P_{DAC}, P_{ADC}, P_{filr}, P_{filt}, P_{mix}, P_{synth}, P_{LNA} and P_{IFA} are the power consumption values of D/A converter, A/D converter, active filters of receiver, active filters of transmitter, mixer, frequency synthesizer, Low Noise Amplifier and Intermediate Frequency Amplifier, respectively. The received signal power (P_r) is modeled as

$$P_r = P_t\left(\frac{\lambda}{4\pi d}\right)^2 \frac{G_t G_r}{M_l N_f} \tag{4.9}$$

Where P_t is the power of transmitting signal and G_t and G_r are the transmitter and receiver antenna gains, respectively. M_l is the link margin compensating the hardware process variations and $N_f = \frac{N_r}{N_o}$ is the receiver noise figure.

C) Distortion

In both systems sensors have l–bit messages that have to be transmitted through the channel and decoded by the Fusion Center. The scalar quantizer has 2^l levels and as a result, we can divide the Real axis into 2^l sub-intervals (W_i). We apply Gray coding to map the Real axis into l-bit codewords and denote the quantizing points of each sub-interval as U_i where i is the decimal value of the corresponding codeword of that sub-interval ($i = 0,1,\ldots,2^l - 1$). With these assumptions in mind, occurrence of an error in the j^{th} Least Significant Bit (LSB) will result in decoding $U_{(i+2^{j-1}) \bmod 2^l}$ instead of U_i. As the probability of bit error is very small, we assume that in each packet, only one bit might be in error. Therefore, the resulting distortion will be

$$D = \sum_{j=1}^{l} \sum_{i=1}^{2^l} P_j \int_{W_i} \left(x - U_{(i+2^{j-1}) \bmod 2^l}\right)^2 f_X(x) dx \tag{4.10}$$

Where P_j is the probability that the j^{th} LSB is in error and

$$f_X(x) = \frac{1}{\sqrt{2\pi(\sigma_x^2 + \sigma_n^2)}} e^{-\frac{x^2}{2(\sigma_x^2 + \sigma_n^2)}} \tag{4.11}$$

is the probability density function of X_i. Since bits of each packet are equally protected, we have $P_j = p_e$ and P_j can come out of the summations, we will have $D = \alpha.p_e$ where

$$\alpha = \sum_{i=1}^{2^l}\sum_{j=1}^{l}\int_{W_i}\left(x - U_{(i+2^{j-1})\bmod 2^l}\right)^2 f_X(x)dx \tag{4.12}$$

The above distortion D corresponds to one sensor. Therefore, the total distortion after estimating the average at the FC will be

SISO: $$D_{Total} = \frac{D}{L} = \frac{\alpha}{L} P_{e,SISO} \tag{4.13}$$

V-MIMO: $$D_{Total} = \frac{m}{L} D = \frac{\alpha.m}{L} P_{e,MIMO} \tag{4.14}$$

4.1.4 Simulation and Numerical Results

To give a numerical example, we assume $m = 4$ members in each cluster. Therefore, our Virtual-MIMO scheme will consist of 4 transmit antennas. We assume that network has $L = 32$ nodes. Sensor observations are Gaussian with $\sigma_x^2 = 1$ and are corrupted by a Gaussian noise of $\sigma_n^2 = 0.1$. Nodes are deployed uniformly in the field and are 2 meters apart from each other and the Fusion Center is located 100 meters away from the center of the field.

The values for circuit parameters are quoted from [28] and are listed in Table. 4.1. These parameters depend on the hardware design and technological advances. Hence, if we have more efficient hardware, our algorithm provides better Energy–Distortion performance. For the MIMO transmission, we applied orthogonal space time block code

[35] for 4 transmit antennas with rate $\frac{1}{2}$. The coding Matrix used for simulations is shown below:

$$\begin{bmatrix} s_1 & -s_2 & -s_3 & -s_4 & s_1^* & -s_2^* & -s_3^* & -s_4^* \\ s_2 & s_1 & s_4 & -s_3 & s_2^* & s_1^* & s_4^* & -s_3^* \\ s_3 & -s_4 & s_1 & s_2 & s_3^* & -s_4^* & s_1^* & s_2^* \\ s_4 & s_3 & -s_2 & s_1 & s_4^* & s_3^* & -s_2^* & s_1^* \end{bmatrix} \quad (4.15)$$

P_{DAC} = 15.7 mW	P_{ADC} = 6.7 mW
P_{filr} = P_{filt} = 2.5 mW	f_c = 2.5 GHz
P_{mix} = 30.3 mW	P_{syn} = 50 mW
P_{LNA} = 20 mW	P_{IFA} = 3 mW
$G_t G_r$ = 5 dBi	N_0 = – 171 dBm/Hz
M_l = 40 dB	N_f = 10 dB

Table 4.1 – System Parameters

The Bit Error Rate performance of the proposed algorithm (V-MIMO) is compared with reference system (SISO) in Fig. 4.3. The significant difference in performance motivated us to design our algorithm based on MIMO.

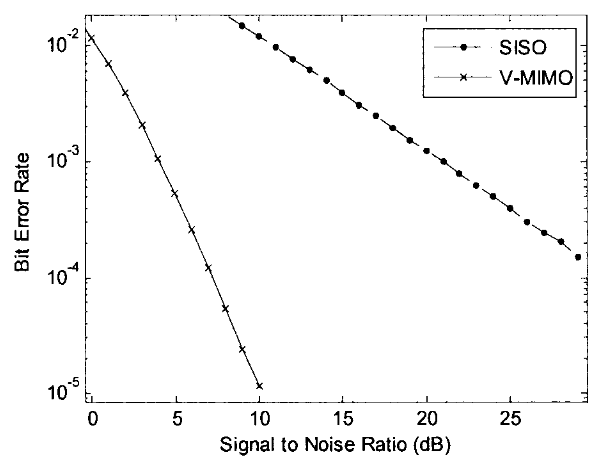

Figure 4.3 – Bit Error Rate vs. Signal to Noise Ratio at the Fusion

In Fig. 4.4 we sketched and compared the Distortion performance versus Transmission energy consumption per bit of the reference system and the proposed Two-Phase V-MIMO scheme. Fig. 4.4(a) shows the curves in linear scale. In order to have better view of the performance, we depicted our curves also on the Logarithmic scale over a wider range in Fig. 4.4(b). As shown in the figures, depending on how much precision needed in the system, we can save energy by using the proposed algorithm.

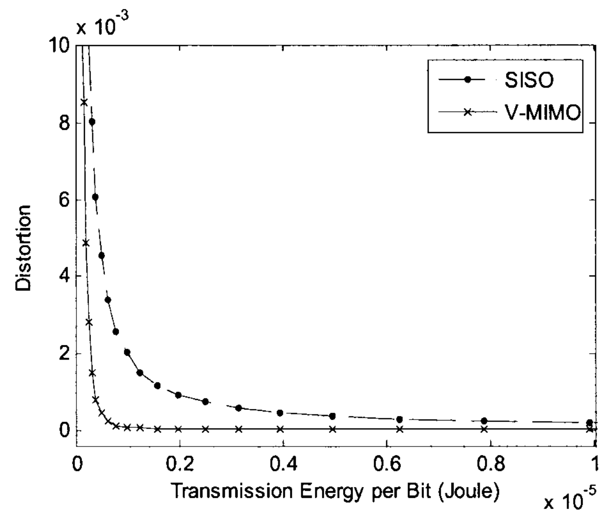

Figure 4.4(a) – Distortion vs. Transmission Energy Consumption per Bit (Linear Scale)

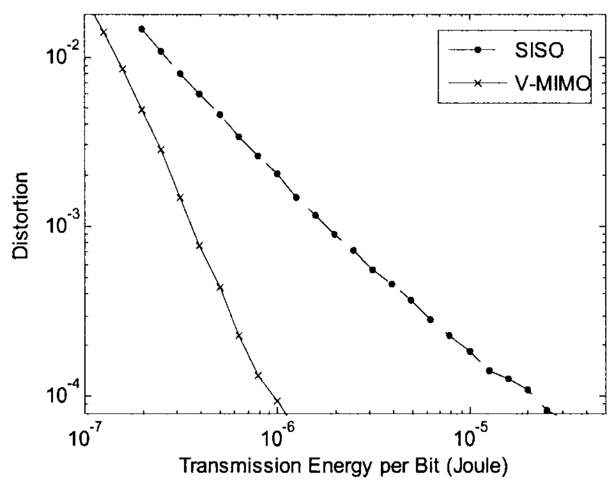

Figure 4.4(b) – Distortion vs. Transmission Energy Consumption per Bit (Logarithmic Scale)

Figures 4.5 illustrate the Distortion versus total energy consumption of sensor nodes. We again depicted the performance over linear and logarithmic scales to have a better and complete view. The parameters that lead us to these results are quoted from [28]. Note that we may design different circuits to achieve better performance. However, from these figures we can conclude that the proposed algorithm outperforms the reference system when we want to have distortion less than 10^{-3} and it can save energy as high as 10 dB.

