PRACTICAL CODING ALGORITHMS FOR CONSENSUS PROBLEMS WITH ZERO ASYMPTOTIC RATE

A Thesis

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Master of Science

by

Mehmet Ercan Yildiz

May 2007

ABSTRACT

We consider the average consensus algorithm under the rate constraint communication network. Average consensus algorithms are protocols to compute the average value of all sensor measurements via near neighbors communications. The main motivation for our work is the observation that *consensus algorithms* offer the perfect example of a network communication problem where there is an increasing correlation between the data exchanged, as the algorithm iterates. Henceforth, it is possible to utilize previously exchanged data and current side information to reduce the demands of quantization bit rate for a certain precision. We analyze the case of a network with a topology built as that of a random geometric graph and with links that are assumed to be reliable at a constant bit rate. We explore the conditions on the quantization noise which lead to a consensus value whose mean squared distance from the initial average is bounded. We propose two main practical schemes and show that they achieve bounded convergence with zero rate asymptotically. We further investigate the problem under regular grid network assumption and observe that computational complexity of the schemes reduce significantly and global knowledge of the network connectivity assumption can be relaxed. Thus, we conclude that the proposed schemes become scalable under dense networks.

BIOGRAPHICAL SKETCH

Ercan Yildiz was born in 1982 in Istanbul, Turkey. He received his Bachelor of Science Degree at University of Wisconsin in Electrical Engineering. He also received additional major in Mathematics and Designation in Research from the same institution. Between 2003-2005, he worked as an undergraduate assistant and focused on network tomography and statistical learning theory.

Since Fall 2005 Ercan has been working under the guidance of Professor Anna Scaglione at Cornell University. His interests include sensor networks, distributed processing, information theory and statistical learning theory. He plans to continue his PhD studies at Cornell university starting May 2007.

To my parents, my sister Elif, and my dearest friend Kamer.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor, Prof. Anna Scaglione. I would like thank her for enthusiasm and motivation for research, and her understanding during my enjoyable but hard journey.

I also would like to thank Prof. Lang Tong for participating in my committee and setting time for me whenever I needed his advice besides his extremely busy schedule.

Many thanks to friend and colleagues who made my stay at Cornell University enjoyable. I would like to thank CRISP members and alumni, Matt, Shrut, Birsen and Azadeh for enjoyable conversations and discussions.

My best attempts to thank my dearest friend, and fiance-to-be Kamer Toker would not be sufficient. I would like to thank her for her courage, support and love.

I would like to thank my parents for their support during the hard times. I am well aware that I would not accomplish what I did without their presence. I also would like to thank my sister, Elif, for her endless love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Introduction

Vicsek et al. in their 1995 study [1] highlighted the complex cooperative behavior arising in many particle systems. The emergence of complex *cooperative* behavior has attracted many researchers from a wide range of fields over the years. For instance, biologist have studied schools of fish, flocks of flying birds, cooperative motion of bacteria under unfavorable conditions [2],[3]. Physicists have applied *cooperation* theory to understand behavior and transition of dynamic systems of particles[4],[1],[5]. Computer scientists have developed cooperation models to simulate flocking and schooling behaviors for the animation industry[5].

The *cooperation* problem has also been studied extensively in control and communication fields. There is a wide range of applications, such as coordination of autonomous and geographically separated field agents, distributed computing and congestion control, tracking objects by several unmanned air vehicles (UAVs) and decentralized reconstruction or compression of a field, where decentralized cooperation strategies have to be designed. In this area, significant contributions came by Morse et al., who studied coordination of mobile agents which move on a plane with different bearings and aim to converge in the same direction [5]. In the same context, rendezvous of these agents on a common point only with local communication has been also studied[6]. In [7] and [8], the problem is analyzed under the context of fault tolerant distributed services for computers. Weigang et al. have proposed a consensus protocol for mobile ad hoc networks in [9]. *Cooperation* problem has also been studied in the context of *information diffusion* subject where the nodes which have already obtained the information, spread it to the rest of the network via local communication as in [10].

In [11], Saber and Murray have focused on a specific kind of cooperation which is called *average consensus*. This problem investigates the case in which, nodes(agents, sensors) want to converge to the average of the initial values by updating themselves using weighted average of their neighbors' values. The authors have analyzed continuous time update scenario with dynamically changing topology (i.e. neighbors). Moreover, they have explored the necessary conditions on the update weights and connectivity of the sensors for convergence. In [12], Ren et al. have studied consensus problem under Kalman filtering context. Rabbat et al. have focused on consensus problem with binary erasure links between neighboring nodes[13]. In [14] *asynchronous* average consensus algorithms are studied where at each iteration a randomly chosen node exchanges its value with one of the neighbors. On the contrary, in regular (*synchronous*) consensus algorithms the nodes exchange values with their neighbors through static or dynamic topology.

The motivation of our work lies in some of the results shown in [15] by Boyd et al., who derived the convergence conditions for discrete time *synchronous average consensus*. The same authors have explored the convergence characteristics of the problem when the state values are communicated in the presence of additive independent noise with *fixed* variance in [16]. There are two discouraging outcomes of this analysis. First, authors show that the system converges to the initial average only in mean, but has increasing error variance as the number of iterations grows. Second, the node values do not necessarily converge to a common value asymptotically. The key observation here is that *consensus algorithms* offer the perfect example of network communications problems in which there is cor-