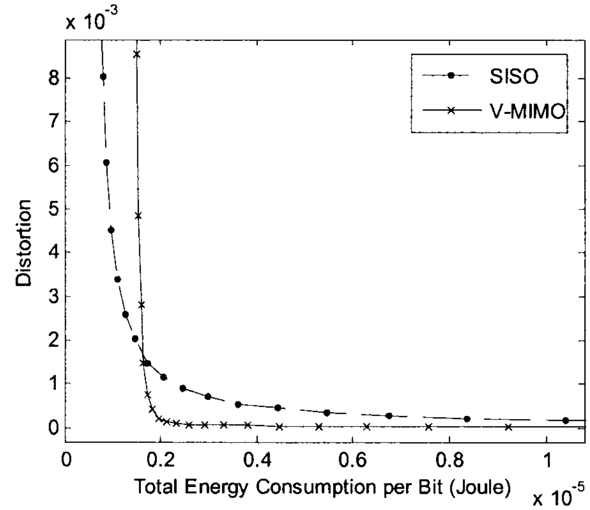

Figure 4.5(a) – Distortion vs. Total Energy Consumption per Bit (Linear Scale)

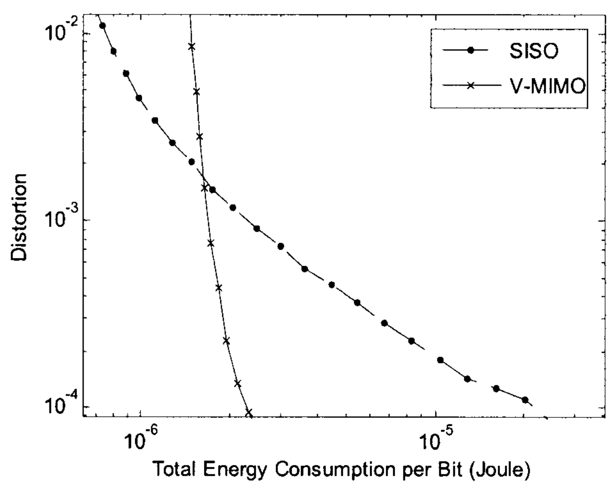

Figure 4.5(b) – Distortion vs. Total Energy Consumption per Bit (Logarithmic Scale)

In summary, we proposed a novel algorithm that takes advantage of cooperation among sensor nodes in two ways: it not only compresses the set of sensor messages at the sensor nodes into one message appropriate for final estimation, but also encodes this common message into orthogonal space-time symbols which are easy to decode and energy-efficient. This algorithm is able to save energy as high as 10 dB.

4.2 Multi-Terminal Communications Problem in Wireless Sensor Networks: Source Coding and Transmission

In this section we extend the CEO problem in Wireless Sensor Networks to a more general model. In some applications, it is required to have all nodes' observations available at the Fusion Center. We assume that nodes' data are highly correlated and we are interested in transmitting observations of all nodes, denoted by $\underline{X} = (X_1, X_2, ..., X_L)$. Therefore, we propose algorithms that jointly compress X_i's before transmission. The block diagram for our system is shown in Fig. 4.6.

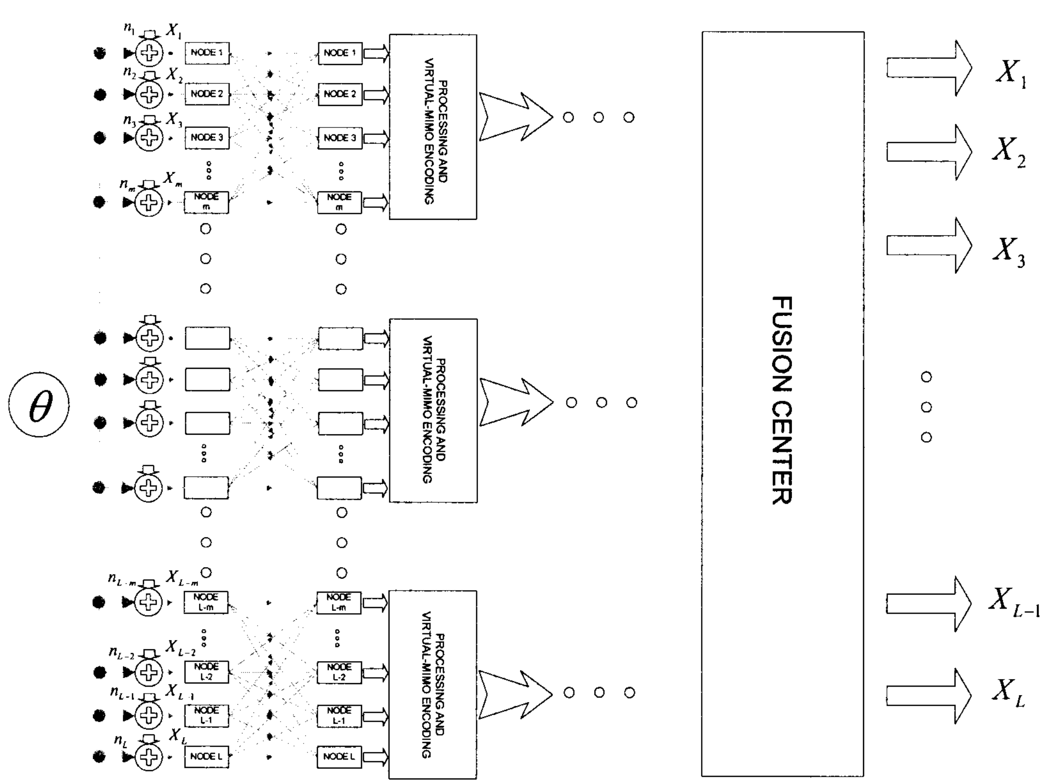

Figure 4.6 – Block diagram of multi-terminal communication problem in wireless sensor networks

We consider a data model similar to the one considered in Section 4.1. Data are Gaussian distributed. Nodes' observations are highly correlated. We denote the correlated part of data by θ and the differences between observations by n_i. Therefore, observations are again additions of two Gaussian random variables. The common part θ has higher variance compared to n_i's. We are interested in transmitting X_i's instead of θ. The number of sensors is assumed to be L.

$$X_i = \theta + n_i \quad , \quad i = 1,2,\ldots,L \tag{4.16}$$

$$\theta \sim N(0,\sigma_x^2) \ , \ n_i \sim N(0,\sigma_n^2) \tag{4.17}$$

$$X_i \sim N(0,\sigma_x^2 + \sigma_n^2) \quad , \quad \sigma_x^2 >> \sigma_n^2 \tag{4.18}$$

The amount of correlation among sensor's data depends on the ratio of $\frac{\sigma_x^2}{\sigma_n^2}$. The higher this ratio, the more correlation among sensors' data. We define this ratio by a new parameter in order to characterize the correlation.

$$K = \frac{\sigma_x^2}{\sigma_n^2} \tag{4.19}$$

Correlation coefficient for two sensors' data is

$$\rho_{X_i X_j} = \frac{\sigma_{X_i X_j}}{\sigma_{X_i}\sigma_{X_j}} = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_n^2} = \frac{K}{K+1} \tag{4.20}$$

4.2.1 Compression via Transforming into Parallel Gaussian Sources

The objective is to transmit the vector $\underline{X} = (X_1, X_2, \ldots, X_L)$ using Virtual MIMO. As described in the previous section, we divide the network into clusters of 'm' members. In order to be able to resemble a Virtual MIMO, members of each cluster should have the same data. They confer with each other (communicate X_i's to their neighbors) and process the set of data, encode it and transmit the new data using Virtual MIMO.