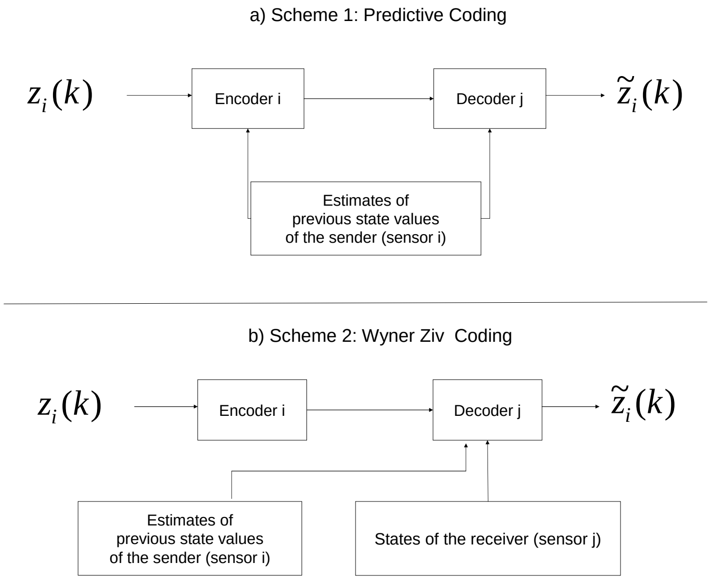

a) Scheme 1: Predictive Coding
$z_i(k)$
Encoder i
Decoder j
$\tilde{z}_i(k)$
Estimates of previous state values of the sender (sensor i)
b) Scheme 2: Wyner Ziv Coding
$z_i(k)$
Encoder i
Decoder j
$\tilde{z}_i(k)$
Estimates of previous state values of the sender (sensor i)
States of the receiver (sensor j)

Figure 1.1: Encoding/Decoding strategies for consensus problem

relation between the data exchanged, and the correlation increases as the system updates its computations. If this fact is properly accounted for in quantizing the messages, the negative result in [16] should no longer be considered as indicative of the performance of the consensus algorithms over digital communication networks. To prove it, we introduce and analyze a more detailed internode communication model, where the noise added to each computation is the result of the data source encoding. In particular, we assume that at every iteration a message can contain a finite number of bits and explore the effect of the quantization errors arising due to this finite communication rate constraint. We explore the constraints on the quantization noise under which the system converges to a consensus without discouraging effects mentioned in [16]. Furthermore, we propose practical scalar quantizers based on predictive and nested lattice Wyner-Ziv encoding schemes, show that a consensus can be achieved with zero transmission rate asymptotically, and give achievable rate regions numerically. We discuss computational complexity and scalability of these schemes under the dense network assumption.

Two main strategies are given in Fig. 1.1. We denote by $z_i(k)$ the unquantized message of node i at iteration k and, by $\tilde{z}_i(k)$ its reconstructed value at the decoder side. For both strategies, we consider the two possible communication scenarios in Fig.1.2 called, respectively, the *peer to peer* case and the *broadcast* case. As seen in Fig.1.2 a) the peer to peer schemes utilize a different encoder for each particular destination while the broadcast schemes in Fig.1.2 b) require only one encoder for all receivers of a given sender. The reason of this distinction is that the *peer to peer* methods generally outperform the *broadcast* methods, but at the price of a more complex encoder structure and of forcing to send a different message to each neighbor. While sending a different message over each link is acceptable in a wired network, it is wasteful in a wireless medium, where communications are naturally broadcast and each transmission reaches all neighbors. For this reason, the *peer to peer* methods are proposed for *wired networks* and the *broadcast* methods are proposed for *wireless networks.* As mentioned above, through this study it is possible to reach a more positive conclusion compared to that in [16].

1.2 Organization

The paper is organized as follows: In Chapter 2, we review the main mathematical relationships characterizing average consensus algorithms. In Chapter 3, we explore the conditions on the noise variance under which the system converges to a consensus. In Chapter 4, we propose a predictive coding scheme which has the structure shown in Fig.1.1a. In Chapter 5, we discuss Wyner-Ziv encoder/decoder

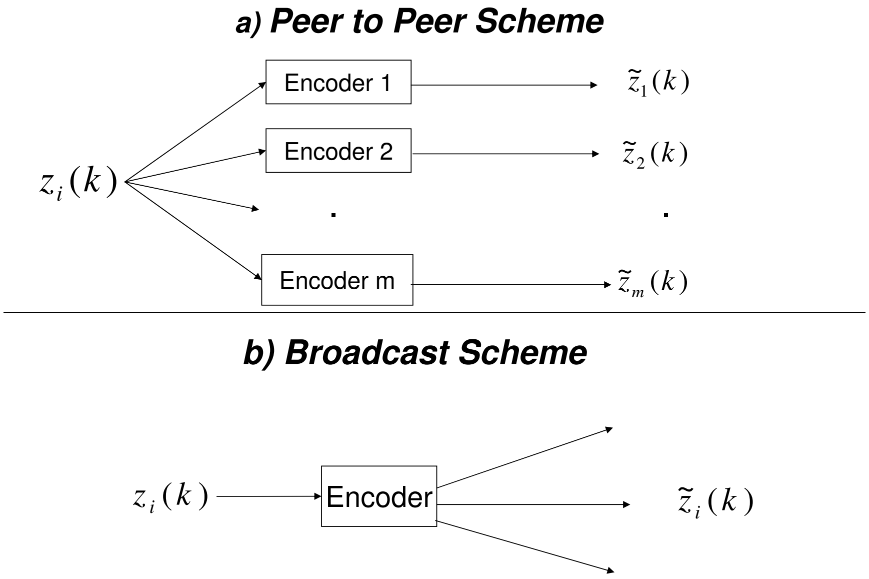


Figure 1.2: Peer to Peer vs Broadcast

scheme as in Fig.1.1b. We discuss scalability of the proposed algorithms under dense sensor networks in Chapter 6. We simulate and give the corresponding results in Chapter 7. We conclude the paper in Chapter 8.

Chapter 2

System Model

2.1 Background on Algebraic Graph Theory

In this section, we introduce some basic concepts and notation in graph theory, which will be used in the rest of the paper. A *graph* G is a pair of sets (V, E) where V is a non-empty set of *vertices* and E is a set of unordered pairs of different vertices which are called *edges*. Fig. 2.1 shows a graph G where $V = \{v_1, v_2, v_3, v_4\}$ and $E = \{e_1, e_2, e_3, e_4, e_5\}$. An edge can also be represented by a unique unordered pair of vertices, i.e. $e_1 = (v_1, v_2) = (v_2, v_1)$.

Remark 1. *We note that there exists another family of graphs which is called directional graphs. In this case, a pair of vertices represents two edges where the order of the vertices represents the direction of an edge. Since we assume that the links are bidirectional and channel transmission is error free, directional graphs are not of particular interest.*

Given a graph G, we denote $(0, 1)$ adjacency matrix of G as A where:

$$\begin{aligned} &a_{ij} = 1, \quad \text{if there is an edge between node } v_i \text{ and } v_j \\ &a_{ij} = 0, \quad \text{if } v_i = v_j \text{ or } v_i \neq v_j \text{ and there is no edge} \end{aligned} \tag{2.1}$$

We note that A matrix is symmetric and there does not exist any edges pointing a particular vertex to itself. The set of *neighbors* of the node v_i is denoted by N_i. Mathematically, $j \in N_i \iff a_{ij} = a_{ji} = 1$. The number of neighbors of a vertex is called its degree and denoted by deg_i:

$$deg_i = \sum_{j=1}^{n} a_{ij} = \sum_{j=1}^{n} a_{ji}$$

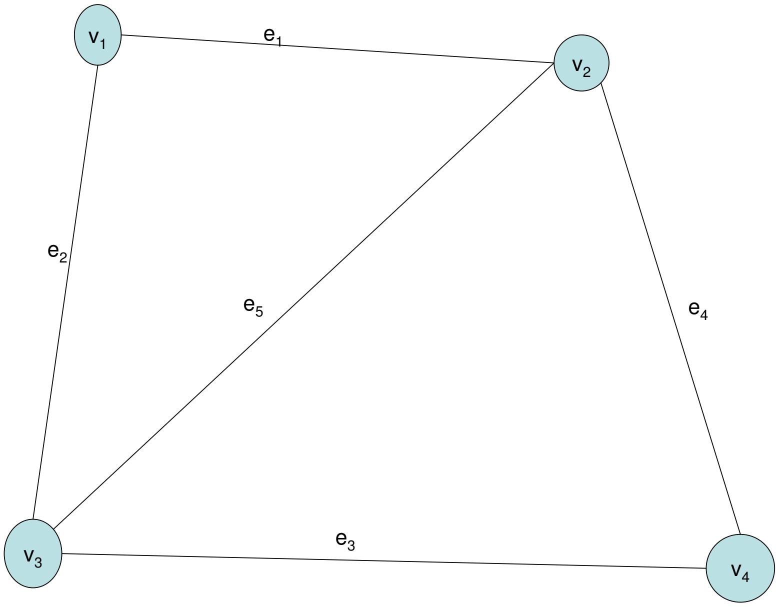


Figure 2.1: A sample graph

We define the degree matrix of a graph G as Δ where Δ is a diagonal matrix each diagonal entry corresponding to the degree of a node:

$$\Delta_{ij} = 0, \qquad \text{if} \;\; i \neq j$$
$$\Delta_{ij} = deg_i, \;\; \text{if} \;\; i = j$$

We can also represent Δ as $\Delta = diag(A\mathbf{1})$ where $\mathbf{1}$ is the all *ones* vector. The difference between degree matrix and adjacency matrix of a graph G is the *Laplace* matrix:

$$L = \Delta - A$$

While we do not use the Laplace matrix in our study explicitly, previous research on consensus algorithms include the extensive analysis of L. Interested readers may refer to [17] for the detailed discussions about Laplace matrices.

Remark 2. *A graph G is strongly connected if for every pair of vertices v_i and v_j, there is a path from v_i to v_j, and v_j to v_i. Since we are constraining W matrices* *to be symmetric, existence of a path from v_i to v_j implies that there exists a path from v_j to v_i.*

2.2 Problem Formulation

We consider distributed linear average consensus algorithms. In the rest of the paper, we will substitute *vertex* with *node* and *edge* with *link* for the sake of consistency with communication literature.

We denote the value of node i at iteration k by $x_i(k)$. Then, consensus update model is given as:

$$x_i(k+1) = x_i(k) - \sum_{j=1, j\neq i}^{n} \epsilon l_{ij} \big(x_j(k) - x_i(k)\big), \tag{2.2}$$

for $i = 1, \ldots, n \;\; k = 0, 1, \ldots$. We can rewrite (2.2) in vector form as follows:

$$x(k+1) = (I - \epsilon L)x(k) = Wx(k). \tag{2.3}$$

where $x(k) = [x_1(k), \ldots, x_n(k)]^T$,$L$ is the Laplace matrix, I is the rank n identity matrix and $\epsilon \in R^+$.

It was shown by Xiao and Boyd that above system converges to the average of any initial vector $x(0) \in R^n$ if and only if

1. $\mathbf{1}^T W = \mathbf{1}^T$
2. $W\mathbf{1} = \mathbf{1}$
3. $\parallel W - \frac{1}{n}\mathbf{1}\mathbf{1}^T \parallel < 1$

where the norm is maximum singular value norm [15]. In other words, convergence is satisfied if 1 is an eigenvalue of W and it is also the eigenvalue with greatest magnitude. The intuition behind these conditions can be summarized as follows:

The algorithm tends to compute the average of initial state vector ($x(0)$), therefore it should preserve sums, i.e. $\mathbf{1}^T W = \mathbf{1}^T$. Moreover, the vector of averages has to be a fixed (stable) point of the iteration , i.e. $W\mathbf{1} = \mathbf{1}$. The third condition guarantees that $lim_{k\to\infty} W^k = \frac{\mathbf{1}\mathbf{1}^T}{n}$, therefore $x(k) \to \frac{\mathbf{1}\mathbf{1}^T}{n} x(0)$.

Similar results for continuous time iterations are in [11]. For $A = A^T$, by constraining $0 < \epsilon < 1/max(A\mathbf{1})$, we guarantee that eq.(2.3) asymptotically converges to average as in [15].

If we decompose eq. (2.3) as follows:

$$x(k+1) = (I - \epsilon\Delta)x(k) + \epsilon A x(k) \tag{2.4}$$

we see that there are two different parts in the update: computing $(I - \epsilon\Delta)x(k)$ requires only local values and $\epsilon A x(k)$ uses the neighbors' values. We can therefore define

$$z(k) = \epsilon x(k), \tag{2.5}$$

as the vector of variables, which needs to be exchanged over the links available in G at each iteration k. The entries of the vector $z(k)$, quantized with finite precision, are reconstructed as:

$$\tilde{z}(k) = z(k) + w(k)$$

where $w(k)$ is the quantization error.