We calculate the cross-correlation of sensors' data and build up the covariance matrix.

$$E[X_iX_j] = E[(\theta + n_i)(\theta + n_j)] = E[\theta^2] + E[\theta.n_i] + E[\theta.n_j] + E[n_in_j]$$

$$= \begin{cases} \sigma_x^2 + \sigma_n^2 & , \quad i = j \\ \sigma_x^2 & , \quad i \neq j \end{cases} \tag{4.21}$$

The covariance matrix will be

$$C_{XX} = E[X_iX_j] = \begin{bmatrix} \sigma_x^2 + \sigma_n^2 & \sigma_x^2 & \cdots & \sigma_x^2 \\ \sigma_x^2 & \sigma_x^2 + \sigma_n^2 & \cdots & \sigma_x^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_x^2 & \sigma_x^2 & \cdots & \sigma_x^2 + \sigma_n^2 \end{bmatrix}_{m \times m} \tag{4.22}$$

As the covariance matrix is symmetric, we can decompose it using eigenvalue decomposition method.

$$C_{XX} = E[X_iX_j] = \begin{bmatrix} \sigma_x^2 + \sigma_n^2 & \sigma_x^2 & \cdots & \sigma_x^2 \\ \sigma_x^2 & \sigma_x^2 + \sigma_n^2 & \cdots & \sigma_x^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_x^2 & \sigma_x^2 & \cdots & \sigma_x^2 + \sigma_n^2 \end{bmatrix}_{m \times m} = U\Lambda U^T \tag{4.23}$$

where,

$$\Lambda = \begin{bmatrix} m\sigma_x^2 + \sigma_n^2 & 0 & \cdots & 0 \\ 0 & \sigma_n^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix}_{m \times m} \quad \text{and} \quad UU^T = I_m \tag{4.24}$$

The diagonal elements of Λ are the eigenvalues of C_{XX}. The corresponding orthogonal eigenvectors are as following:

$$V_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \\ \\ \vdots \\ 1 \end{bmatrix} \quad V_2 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad V_3 = \begin{bmatrix} 1 \\ 1 \\ -2 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \cdots \quad V_m = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ \\ 1 \\ -m \end{bmatrix} \tag{4.25}$$

The unitary matrix U is constructed by normalizing the eigenvectors (V_i's) and putting them in the columns of U. That is, $U_i = \frac{V_i}{\|V_i\|}$.

Basically, by transforming the data, we are removing the correlation among sources and they are turning to become m parallel independent Gaussian sources. According to Reverse Water-Filling solution [39], we can quantize these data with the following rates.

$$R_i = \begin{cases} \frac{1}{2}\log\frac{\lambda_i}{\left(\frac{D}{m}\right)} & , \quad \lambda_i > \frac{D}{m} \\ 0 & , \quad \lambda_i < \frac{D}{m} \end{cases} \tag{4.26}$$

Therefore,

$$R_{Total} = \frac{1}{2}\log\frac{m\sigma_x^2 + \sigma_n^2}{\left(\frac{D}{m}\right)} + (m-1)\frac{1}{2}\log\frac{\sigma_n^2}{\left(\frac{D}{m}\right)} = \frac{1}{2}\log\frac{m\sigma_x^2(\sigma_n^2)^{m-1} + (\sigma_n^2)^m}{\left(\frac{D}{m}\right)^m} \tag{4.27}$$

In terms of K

$$R_{Total} = \frac{1}{2}\log\frac{m\sigma_x^2(\sigma_n^2)^{m-1} + (\sigma_n^2)^m}{\left(\frac{D}{m}\right)^m} = \frac{1}{2}\log\frac{(mK+1)(\sigma_n^2)^m}{\left(\frac{D}{m}\right)^m} \tag{4.28}$$

The processing and compression block diagram in depicted in Fig. 4.7.

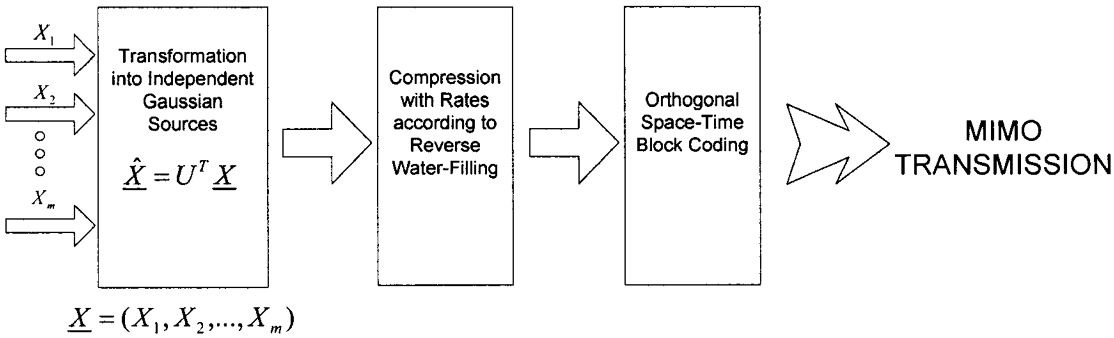

Figure 4.7 – Block diagram of processing and compression for cooperative source coding

Since $\sigma_x^2 >> \sigma_n^2$, variance of $\varepsilon_j = X_j - X_i$, which is $2\sigma_n^2$, is much less than variance of X_j, which is $\sigma_x^2 + \sigma_n^2$. For communication inside clusters during conference phase we can reduce the rate of transmission. We propose that in each cluster one of the nodes transmit X_i and the rest of the nodes transmit $\varepsilon_j = X_j - X_i$. Without loss of generality we assume that X_1 is transmitted completely and $X_2, X_3, \ldots, X_m$ subtract the received data (X_1) from themselves and encode the difference.

Transmission scenario scheme is depicted in Fig. 4.8.

$$X_1 = \theta + n_1 \xrightarrow{transmit}$$

$$X_2 = \theta + n_2 \qquad \varepsilon_2 = X_2 - X_1 \xrightarrow{transmit}$$

$$X_3 = \theta + n_3 \qquad \varepsilon_3 = X_3 - X_1 \xrightarrow{transmit}$$

$$\vdots \qquad \vdots$$

$$X_m = \theta + n_m \qquad \varepsilon_m = X_m - X_1 \xrightarrow{transmit}$$

Figure 4.8 – Transmission scenario scheme inside each cluster

All nodes will have $(X_1, \varepsilon_2, \varepsilon_3, \ldots, \varepsilon_m)$ where,

$$X_1 \sim N(0, \sigma_x^2 + \sigma_n^2) \tag{4.29}$$

$$\varepsilon_j \sim N(0, 2\sigma_n^2) \quad , \quad j = 2, 3, \ldots, m \tag{4.30}$$

The rate-distortion function for the new method is,

$$R_1 = \frac{1}{2}\log\frac{\sigma_x^2+\sigma_n^2}{D} \tag{4.31}$$

$$R_j = \frac{1}{2}\log\frac{2\sigma_n^2}{D} \quad , \quad for\ j = 2,3,\ldots,m \tag{4.32}$$

Leading to a sum-rate distortion region:

$$R_{Total} = \sum_{i=1}^{m} R_i = \frac{1}{2}\log\frac{\sigma_x^2+\sigma_n^2}{D} + (m-1)\frac{1}{2}\log\frac{2\sigma_n^2}{D} = \frac{1}{2}\log\frac{(K+1)2^{m-1}(\sigma_n^2)^m}{D^m} \tag{4.33}$$

Compared to the first method (all nodes transmitting X_i's during conference phase), the new algorithm saves $\frac{m-1}{2}\log\left(\frac{K+1}{2}\right)$ bits. Because,

$$m\times\frac{1}{2}\log\left(\frac{\sigma_x^2+\sigma_n^2}{D}\right) - R_{Total} = (m-1)\left[\frac{1}{2}\log\left(\frac{\sigma_x^2+\sigma_n^2}{D}\right) - \frac{1}{2}\log\left(\frac{2\sigma_n^2}{D}\right)\right] = \frac{m-1}{2}\log\left(\frac{K+1}{2}\right) \tag{4.34}$$

The transmission rate reduction diagram is shown in Fig. 4.9

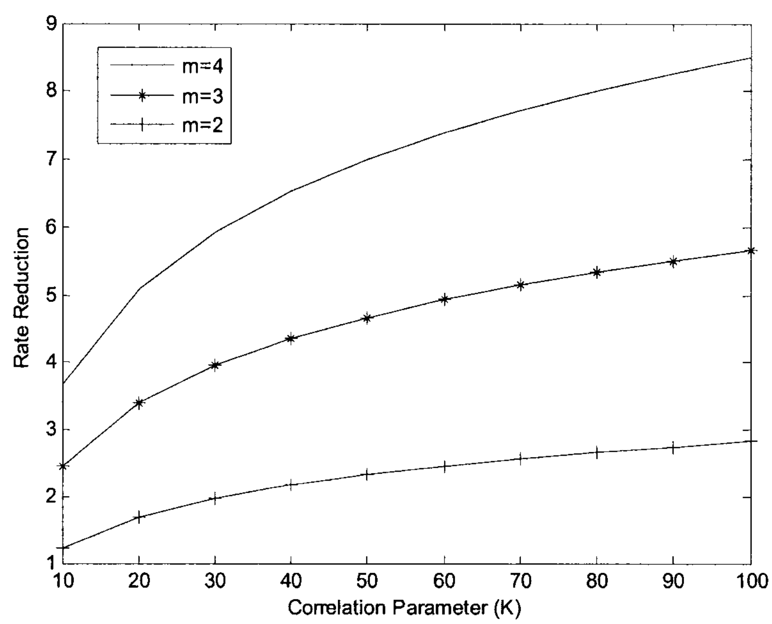

Figure 4.9 Transmission Rate Reduction vs. Correlation

We calculate the cross correlation among set of $(X_1, \varepsilon_2, \varepsilon_3, ..., \varepsilon_m)$

$$E[X_1\varepsilon_j] = E[(\theta + n_1)(n_j - n_1)] = E[\theta.n_j] + E[n_1.n_j] - E[\theta.n_1] - E[n_1^2] = -\sigma_n^2 \quad (4.35)$$

$$E[\varepsilon_i\varepsilon_j] = E[(n_i - n_1)(n_j - n_1)] = E[n_i.n_j] + E[n_1.n_i] - E[n_1.n_j] + E[n_1^2]$$

$$= \begin{cases} 2\sigma_n^2 & , \quad i = j \ (i \neq 1) \\ \sigma_n^2 & , \quad i \neq j \ (i \neq 1) \end{cases} \quad (4.36)$$

The covariance matrix of $(X_1, \varepsilon_2, \varepsilon_3, ..., \varepsilon_m)$ will be

$$C_{XX} = \begin{bmatrix} \sigma_x^2 + \sigma_n^2 & -\sigma_n^2 & -\sigma_n^2 & \cdots & -\sigma_n^2 \\ -\sigma_n^2 & 2\sigma_n^2 & \sigma_n^2 & \cdots & \sigma_n^2 \\ -\sigma_n^2 & \sigma_n^2 & 2\sigma_n^2 & & \sigma_n^2 \\ \vdots & \vdots & & \ddots & \vdots \\ -\sigma_n^2 & \sigma_n^2 & \sigma_n^2 & \cdots & 2\sigma_n^2 \end{bmatrix}_{m\times m} = U\Lambda U^T \quad (4.37)$$

where,

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_m \end{bmatrix}_{m\times m} \quad \text{and} \quad UU^T = U^TU = I \quad (4.38)$$

We calculate the eigenvalues $(\lambda_1, \lambda_2, ..., \lambda_m)$ of the above matrix.

$$\lambda_1 = \frac{\sigma_x^2 + (m+1)\sigma_n^2}{2} + \frac{\sqrt{(\sigma_x^2)^2 - 2(m-1)\sigma_x^2\sigma_n^2 + (m^2 + 2m - 3)(\sigma_n^2)^2}}{2}$$

$$= \frac{K + m + 1 + \sqrt{(K - m + 1)^2 + 4(m-1)}}{2}\sigma_n^2 \quad (4.39)$$

$$\lambda_2 = \frac{\sigma_x^2 + (m+1)\sigma_n^2}{2} - \frac{\sqrt{(\sigma_x^2)^2 - 2(m-1)\sigma_x^2\sigma_n^2 + (N^2 + 2m - 3)(\sigma_n^2)^2}}{2}$$

$$= \frac{K + m + 1 - \sqrt{(K - m + 1)^2 + 4(m-1)}}{2}\sigma_n^2 \quad (4.40)$$

$$\lambda_3 = \lambda_4 = \ldots = \lambda_m = \sigma_n^2 \tag{4.41}$$

If $K >> m$,

$$\lambda_1 \cong (K+1)\sigma_n^2 = \sigma_x^2 + \sigma_n^2 \quad \text{and} \quad \lambda_2 \cong m\sigma_n^2$$

The corresponding (orthogonal) eigenvectors are

$$V_1 = \begin{bmatrix} -\dfrac{\sigma_x^2 - (m-1)\sigma_n^2 + \sqrt{(\sigma_x^2)^2 - 2(m-1)\sigma_x^2.\sigma_n^2 + (m^2+2m-3)(\sigma_n^2)^2}}{2\sigma_n^2} \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} -\dfrac{K-m+1+\sqrt{(K-m+1)^2+4m+4}}{2} \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \cong \begin{bmatrix} -K+m-1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad , \quad \lambda_1 \tag{4.42}$$

$$V_2 = \begin{bmatrix} -\dfrac{\sigma_x^2 - (m-1)\sigma_n^2 - \sqrt{(\sigma_x^2)^2 - 2(m-1)\sigma_x^2.\sigma_n^2 + (m^2+2m-3)(\sigma_n^2)^2}}{2\sigma_n^2} \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} -\dfrac{K-m+1-\sqrt{(K-m+1)^2+4m+4}}{2} \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \cong \begin{bmatrix} 0 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad , \quad \lambda_2 \tag{4.43}$$

$$V_3 = \begin{bmatrix} 0 \\ 1 \\ -1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad V_4 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ -2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad V_5 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ -3 \\ \vdots \\ 0 \end{bmatrix} \quad \cdots \quad V_m = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ \vdots \\ -m+2 \end{bmatrix}, \ \lambda_3, \lambda_4, \ldots, \lambda_m \tag{4.44}$$

The unitary matrix U has normalized eigenvectors as its columns. That is, in order to form U, we have to define $U_i = \frac{V_i}{\|V_i\|}$ and put the normalized vectors in columns of U.

Now, if we transform the data vector $Y = (X_1, \varepsilon_2, \varepsilon_3, \ldots, \varepsilon_m)^T$ by matrix U^T, the new set of data will be uncorrelated and we have

$$\hat{Y} = U^T Y \tag{4.45}$$

$$E[\hat{Y}\hat{Y}^T] = E[U^T Y.Y^T U] = E[U^T (U \Lambda U^T) U] = E[I_m \Lambda I_m] = \Lambda \ (diagonal) \tag{4.46}$$

At the receiver we again transform the data by multiplying matrix U to the received vector. We then recover $Y = (X_1, \varepsilon_2, \varepsilon_3, \ldots, \varepsilon_m)^T$ and finally $\underline{X} = (X_1, X_2, \ldots, X_L)$.

According to Water-Filling solution we have

$$R_{Total} = \frac{1}{2}\log\frac{\lambda_1}{\left(\frac{D}{m}\right)} + \frac{1}{2}\log\frac{\lambda_2}{\left(\frac{D}{m}\right)} + (m-2)\frac{1}{2}\log\frac{\lambda_m}{\left(\frac{D}{m}\right)} = \frac{1}{2}\log\frac{\lambda_1\lambda_2}{\left(\frac{D}{m}\right)^2} + \frac{1}{2}\log\frac{\lambda_m^{\ m-2}}{\left(\frac{D}{m}\right)^{m-2}}$$

$$= \frac{1}{2}\log\frac{m\sigma_x^2.\sigma_n^2 + (\sigma_n^2)^2}{\left(\frac{D}{m}\right)^2} + \frac{1}{2}\log\frac{(\sigma_n^2)^{m-2}}{\left(\frac{D}{m}\right)^{m-2}} = \frac{1}{2}\log\frac{m\sigma_x^2.(\sigma_n^2)^{m-1} + (\sigma_n^2)^m}{\left(\frac{D}{m}\right)^m} \tag{4.47}$$

The processing block diagram of the improved algorithm is depicted in Fig. 4.10.