While there exists a substantial body of work on average consensus protocols under infinite precision and noiseless peer to peer communications, little research has been done introducing distortions in the message exchange, such as the noisy update assumption made in [16]. Specifically, Xiao and Boyd consider the following extension of (2.2):

$$x_i(k+1) = x_i(k) - \sum_{j=1, j\neq i}^{n} \epsilon l_{ij}\big(x_j(k) - x_i(k)\big) + w_i(k) \tag{2.6}$$

where $w_i(k), i = 1, \ldots, n, \quad k = 0, 1, \ldots$ are independent zero mean *fixed* variance identically distributed random variables. The authors show that under these assumptions, the system converges to the initial average only in mean (i.e. we do not have mean squared convergence) and that the mean squared error (MSE) deviation from the actual average increases beyond a certain iteration. The authors also show that the node values do not converge to a common value as the number of iterations increases. We propose that under a detailed internode communication model $w_i(k)$ can be characterized as the quantization noise. If increasing correlation among the node states is taken into account, it can be shown that the variance of the quantization noise diminishes. Thus the negative results of [16] should not be used as the performance indicators of consensus algorithms over digital networks. This is the main contribution of our study.

2.3 Main Assumptions

In the rest of the paper, we assume that the data in the initial state vector $x(0)$ are random variables with zero mean and finite variance. We also assume that communication noises at each step and sensor are spatially and temporally independent zero mean random variables. These assumptions are met to a very close approximation in the context of quantization theory as discussed in [18]. Since all the variables we are going to deal with are zero mean, the appellative of covariance and correlation will be used interchangeably.

While our results and encoding methods can be applied in the most general case, all our numerical results will consider random geometric graphs to set up the topology of G. A random geometric graph $G(n, r)$ is a graph whose nodes are uniformly distributed points over a fixed area and where a link exists between any two nodes that are at a range less than r (r is called connectivity radius). We assume that transmission over the range r are always successful, that no channel errors are added, and that the nodes have fixed locations so that the topology of G is fixed. We also assume that nodes are strongly connected, and W satisfies convergence conditions for update structure defined in (2.3).

Chapter 3

Theoretical Results

3.1 Convergence Conditions Under Noisy Communication Model

In this section, we derive necessary and sufficient conditions on the quantization noise variances at each iteration and sensor, so that the nodes converge to a common value. Then, we give additional constraints that lead to a consensus where the final value is bounded from the initial mean in the mean squared sense. We will focus on the behavior of the transmitted random vector $z(k) = \epsilon x(k)$ rather than $x(k)$ for easy to follow derivations. In the rest of the paper we will be using the following system model:

$$\begin{aligned} x(k+1) &= (I - \epsilon\triangle)x(k) + A\tilde{z}(k) & (3.1) \\ z(k+1) &= Wz(k) + \epsilon A w(k) & (3.2) \end{aligned}$$

where $w(k)$ is the quantization error vector. The noise vector is assumed to be spatially and temporally independent. The equations above are straightforward to derive from eq. (2.4) and eq. (2.5).

Lemma 1. *The nodes converge to a consensus, if and only if the noise variance at each sensor converges to* 0*, i.e.* $E\{w_i^2(k)\} \to 0$ *as* $k \to \infty$ $\forall i \in \{1, \ldots, n\}$.

Proof. Define the mean-squared deviation from the current mean as:

$$\delta(k) = \sum_{i=1}^{n} E\{z_i(k) - \frac{\mathbf{1}^T}{n} z(k)\}^2 \qquad (3.3)$$

It is clear that the nodes reach to a consensus if and only if $\delta(k) \to 0$. Define:

$$m(k) = z(k) - \frac{1}{n}\mathbf{1}\mathbf{1}^T z(k) = z(k) - Jz(k) = (I - J)z(k) \qquad (3.4)$$

$m(k)$ is the distances of the node values from their mean. We denote covariance matrix of $m(k)$ as, $\Theta(k) = E\{m(k)m^T(k)\}$. It is obvious that $\delta(k) = Trace(\Theta(k))$. Recursion structure of $m(k)$ can be written as:

$$\begin{aligned} m(k+1) &= (I-J)z(k+1) = (I-J)Wz(k) + (I-J)\epsilon Aw(k) \\ &= (W-J)z(k) + (I-J)v(k) = (W-J)m(k) + (I-J)v(k) \end{aligned}$$

where $v(k) = \epsilon Aw(k)$. Then,

$$\Theta(k+1) = (W-J)\Theta(k)(W-J) + (I-J)E\{v(k)v^T(k)\}(I-J) \tag{3.5}$$

Since $|W-J| < 1$, $\Theta(k+1)$ converges to $\mathbf{0}$ if and only if $E\{v(k)v^T(k)\}$ approaches $\mathbf{0}$ as $k \to \infty$. Then, $\epsilon^2 AE\{w(t)w^T(t)\}A \;\; \to \mathbf{0}$. Since $\epsilon > 0$, $a_{ij} \geq 0 \ \forall i,j \in \{1,\ldots,n\}$, and $E\{w(t)w^T(t)\}$ is a diagonal matrix, each of the diagonal entries approaches 0, i.e. $E\{w_i(k)\}^2 \to 0$. □

We note that speed of the convergence does not change the fact that the nodes will reach to a consensus.

Corollary 1. *If the nodes values converge to a consensus then,* $E\{z(k)z^T(k)\} \to \Sigma^\star$ *where* $\Sigma^\star$ *is in the form of* $\alpha\mathbf{1}\mathbf{1}^T$*.*

Proof. Denote $\Sigma(k) \triangleq E\{z(k)z^T(k)\}$. By recursion:

$$\Sigma(k) = W\Sigma(k-1)W + \epsilon^2 AE\{w(k-1)w^T(k-1)\}A \tag{3.6}$$

If the nodes converge, $\epsilon^2 AE\{w(k)w^T(k)\}A \to \mathbf{0}$ by Lemma 1. Moreover, since W is balanced and the largest eigenvalue of W is 1, $\lim_{k\to\infty} W^k = \frac{\mathbf{1}\mathbf{1}^T}{n}$. Then there exist an N such that,

$$\Sigma^\star = \frac{\mathbf{1}\mathbf{1}^T}{n}\Sigma(N)\frac{\mathbf{1}\mathbf{1}^T}{n} \tag{3.7}$$

□

Unfortunately without further constraints on the quantization noises, the nodes may agree on a value which is very far from the initial average. In fact, if the noise variances converge to zero slow enough, the consensus value may even be infinity. For this reason, we derive the conditions for bounding the final value from the initial mean in mean squared sense.