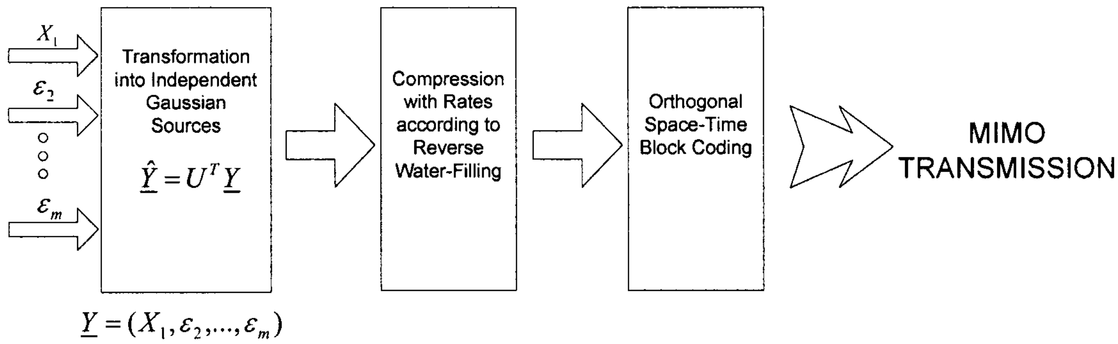

Figure 4.10 – Improved algorithm, processing and compression block diagram

4.2.2 Compression via Vector Quantization

As an alternative approach, we suggest using a Vector Quantizer for joint compression of correlated sources. That is, we consider that when nodes have shared their data inside their clusters and all have received the same set of data, they can use a Vector Quantizer to compress and encode the data jointly. Since they will reach to the same codeword, they can transmit this codeword by MIMO techniques. We compare the two systems (transformation into parallel independent sources & Vector Quantization) in performance by simulation. The block diagram of Vector Quantization based algorithm is depicted in Fig. 4.11.

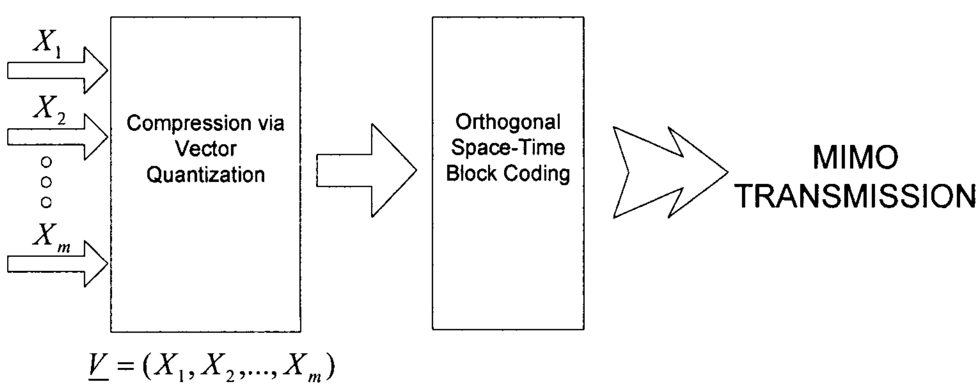

Figure 4.11 – Compression via Vector Quantization

We design a Vector Quantizer by applying the algorithm described in Chapter 2, on 10^6 randomly generated samples.

4.2.3 Numerical Examples and Simulations

In this section we present some examples of compression using the methods described above. We designed several Scalar and Vector Quantizers for the corresponding rates and variances.

We assume that nodes in the network are clustered in groups of $m = 4$ and are observing correlated sources with parameters $\sigma_x^2 = 1$, $\sigma_n^2 = 0.01$ (That is $K = \frac{\sigma_x^2}{\sigma_n^2} = 100$)

The pre-calculated covariance matrices for compression methods are:

Method 1:

$$C_{XX} = E[X_i X_j] = \begin{bmatrix} \sigma_x^2 + \sigma_n^2 & \sigma_x^2 & \cdots & \sigma_x^2 \\ \sigma_x^2 & \sigma_x^2 + \sigma_n^2 & \cdots & \sigma_x^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_x^2 & \sigma_x^2 & \cdots & \sigma_x^2 + \sigma_n^2 \end{bmatrix}_{m \times m} = U \Lambda U^T \tag{4.48}$$

where,

$$\Lambda = \begin{bmatrix} m\sigma_x^2 + \sigma_n^2 & 0 & \cdots & 0 \\ 0 & \sigma_n^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix}_{m \times m} \quad \text{and} \quad UU^T = I_m \tag{4.49}$$

Substituting for parameters, we have

$$C_{XX} = \begin{bmatrix} 1.01 & 1 & 1 & 1 \\ 1 & 1.01 & 1 & 1 \\ 1 & 1 & 1.01 & 1 \\ 1 & 1 & 1 & 1.01 \end{bmatrix} \tag{4.50}$$

$$
= \begin{bmatrix} \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & 0 & -\frac{2}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & 0 & 0 & -\frac{3}{2\sqrt{3}} \end{bmatrix} \times \begin{bmatrix} 4.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \times \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{6}} & 0 \\ \frac{1}{2\sqrt{3}} & \frac{1}{2\sqrt{3}} & \frac{1}{2\sqrt{3}} & -\frac{3}{2\sqrt{3}} \end{bmatrix}
$$

(4.51)

The total rate for compression of sources as derived in (4.27) and (4.47) is

$$
R = \frac{1}{2}\log\frac{m\sigma_x^2.(\sigma_n^2)^{m-1} + (\sigma_n^2)^m}{\left(\frac{D}{m}\right)^m} = \frac{1}{2}\log\frac{(mK+1)(\sigma_n^2)^m}{\left(\frac{D}{m}\right)^m} \quad (4.52)
$$

We set this rate equal to 8 and 12 and find the corresponding compression rate for each source, the rounded rates for compression are

$$
R_{Total} = 8 \quad \rightarrow \quad R_1 = 5,\ R_2 = 1,\ R_3 = 1,\ R_4 = 1 \quad (4.53)
$$

$$
R_{Total} = 12 \quad \rightarrow \quad R_1 = 6,\ R_2 = 2,\ R_3 = 2,\ R_4 = 2 \quad (4.54)
$$

Note that the quantizers with the above compression rates are designed for Gaussian sources with variances $\lambda_1 = 4.01, \lambda_2 = 0.01, \lambda_3 = 0.01, \lambda_4 = 0.01$.

Sample data:

$$
X_1 = 1.3098 \quad X_2 = 1.1872 \quad X_3 = 1.2236 \quad X_4 = 1.2084 \qquad \underline{X} = [1.3098, 1.1872, 1.2236, 1.2084]^T \quad (4.55)
$$

$$\underline{\hat{X}} = U^T \underline{X} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{6}} & 0 \\ \frac{1}{2\sqrt{3}} & \frac{1}{2\sqrt{3}} & \frac{1}{2\sqrt{3}} & -\frac{3}{2\sqrt{3}} \end{bmatrix} \times \begin{bmatrix} 1.3098 \\ 1.1872 \\ 1.2236 \\ 1.2084 \end{bmatrix} = \begin{bmatrix} 2.4645 \\ 0.0867 \\ 0.0203 \\ 0.0276 \end{bmatrix} \tag{4.56}$$

$\underline{\hat{X}}$ will be quantized to

$$R_{Total} = 8 \quad , \quad \hat{X}_q = \begin{bmatrix} 2.6073 \\ 0.0785 \\ 0.0785 \\ 0.0785 \end{bmatrix} \tag{4.57}$$

and the transformed data will be

$$\widetilde{X} = U\hat{X}_q = \begin{bmatrix} \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & 0 & -\frac{2}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & 0 & 0 & -\frac{3}{2\sqrt{3}} \end{bmatrix} \begin{bmatrix} 2.6073 \\ 0.0785 \\ 0.0785 \\ 0.0785 \end{bmatrix} = \begin{bmatrix} 1.4138 \\ 1.3028 \\ 1.2622 \\ 1.2357 \end{bmatrix} \tag{4.58}$$

The distortion for this example is 0.0066

We simulated the algorithm for 10^6 sample vectors and achieved an average distortion of

$$Average\ Distortion = 0.042$$

With higher quantization rate, $R_{Total} = 12$ we will have

$$R_{Total} = 12 \quad , \quad \hat{X}_q = \begin{bmatrix} 2.4554 \\ 0.0393 \\ 0.0393 \\ 0.0393 \end{bmatrix} \tag{4.59}$$

and the transformed data will be

$$\tilde{X} = U\hat{X}_q = \begin{bmatrix} \frac{1}{2} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & 0 & -\frac{2}{\sqrt{6}} & \frac{1}{2\sqrt{3}} \\ \frac{1}{2} & 0 & 0 & -\frac{3}{2\sqrt{3}} \end{bmatrix} \begin{bmatrix} 2.4554 \\ 0.0393 \\ 0.0393 \\ 0.0393 \end{bmatrix} = \begin{bmatrix} 1.2829 \\ 1.2273 \\ 1.2069 \\ 1.1936 \end{bmatrix} \tag{4.60}$$