We denote the average of the state values at time k by $a(k)$:

$$a(k) = \frac{1}{n}\sum_{i=1}^{n} x_i(k) = \frac{1}{n}\mathbf{1^T}x(k)$$

We also denote the average of the exchanged vector $z(k)$ by $b(k)$, which has the simple relation of $b(k) = \epsilon a(k)$. In the rest of the section, we focus on $b(k)$ to streamline the derivations. Then,

$$\begin{aligned} b(k+1) &= \frac{1}{n}\mathbf{1^T}z(k+1) = \frac{1}{n}\mathbf{1^T}Wz(k) + \frac{\epsilon}{n}\mathbf{1^T}Aw(k) \\ &= b(k) + \frac{\epsilon}{n}\mathbf{1^T}Aw(k) \qquad (3.8) \end{aligned}$$

where (3.8) follows the fact that $\mathbf{1}^T$ is an eigenvector of W with corresponding eigenvalue 1. We note that $E\{b(k)\} = E\{b(0)\} = 0$ since the noise vector is an independent quantity with zero mean and initial states are assumed to be zero mean. We are interested in the behavior of the expected mean squared distance between the asymptotic average and initial average. In particular from (3.8):

$$\begin{aligned} E\{(b(k)-b(0))^2\} &= E\left\{\left(\frac{\epsilon}{n}\sum_{t=0}^{k-1}\mathbf{1^T}W^{k-1-t}Aw(t)\right)^2\right\} \\ &= (\frac{\epsilon}{n})^2 E\left\{\left(\sum_{t=0}^{k-1}\mathbf{1^T}Aw(t)\right)^2\right\} \\ &= (\frac{\epsilon}{n})^2\sum_{t=0}^{k-1}E\{(\mathbf{1^T}Aw(t))^2\} \qquad (3.9) \\ &= (\frac{\epsilon}{n})^2\sum_{t=0}^{k-1}\sum_{i=1}^{n}\sum_{l=1}^{n}a_{li}^2 E\{w_i^2(t)\} \qquad (3.10) \end{aligned}$$

$$= (\frac{\epsilon}{n})^2 \sum_{i=1}^{n} \left(\sum_{t=0}^{k-1} E\{w_i^2(t)\} \sum_{l=1}^{n} a_{li}^2 \right) \quad (3.11)$$

where (3.9) is due to temporal independence of the noise, and (3.10) follows spatial independence of the noise. We would like to explore the conditions under which the above sum is bounded as $k \to \infty$.

Lemma 2. *Given finite number of sensors ($n < \infty$) and bounded A matrix ($a_{li} < \infty$; $\forall\, l, i \in \{1, \ldots, n\}$), mean squared deviation from the initial average is bounded if and only if $lim_{k\to\infty} \sum_{t=0}^{k-1} E\{w_i^2(t)\}$ converges $\forall\ i \in \{1, \ldots, n\}$. Therefore, a necessary and sufficient condition for $E\{(b(k) - b(0))^2\}$ to be bounded is that the noise variances at each sensor form a convergent series.*

Proof. Proof of the lemma is straightforward from (3.11) and omitted. □

We would like to note that by bounding $E\{(b(k) - b(0))^2\}$, we guarantee that $E\{(a(k) - a(0))^2\}$ is also bounded since $E\{(b(k) - b(0))^2\} = \epsilon^2 E\{(a(k) - a(0))^2\}$.

We derived necessary and sufficient conditions on the noise variances for agreeing on a common value whose mean squared distance from the average of the initial states $a(0)$ is bounded. The implication of this is that if the quantization rates can be chosen such that noise variances at each sensor form a convergent series one can guarantee that the nodes will converge to the same value and that the error with respect to the actual value will be bounded. One way to achieve this convergence is to choose the communication rates such that the quantization noise variances decay exponentially. In fact, any convergent sequences such as p-series[2] with $p > 1$ and geometric series[3] with $\alpha > 1$ would be sufficient.

[2] $\sum_{k=1}^{\infty} \frac{1}{p^k}$

[3] $\sum_{k=1}^{\infty} \frac{1}{k^\alpha}$

If one is to consider simple source coding (i.e. just quantize and transmit the state values $z(k)$), one would need a non-zero quantization rate(in bits) as $k \to \infty$ for achieving zero noise variance. In the following sections, we will propose two practical schemes and show that they achieve bounded convergence with zero rate asymptotically.

Chapter 4

Predictive Coding

In this chapter, we study the predictive encoding/decoding algorithm in Fig.1.1 a) for the *broadcast* and *peer to peer* scenarios.

4.1 Background on Predictive Coding

The predictive coding method utilizes the past quantized node values to decrease uncertainty of the present value, thus decreasing the transmission rate. Fig. (4.1) shows the block diagram of a predictive encoder/decoder pair. The current value $z_i(k)$ is predicted by linear combination of N previous outputs of the system:

$$\hat{z}_i(k) = a^{(1)}(k)\tilde{z}_i(k-1) + a^{(2)}(k)\tilde{z}_i(k-2) + \ldots + a^{(n)}(k)\tilde{z}_i(k-n)$$

The difference between the state value and the predicted value is called prediction error and applied to the quantizer whose characteristics are also known at the decoder. We note that the predictor is not in terms of the original previous state values but the previous outputs of the encoder. The idea behind is that the original previous state values are not available at the decoder. Interested readers may refer to [19] for more discussion regarding predictive coding algorithm and its performance.

This method is suitable for the consensus problem since as the algorithm iterates, the previous outputs of the decoders become more and more correlated with the current value, thus decreasing prediction error. As mentioned earlier, this reduces the quantization rate demand significantly. Furthermore, since the prediction error approaches 0 as algorithm iterates, zero quantization rate and

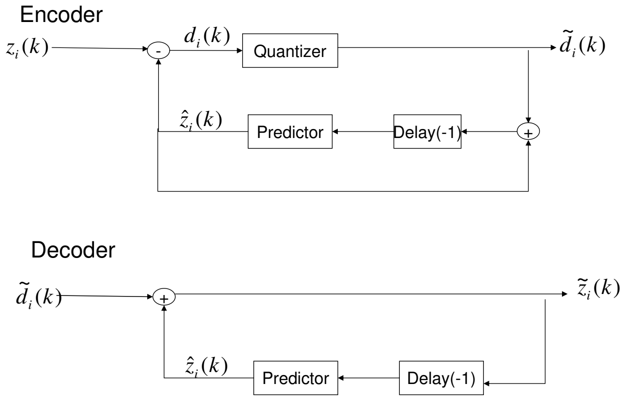


Figure 4.1: Differential encoder/decoder diagram

convergence can be achieved at the same time asymptotically.

4.2 Broadcast Predictive Coding

To exploit the local temporal correlation, the nodes can digitize $z_i(k)$ in (2.5) via the differential encoding/decoding scheme in Fig.4.1.

For each node i and time instant k, define:

$$\begin{aligned} \hat{z}_i(k) &= \quad \text{prediction} \\ d_i(k) &= \quad \text{prediction error} \\ \tilde{d}_i(k) &= \quad \text{quantized prediction error} \\ \tilde{z}_i(k) &= \quad \text{noisy reconstruction} \\ w_i(k) &= \quad \text{quantization error} \end{aligned}$$

$\tilde{z}_i(k)$ is derived through the following steps:

$$\hat{z}_i(k) = \sum_{l=1}^{p\leq k} a_i^{(k)}(l)\tilde{z}_i(k-l) \qquad (4.1)$$

$$\begin{aligned} d_i(k) &= z_i(k) - \hat{z}_i(k) \\ &= z_i(k) - \sum_{l=1}^{p\leq k} a_i^{(k)}(l)\tilde{z}_i(k-l) \end{aligned} \tag{4.2}$$

$$\begin{aligned} \tilde{d}_i(k) &= Q[d_i(k)] = Q[z_i(k) - \hat{z}_i(k)] \\ &\approx z_i(k) - \hat{z}_i(k) + w_i(k) \end{aligned} \tag{4.3}$$

$$\tilde{z}_i(k) = \hat{z}_i(k) + \tilde{d}_i(k) \tag{4.4}$$

$$= z_i(k) + w_i(k). \tag{4.5}$$

where in (4.1) $\hat{z}_i(k)$ is a linear minimum mean squared estimate(LMMSE) of $z_i(k)$ of order p; $d_i(k)$ in (4.2) is the prediction error, to be quantized and transmitted; (4.3) is because quantization noise can be modeled as additive, and (4.4) is the reconstruction of $z_i(k)$ at the decoder. (4.5) shows us that $z_i(k)$ can be reconstructed at the receiver within some noise $w_i(k)$. We note that in predictive coding scheme prediction error is applied to the quantizer, and the output is transmitted to the decoder. Since linear predictor $\hat{z}_i(k)$ is also available at the decoder, $z_i(k)$ can be reconstructed.

Assuming the $R_i(k)$ is the transmission rate of sensor i and that we use a finite range uniform scalar quantizer[4] at iteration k, the range of the quantizer and step-size can be chosen as follows:

$$\eta_i(k) = c * STD\left[d_i(k)\right] \ , \quad \triangle_i(k) = \eta_i(k)2^{-R_i(k)}. \tag{4.6}$$

where $STD[.]$ is the standard deviation and c is the parameter to be tweaked for the performance. Under the high resolution assumption, we can further approximate the quantization error as a uniform random variable independent from $d_i(k)$ and

[4] Nonuniform quantizers such as Lloyd-Max may also be implemented and may result in better performance. Our choice is just for the sake of simplicity.

all the above variables at time k. Hence:

$$VAR\left[w_i(k)\right] \approx \frac{\Delta_i^2(k)}{12} = \frac{c^2 VAR\left[d_i(k)\right] 2^{-2R_i(k)}}{12}, \tag{4.7}$$

where $VAR\left[d_i(k)\right]$ is the prediction error variance, and it is calculated below. If one increases the value of c, probability that a specific realization of the random variable $d_i(k)$ falls into the quantizer range increases. But, by (4.7) the variance of the quantization error also increases. Clearly, there is a tradeoff between c and error variance. This parameter will be tweaked numerically in the Section 7.

The optimum linear prediction coefficients in the mean squared sense are:

$$a_i^{(k)} = v_{z_i(k)}^T M_{\tilde{z}_i(k-1)}^{-1}, \tag{4.8}$$

where, for $l, m = 1, \ldots, p \leq k$:

$$\begin{aligned} \left[M_{\tilde{z}_i(k-1)}\right]_{lm} &= E\{\tilde{z}_i(k-l)\tilde{z}_i(k-m)\}, & (4.9) \\ \left[v_{z_i(k)}\right]_m &= E\{z_i(k)\tilde{z}_i(k-m)\}. & (4.10) \end{aligned}$$

Hence:

$$VAR\left[d_i(k)\right] = VAR\left[z_i(k)\right] - v_{z_i(k)}^T M_{\tilde{z}_i(k-1)}^{-1} v_{z_i(k)}, \tag{4.11}$$

and, plugging (4.11) into (4.7), we can calculate variance of quantization error for a given node i and iteration k.