The distortion for this example is 0.0007

We simulated the algorithm for 10^6 sample vectors and achieved an average distortion of

$$\textit{Average Distortion} = 0.0017$$

Method 2:

$$C_{XX} = \begin{bmatrix} \sigma_x^2 + \sigma_n^2 & -\sigma_n^2 & -\sigma_n^2 & \cdots & -\sigma_n^2 \\ -\sigma_n^2 & 2\sigma_n^2 & \sigma_n^2 & \cdots & \sigma_n^2 \\ -\sigma_n^2 & \sigma_n^2 & 2\sigma_n^2 & & \sigma_n^2 \\ \vdots & \vdots & & \ddots & \vdots \\ -\sigma_n^2 & \sigma_n^2 & \sigma_n^2 & \cdots & 2\sigma_n^2 \end{bmatrix}_{m\times m} = U\Lambda U^T \tag{4.61}$$

where,

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_m \end{bmatrix}_{m\times m} \quad \text{and} \quad UU^T = U^TU = I \tag{4.62}$$

We substitute for the parameters and using (4.42)–(4.44) we have

$$C_{XX} = \begin{bmatrix} 1.01 & -0.01 & -0.01 & -0.01 \\ -0.01 & 0.02 & 0.01 & 0.01 \\ -0.01 & 0.01 & 0.02 & 0.01 \\ -0.01 & 0.01 & 0.01 & 0.02 \end{bmatrix} = \begin{bmatrix} -0.9998 & 0.0178 & 0 & 0 \\ 0.0103 & 0.5773 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0.0103 & 0.5773 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0.0103 & 0.5773 & 0 & -\frac{2}{\sqrt{6}} \end{bmatrix}$$

$$\times\begin{bmatrix}1.0103 & 0 & 0 & 0\\ 0 & 0.0397 & 0 & 0\\ 0 & 0 & 0.01 & 0\\ 0 & 0 & 0 & 0.01\end{bmatrix}\times\begin{bmatrix}-0.9998 & 0.0103 & 0.0103 & 0.0103\\ 0.0178 & 0.5773 & 0.5773 & 0.5773\\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0\\ 0 & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{6}}\end{bmatrix} \tag{4.63}$$

$$\begin{aligned}X_1 &= 1.3098\\ X_2 &= 1.1872\\ X_3 &= 1.2236\\ X_4 &= 1.2084\end{aligned} \qquad \underline{X} = [1.3098, 1.1872, 1.2236, 1.2084]^T \tag{4.64}$$

$$Y = \begin{bmatrix}X_1\\ X_2 - X_1\\ X_3 - X_1\\ X_4 - X_1\end{bmatrix} = \begin{bmatrix}1.3098\\ -0.1226\\ -0.0862\\ -0.1014\end{bmatrix} \tag{4.65}$$

By transforming Y using U we will have

$$\hat{Y} = U^T Y = \begin{bmatrix}-0.9998 & 0.0103 & 0.0103 & 0.0103\\ 0.0178 & 0.5773 & 0.5773 & 0.5773\\ 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0\\ 0 & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{6}}\end{bmatrix}\begin{bmatrix}1.3098\\ -0.1226\\ -0.0862\\ -0.1014\end{bmatrix} = \begin{bmatrix}-1.3128\\ -0.1557\\ 0.0257\\ -0.0024\end{bmatrix} \tag{4.66}$$

Using a total rate of $R_{Total} = 8$ bits, $\hat{Y}$ will be quantized to

$$\hat{Y}_q = \begin{bmatrix}-1.4899\\ -0.1025\\ 0.0785\\ -0.0809\end{bmatrix} \tag{4.67}$$

By retransforming using U, we have

$$\widetilde{Y} = U\hat{Y}_q = \begin{bmatrix} -0.9998 & 0.0178 & 0 & 0 \\ 0.0103 & 0.5773 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0.0103 & 0.5773 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0.0103 & 0.5773 & 0 & -\frac{2}{\sqrt{6}} \end{bmatrix} \begin{bmatrix} -1.4899 \\ -0.1025 \\ 0.0785 \\ -0.0809 \end{bmatrix} = \begin{bmatrix} 1.4879 \\ -0.163 \\ -0.0521 \\ -0.0084 \end{bmatrix} \tag{4.68}$$

$$\widetilde{X} = \begin{bmatrix} \widetilde{Y}_1 \\ \widetilde{Y}_2 - \widetilde{Y}_1 \\ \widetilde{Y}_3 - \widetilde{Y}_1 \\ \widetilde{Y}_4 - \widetilde{Y}_1 \end{bmatrix} = \begin{bmatrix} 1.4879 \\ 1.3248 \\ 1.4358 \\ 1.4794 \end{bmatrix} \tag{4.69}$$

The distortion for this example is 0.0423

We simulated the algorithm for 10^6 sample vectors and achieved an average distortion of

$$Average\ Distortion = 0.0126$$

Using a total rate of $R_{Total} = 12$ bits, $\hat{Y}$ will be quantized to

$$\hat{Y}_q = \begin{bmatrix} -1.3731 \\ -0.1479 \\ 0.0393 \\ -0.0393 \end{bmatrix} \tag{4.70}$$

By retransforming using U, we have

$$\widetilde{Y} = U\hat{Y}_q = \begin{bmatrix} -0.9998 & 0.0178 & 0 & 0 \\ 0.0103 & 0.5773 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0.0103 & 0.5773 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0.0103 & 0.5773 & 0 & -\frac{2}{\sqrt{6}} \end{bmatrix} \begin{bmatrix} -1.3731 \\ -0.1479 \\ 0.0393 \\ -0.0393 \end{bmatrix} = \begin{bmatrix} 1.3702 \\ -0.1113 \\ -0.0557 \\ -0.1316 \end{bmatrix} \tag{4.71}$$

$$\widetilde{X} = \begin{bmatrix} \widetilde{Y}_1 \\ \widetilde{Y}_2 - \widetilde{Y}_1 \\ \widetilde{Y}_3 - \widetilde{Y}_1 \\ \widetilde{Y}_4 - \widetilde{Y}_1 \end{bmatrix} = \begin{bmatrix} 1.3702 \\ 1.2589 \\ 1.3145 \\ 1.2386 \end{bmatrix} \tag{4.72}$$

The distortion for this example is 0.0045

We simulated the algorithm for 10^6 sample vectors and achieved an average distortion of

$$Average\ Distortion = 0.0036$$

We simulated the scheme for Vector Quantization method over 10^6 samples and achieved:

$$R_{Total} = 8 \qquad Average\ Distortion = 0.0094$$

$$R_{Total} = 12 \qquad Average\ Distortion = 0.004$$

Table 4.2 summarizes the simulation results and Fig. 4.12 gives a visual comparison for different methods.

Summary of Results	Compression Method 1		Compression Method 2		Vector Quantization	
	8 bits	12 bits	8 bits	12 bits	8 bits	12 bits
Average Distortion	0.042	0.0017	0.0126	0.0036	0.0094	0.004
SQNR= $10\log_{10}\frac{\sigma_x^2+\sigma_n^2}{D}$ (dB)	13.81	27.74	19.04	24.48	20.31	24.02