4.2.1 Analytical Framework

To be able to compute $\left[M_{\tilde{z}_i(k-1)}\right]$ and $\left[v_{z_i(k)}\right]$, we need to calculate $E\{\tilde{z}_i(k-l)\tilde{z}_i(k-m)\}$ and $E\{z_i(k)\tilde{z}_i(k-l)\}$ for $k, l \in \{1, \ldots, p\}$. Either terms can be obtained taking the ii element of the cross-covariance matrices for a given k and l:

$$\begin{aligned} E\{\tilde{z}_i(k-l)\tilde{z}_i(k-m)\} &= \left[E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}\right]_{ii} \\ E\{z_i(k)\tilde{z}_i(k-m)\} &= \left[E\{z(k)\tilde{z}^T(k-m)\}\right]_{ii} \end{aligned}$$

which are easier to calculate because the recursions are more compact to express in terms of vectors. Note that the noisy recursion has the following simple form:

$$x(k) = (I - \epsilon diag(A1))x(k-1) + A\tilde{z}(k-1), \qquad (4.12)$$

where $\tilde{z}(k-1) = z(k-1) + w(k-1)$ by eq.(4.5). Therefore:

$$\begin{aligned} z(k) &= \epsilon x(k) = (I - \epsilon diag(A1))z(k-1) + \epsilon A\tilde{z}(k-1), \\ &= (I - \epsilon diag(A1))z(k-1) + \epsilon Az(k-1) + \epsilon Aw(k-1) \\ &= Wz(k-1) + \epsilon Aw(k-1) \qquad (4.13) \end{aligned}$$

The simple expression in eq.(4.13) will be used to express the state covariance matrix in a recursive fashion.

Further details on the calculation of $E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}$ and $E\{z(k)\tilde{z}^T(k-m)\}$ are given in Appendix A and B. We define state and noise vector covariances as follows:

$$\begin{aligned} \Sigma(k-m) &\triangleq E\{z(k-m)z^T(k-m)\} \\ \Upsilon(k-m) &\triangleq E\{w(k-m)w^T(k-m)\} \end{aligned}$$

$E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}$ is presented in Table 4.1 for a given k, m, l triplet in terms of the state and noise vector covariances, W, A, and ϵ. Similarly, $E\{z(k)\tilde{z}^T(k-m)\}$ can be written in terms of these quantities as:

$$E\{z(k)\tilde{z}^T(k-m)\} = W^m\Sigma(k-m) + \epsilon W^{m-1}A\Upsilon(k-m) \qquad (4.14)$$

The values of $\Sigma(k-m)$, and $\Upsilon(k-m)$, that change with the index $k-m$, can be calculated in an iterative fashion. In fact, using eq.(4.13), we can express state

Table 4.1: Cross Correlations of Noisy States

	$E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\} =$
$l = m$	$\Sigma(k-m) + \Upsilon(k-m)$
$l < m$	$W^{m-l}\Sigma(k-m) + \epsilon W^{m-l-1}A\Upsilon(k-m)$
$l > m$	$\Sigma(k-l)W^{l-m} + \Upsilon(k-l)\epsilon W^{l-m-1}A$

covariances in terms of *known* previous state covariances and noise covariances as follows:

$$\Sigma(k) = W\Sigma(k-1)W^T + \epsilon^2 A\Upsilon(k-1)A^T \tag{4.15}$$

Moreover, by plugging (4.11) into (4.7), and assuming noises are independent spatially, the noise covariance matrix can also be written as:

$$[\Upsilon(k-1)]_{ii} = \frac{c^2 VAR[d_i(k-1)]2^{-2R_i(k)}}{12} \tag{4.16}$$

and $[\Upsilon(k-1)]_{ij} = 0$ if $i \neq j$.

4.3 Algorithm Summary

At iteration k; node i

- obtains the linear predictor coefficients $a_i^{(k)}$, the prediction error variance $VAR[d_i(k)]$, and the quantization interval length $\triangle_i(k)$,
- quantizes and transmits prediction error $d_i(k)$,
- updates state and noise covariances matrices for the next iteration by (4.15).

Receiving transmitted value, neighbor j

- obtains the linear predictor coefficients $a_i^{(k)}$,

- reconstructs $z_i(k)$ as $\tilde{z}_i(k)$,
- updates state and noise covariances matrices for the next iteration by (4.15).

Once, transmissions are completed by all nodes, state values are updated by (3.1).

Remark 3. *If the network connectivity* (W), *initial distribution of the observations* $(\Sigma(0))$, *and the rate allocation* $(R_i(k))$ *are fixed, then the prediction coefficients can be calculated offline and stored at each sensor. In this case, each sensor stores* $(n_i + 1) \times p \times K$ *values where* n_i *is the number of neighbors of sensor* i, p *is the predictor order, and* K *is the maximum number of iterations. Assuming* p *and* K *are fixed as reasonable quantities, the significant complexity factor will be the number of neighbors. If the network is dense, this factor scales as* $log(n)$ *under the connectivity constraints. Though this results in scalability issues, if the network is dense enough it can be modeled in a more structured way. In this case, the same coefficient can be used for all neighbors, and each node stores only* $p \times K$ *coefficients. Detailed discussions are given in Section 6.2.*

4.4 Asymptotic Rate Behavior

In this section, we show that our proposed scheme requires zero rate asymptotically. To prove it, we will first introduce Lemma 3.

Lemma 3. *The system converges to a consensus if and only if* $\sigma^2_{d_i}(k) \to 0$ *as* $k \to \infty$ *where* $d_i(k) = z_i(k) - \hat{z}_i(k)$.

Proof. Forward: If $\sigma^2_{d_i}(k) \to 0$, then the node values converge to a consensus.

Define $\sigma^2_{w_i(k)} \triangleq E\{w_i^2(k)\}$. If $\sigma^2_{d_i(k)} \to 0$, then

$$\sigma^2_{w_i(k)} = \frac{c^2}{12} 2^{-R_i(k)} \sigma^2_{d_i(k)} \to 0.$$

Reverse: If the nodes values converge to a consensus, then $\sigma^2_{d_i}(k) \to 0$.

Assume that the nodes converge. Then by Lemma 1 $E\{w(k)w^T(k)\} \to \mathbf{0}$ and, by Corollary 1 $\Sigma(k) \to \Sigma^\star$.

By (4.14),

$$\begin{aligned} E\{z(k)\tilde{z}^T(k-m)\} &= W^m E\{z(k-m)z^T(k-m)\} \\ &+ \epsilon W^{m-1} A E\{w(k-m)w^T(k-m)\} \end{aligned}$$

Then, $E\{z(k)\tilde{z}(k-m)\} \to W^m\Sigma^\star + \mathbf{0} = \Sigma^\star;\ \forall\, m \in \{1,\ldots,p\}$. Thus, $[v_{z_i(k)}]_m \to [\Sigma^\star]_{ii}$. By Table 4.1,

$$E\{\tilde{z}(k-l)\tilde{z}(k-m)\} \to \Sigma^\star + \mathbf{0} = \Sigma^\star$$

Thus, $[M_{\tilde{z}_i(k-1)}]_{lm} \to [\Sigma^\star]_{ii};\ \forall\ \ l,m \in \{1,\ldots,p\}$.