Table 4.2 – Summary of Results for Cooperative Compression algorithms

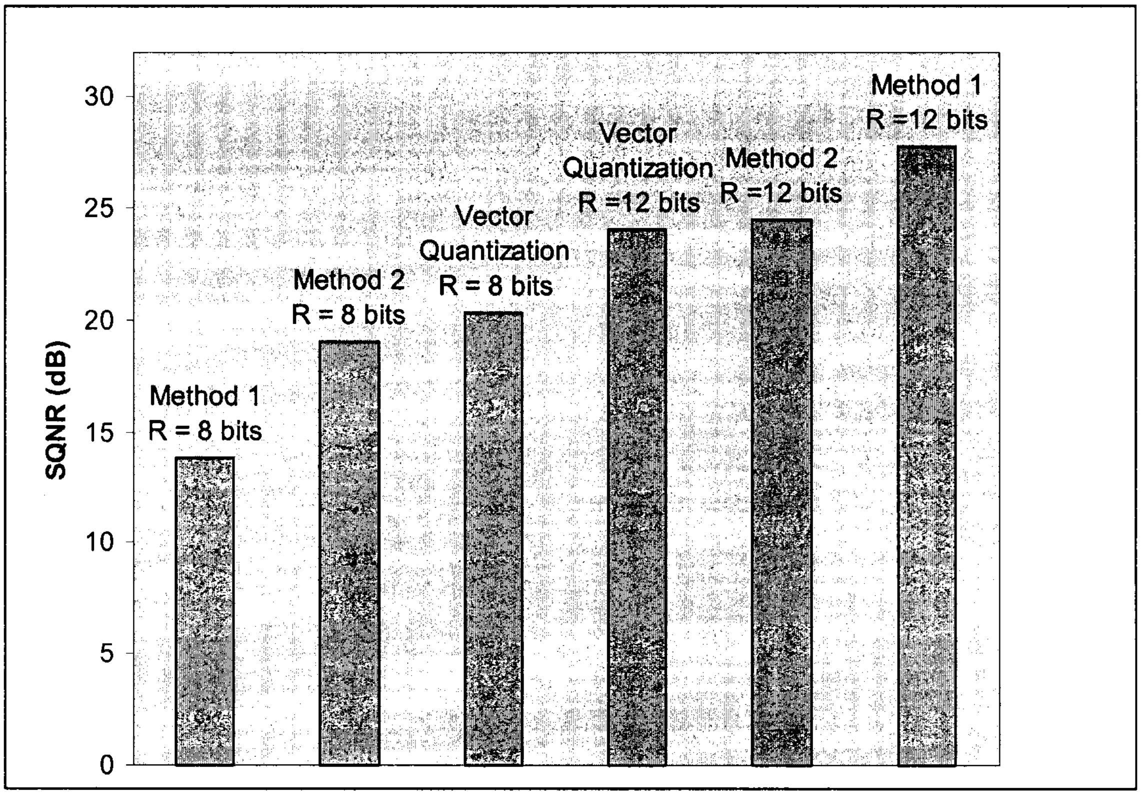

Figure 4.12 – Visual comparison among compression algorithms for rates 8 and 12

Comparison between compression Methods 1 and 2

We simulated the system considering both the conference phase and cooperative compression phase. We compared the distortion achieved by Methods 1 and 2.

The associated distortion considering only conference phase is:

Conference-Phase Distortion (*Method 1*) = 1.1912×10^{-4}

Conference-Phase Distortion (*Method 2*) = 1.3512×10^{-4}

The total distortion due to Conference Phase and Cooperative Compression Phase for total rate of 8 bits is:

Total Distortion (*Method 1*) = 0.0421

Total Distortion (*Method 2*) = 0.0128

Finally, we added the MIMO transmission block to the system and simulated the whole sensor network. The simulation results are depicted in Fig. 4.13.

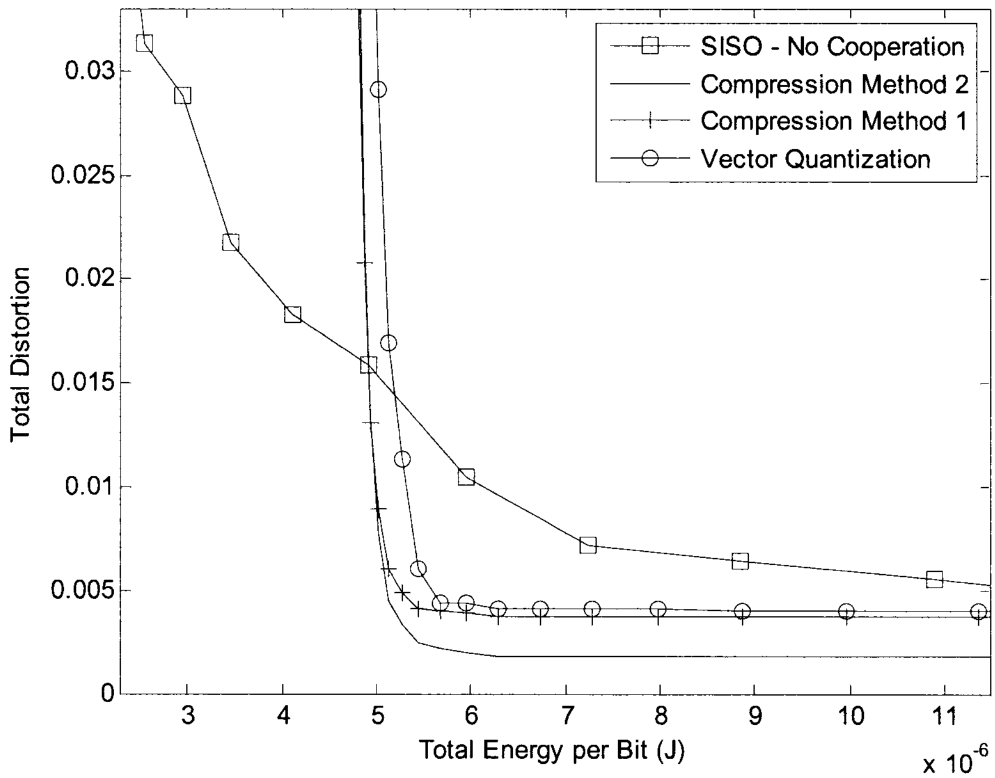

Figure 4.13 – Comparison of Total Distortion vs. Total Energy consumption per bit

From Fig. 4.13 it is observed that for distortion levels below 0.015 compression method 2 has the best performance. It is also observed that compression method 1 performs very close, but slightly better than Vector Quantization.

CHAPTER FIVE

CONCLUSION

In this thesis we considered the problems of data compression and transmission in wireless sensor networks.

In Chapter 2 we presented a background on the main concepts, theorems and algorithms which were defined in the literature and used throughout the thesis. We presented a review on the classical CEO problem and its relations to Wireless Sensor Networks. We studied Slepian-Wolf theorem, achievable rate region for Distributed Source Coding and its application in WSNs. We reviewed the Cooperative Source Coding for correlated Gaussian information by eigenvalue decomposition and transferring into parallel independent sources followed by Vector Quantization method. Finally we presented the background on Cooperative Transmission (Virtual MIMO) in WSNs and Space-Time Coding.

In Chapter 3 we considered the CEO problem for Binary sources. We modeled data for observations by an i.i.d. unbiased sequence Y which is connected to each sensor via a Binary Symmetric Channel (BSC). In other words, observations X_i ($i = 1,2,\ldots, n$) are corrupted by a random error E, with $\Pr\{E_i = 1\} = p_e$. We analyzed the behavior of the system and showed that sensors can compress their observations according to Slepian-Wolf rate limit and a joint decoder can estimate the common sequence Y using this information. We derived $H(Y \mid X_1, X_2, ..., X_n)$ as a function of n and p_e and proved that it converges to zero as n tends to infinity. Substituting this function in *Fano's inequality*, we determined the minimum number of sensors required to achieve a desired probability of error.

In Chapter 4 we considered energy-efficient compression and transmission techniques for wireless sensor networks. We assumed two scenarios for sensor networks and for both scenarios presented cooperative compression and transmission schemes. In the first scenario we considered the Gaussian CEO problem where sensors observe a common Gaussian source and report noisy versions of this source to the CEO. The Fusion Center (CEO) is interested in estimating the source by processing the observations and extracting the information. We proposed an energy-efficient cooperative algorithm for data estimation in CEO-based wireless sensor networks exploiting cooperative (virtual) MIMO technique. In the second scenario we extended the problem to Multi-terminal communication problem and considered the general sensor network case where all nodes wish to transfer their individual correlated information to a center. In this case sources are highly correlated and the Fusion Center is interested in all observations. We proposed cooperative data compression and transmission methods and compared their performance. The proposed algorithms are based on joint compression of correlated Gaussian sources. In the first algorithm we transform sources into parallel independent Gaussian sources while in the second one we apply Vector Quantization.

5.1 Contributions

Analysis of data estimation in Binary CEO problem:

- Derivation of closed form for joint Entropy $H(Y \mid X_1, X_2, ..., X_n)$ and proving its convergence to zero as n tends to infinity.
- Determination of the minimum number of sensors for any given desired distortion
- Derivation of the estimation rule from a set of observations

Cooperative compression and transmission in Gaussian Wireless Sensor Networks

- Proposed a data estimation algorithm for CEO problem in WSNs using cooperative data processing and energy-efficient MIMO transmission technique
- Proposed cooperative compression methods for general Multi-terminal communication problem in wireless sensor networks by removing the correlation from the set of data using eigenvalue decomposition technique
- Derivation of closed form parametric equations for eigenvalue decomposition
- Proposed a Vector Quantization based cooperative algorithm for compression

5.2 Future Work

For future research we suggest designing of a Distributed Source Coding scheme for data conference among sensors. Since sensors' data are correlated and they want to share their information, a well-designed DSC scheme can achieve the same performance by consuming less energy. We also suggest designing of a coding scheme for implementation of Binary CEO problem, proposed in Chapter 3, according to DSC in order to achieve Slepian-Wolf bound. For this purpose we can use powerful channel codes like Turbo or LDPC codes.

REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "Wireless Sensor Networks: A Survey," *Journal of Computer Networks*, vol. 38, pp. 393–422. 2002.

[2] B. M. Sadler, "Fundamentals of Energy-Constrained Sensor Network Systems," *IEEE A&E Systems Magazine*, vol. 20, no.8, pp. 17–35 , Aug. 2005

[3] T.J. Flynn and R. M. Gray, "Encoding of Correlated Observations", *IEEE Trans. Information Theory,* vol. IT-33, no.6, pp.773–787, Nov. 1987

[4] T. Berger, Z. Zhang and H. Viswanathan, "The CEO Problem," *IEEE Trans. Information Theory*, vol. 42, no. 3, May 1996

[5] T. Berger and Z. Zhang, "On the CEO Problem," *Proc. IEEE International Symposium on Information Theory*, p. 201, Jul. 1994

[6] H. Viswanathan and T. Berger, "The Quadratic Gaussian CEO Problem," *Proc. IEEE International Symposium on Information Theory*, p. 260, Sep. 1995

[7] Y. Oohama, "The rate distortion function for the Quadratic Gaussian CEO Problem," *IEEE Trans. Information Theory*, vol. 44, no. 3, pp. 1057–1070, May 1998

[8] D. Slepian and J. K. Wolf, "Noiseless Coding of Correlated Information Sources," *IEEE Trans. Information Theory*, vol. IT-19, no.4, pp. 471–480, Jul. 1973

[9] A. D. Wyner and J. Ziv, "The Rate-Distortion Function for Source Coding with Side Information at the Decoder," *IEEE Trans. Information Theory*, vol.IT-22, pp. 1–10, Jan. 1976

[10] J. Garcia-Frias and Y. Zhao, "Compression of Correlated Binary Sources Using Turbo Codes", *IEEE Communications Letters*, vol.5, no.10, pp.417–419, Oct. 2001

[11] J. Bajcsy and P. Mitran, "Coding for the Slepian-Wolf Problem With Turbo Codes", *Proc. IEEE Global Telecommunications Conference 2001,* vol. 2 pp. 1400 – 1404, Nov. 2001

[12] P. Mitran and J. Bajcsy, "Turbo Source Coding: A Noise-Robust Approach to Data Compression," *Proc. IEEE Data Compression Conference (DCC)*, p. 465, Apr. 2002

[13] A. Aaron and B. Girod, "Compression with Side Information Using Turbo Codes," in *Proc. IEEE Data Compression Conference (DCC)*, pp. 252–261, Apr. 2002

[14] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "A Distributed Source Coding Technique for Correlated Images Using Turbo-Codes," *IEEE Communication. Letters.*, vol. 6, no.9, pp. 379–381, Sep. 2002

[15] A. D. Liveris, Z. Xiong and C. N. Georghiades, "Compression of Binary Sources With Side Information at the Decoder Using LDPC Codes", *IEEE Communications Letters* vol.6, no.10, pp. 440–442, Oct. 2002

[16] A. D. Liveris, C. Lan, K. R. Narayanan, Z. Xiong, and C. N. Georghiades, "Slepian-Wolf Coding of Three Binary Sources Using LDPC Codes," *Proc. 3rd Intl. Symp. on Turbo codes*, Sep. 2003

[17] D. Schonberg, K. Ramchandran, and S. S. Pradhan, "Distributed Code Constructions for the Entire Slepian-Wolf Rate Region for Arbitrarily Correlated Sources," *Proc. IEEE Data Compression Conference (DCC)*, Mar. 2004

[18] S. S. Pradhan and K. Ramchandran, "Distributed Source Coding Using Syndromes (DISCUS): Design and Construction," *IEEE Trans. Information Theory*, vol. 49, no. 3, pp. 626–643, Mar. 2003

[19] S. S. Pradhan, J. Kusuma and K. Ramchandran, "Distributed Compression in a Dense Microsensor Network," *IEEE Signal Processing Magazine*, pp. 51–60, Mar. 2002

[20] J. Chou, D. Petrovic and K. Ramchandran "A Distributed and Adaptive Signal Processing Approach to Reducing Energy Consumption in Sensor Networks," *Proc. IEEE INFOCOM*, vol. 2, pp. 1054–1062 Mar. 2003

[21] Z. Xiong, A. D. Liveris and S. Cheng, "Distributed Source Coding for Sensor Networks," *IEEE Signal Processing Magazine*, pp. 80–94, Sep. 2004

[22] M. Sartipi and F. Fekri, "Distributed Source Coding in Wireless Sensor Networks using LDPC coding: The entire Slepian-Wolf Rate Region," *Proc. IEEE WCNC 2005*, pp. 1939–1944, Mar. 2005

[23] Z.-Q. Luo, "An Isotropic Universal Decentralized Estimation Scheme for a Bandwidth Constrained Ad Hoc Sensor Network," *IEEE Jour. Selected Areas. Comm.* vol.23, no.4, pp. 735–744, Apr. 2005

[24] Z.-Q. Luo, "Universal Decentralized Estimation in a Bandwidth Constrained Sensor Network," *IEEE Trans. Information Theory*, vol.51, no.6, pp. 2210–2219, Jun. 2005

[25] Z.-Q. Luo and J.-J. Xiao, "Decentralized Estimation in an Inhomogeneous Sensing Environment," *IEEE Trans. Information Theory,* vol.51, no.10, Oct. 2005

[26] J.-J. Xiao, S. Cui, Z.-Q. Luo and A.J. Goldsmith, "Joint Estimation in Sensor Networks under Energy Constraints," *Proc. IEEE First conference on Sensor and Ad Hoc Communications and Networks, (SECON 04),* Oct. 2004

[27] J.-J. Xiao, S. Cui, Z.-Q. Luo and A.J. Goldsmith, "Power Scheduling of Universal Decentralized Estimation in Sensor Networks," *IEEE Trans. Signal Processing*, vol. 54, no. 2 Feb. 2006

[28] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-Efficiency of MIMO and Cooperative MIMO Techniques in Sensor Networks," *IEEE Jour. Selected Areas Comm.,* vol. 22, no. 6 pp. 1089–1098, Aug. 2004

[29] S. Cui and A. J. Goldsmith, "Energy Efficient Routing Based on Cooperative MIMO Techniques," *Proc. IEEE ICASSP 2005*, pp.805–808, Mar. 2005

[30] S. K. Jayaweera, "Energy Analysis of MIMO Techniques in Wireless Sensor Networks", *38th Annual Conference on Information Sciences and Systems (CISS 04)*, Mar. 2004

[31] S. K. Jayaweera, "Virtual MIMO based Cooperative Communication for Energy-constrained Wireless Sensor Networks", *IEEE Trans. Wireless Communications*, vol. 5, no. 5, May 2006

[32] S. K. Jayaweera and M. L. Chebolu, "Virtual MIMO and Distributed Signal Processing for Sensor Networks–An integrated approach", *Proc. IEEE International Conf. Comm. (ICC 05)*, May 2005

[33] M. L. Chebolu and S. K. Jayaweera, "Integrated Design of STBC-based Virtual-MIMO and Distributed Compression in Energy-limited Wireless Sensor Networks" *Proc. 2nd European Workshop on Wireless Sensor Networks (EWSN 05)*, Istanbul, Turkey, Feb. 2005

[34] S. M. Alamouti, "A Simple Transmit Diversity Technique for Wireless Communications," *IEEE Jour. Selected Areas Comm.*, vol. 16, no. 8, pp. 1451–1458, Oct. 1998

[35] V. Tarokh, H. Jafarkhani, and A. R. Calderbank. "Space-Time Block Codes from Orthogonal Designs,'' *IEEE Trans. Information Theory*, vol. 45, no. 5, pp. 1456–1467, Jul. 1999

[36] R. M. Gray, "Vector Quantization," *IEEE ASSP Magazine*, pp. 4–29, Apr. 1984

[37] J. Max, "Quantizing for Minimum Distortion," *IRE Trans. Information Theory,* vol. IT-6, pp.7–12, Mar. 1960

[38] S.P. Lloyd, "Least Squares Quantization in PCM ," *IEEE Trans. Information Theory*, vol. IT-28, pp. 129–137, Mar. 1982.

[39] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, 1991

[40] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*. McGraw-Hill , 2002

[41] A. Heshmati and M. R. Soleymani, "An Energy-Efficient Cooperative Algorithm for Data Estimation in Wireless Sensor Networks" *Proc. Canadian Conference on Electrical and Computer Engineering*, Apr. 2007