At step k, and sensor i, equation (4.8) can be rewritten as:

$$M_{\tilde{z}_i(k-1)} a_i^{(k)} = v^T_{z_i(k)} \tag{4.17}$$

As $k \to \infty$, all entries of M matrix and v vector converges to $[\Sigma^\star]_{ii}$ as shown above. Therefore, (4.17) has infinitely many *MMSE* solutions. We pick $a_i^{(k)} = [1\ 0 \ldots 0]^T$. In other words, $d_i(k) = z_i(k) - \tilde{z}_i(k-1)$. Then,

$$\begin{aligned} \sigma^2_{d_i(k)} &\to \sigma^2_{z_i(k)} - \frac{\left(E\{z_i(k)\tilde{z}_i(k-1)\}\right)^2}{E\{\tilde{z}_i(k-1)\tilde{z}_i(k-1)\}} \\ &= \Sigma^\star_{ii} - \frac{(\Sigma^\star_{ii})^2}{\Sigma^\star_{ii}} \\ &= 0 \end{aligned}$$

□

Lemma 4. *All rate allocations that allow the node values to converge to a common value will eventually converge to zero rate under predictive coding scheme.*

Proof. By Lemma 1 and Lemma 3, the nodes converge to a common value if and only if $\sigma^2_{w_i(k)}$ and $\sigma^2_{d_i(k)} \rightarrow 0$. By (4.7), $R_i(k)$ can be chosen as any non-negative number. Then, $R_i(k) \rightarrow 0 \quad \forall i \in \{1, \ldots, n\}$. □

By Lemma 4, we have showed that under the predictive coding scheme, a consensus can be achieved with zero asymptotic rate. While we do not provide minimum rates for a given MSE deviation(eq. 3.11), we propose a constructive method to allocate quantization rates. If the quantization rates are chosen such that $\sigma^2_{w_i}(k)$ sequence converges to 0 for all sensor i, then a consensus will be achieved. More interestingly, if the rates are chosen such that $\sum_{k=0}^{\infty} \sigma^2_{w_i}(k) < \infty$ for all sensor i, then not only a consensus will be achieved but also the MSE deviation from the initial average will be bounded. We define the ratio of consecutive noise variances at sensor i and iteration k as; $\frac{\sigma^2_{w_i}(k+1)}{\sigma^2_{w_i}(k)} = \eta_i(k+1)$. For instance, for $\eta_i(k) = \left(\frac{k}{k+1}\right)^{\beta}$ and $\beta > 1$,

$$\sum_{k=1}^{\infty} \sigma^2_{w_i}(k) = \zeta(\beta)$$

where $\zeta(.)$ is a Riemann-Zeta function and finite. Under these conditions, a consensus will be achieved with bounded MSE deviation and zero asymptotic rate. The behavior of the quantization rates for a given β is simulated in Section 7.

4.5 Peer to Peer Predictive Coding

At a given time instant k, sensor i knows not only its own previously quantized values $(\tilde{z}_i(k-1), \tilde{z}_i(k-2), \ldots)$, but also neighbors' previously quantized values, i.e. $\tilde{z}_j(k-1), \tilde{z}_j(k-2), \ldots$ where j is a neighbor of i. If the link is bidirectional the prediction error at node i can be further reduced by having:

$$\hat{z}_i(k) = \sum_{l=1}^{p} a_i^{(k)}(l)\tilde{z}_i(k-l) + b_i^{(k)}(l)\tilde{z}_j(k-l). \tag{4.18}$$

where $a_i^{(k)}$ and $b_i^{(k)}$ are corresponding LMMSE coefficients. The derivations related to this method follow exactly the same logic steps in Section 4.2. For brevity, the detailed calculations are given in Appendix C.

Chapter 5

Coding With Side Information

In this chapter, we exploit the fact that a node can use its own sequence of present and past values locally, all or in part, as *side information* ([20],[21]) and therefore we can improve the accuracy in the reconstruction of the neighbors' state in a way that is comparable to predictive coding. To this end, we note that the calculations done in Section 4.2.1 provide the covariance matrices of the current states and the cross covariance of current states with previous states of all nodes. We propose the use of a simple scalar nested lattice code to utilize the side information. We analyze two schemes which are the *broadcast* and the *peer to peer* versions of the strategy.

5.1 Background on WZ Scalar Quantization

Coding with side information or Wyner-Ziv (WZ) coding is the encoding strategy that leads to reduced rate or improved performance by relying on the fact that the decoder can make use of side information correlated with the incoming message. In the classical WZ scheme, there is a single decoder-encoder pair and the decoder has data that can be represented as a noisy version of the data to encode. The rate distortion for this problem was studied by Wyner and Ziv in [20]. Our implementation of WZ will be based on a *nested lattice code* construction (see e.g. [22] and [23]). We will use the simple encoding approach discussed in [24] and [25] that we summarize next.

The decoder has access to the side information $y \in \mathbb{R}$. The encoder compresses its observation $x \in \mathbb{R}$ in the following way (see Fig.5.1 which shows an example of a scalar nested lattice code with rate=1 bit, and quantization interval length $\triangle$):

- The encoder partitions the observation space into disjoint uniform quantization intervals of length $\triangle$ and determines the index of the interval to which x belongs to. (e.g. $3\triangle$ in Fig.5.1)
- The encoder assigns a label l to each of the quantization intervals where $l \in \{1, \ldots, 2^R\}$ and R represents the transmission rate.
- The encoder transmits the label(coset index) of the interval to which x belongs to.(e.g. bin 1 in Fig.5.1)

The reader should note that there are infinitely many intervals which are assigned the same *coset index* by the encoder in this scheme. Assuming that the coset index is available at the decoder error free, the decoder uses the side information (y) and the coset index (bin 1) to determine which one is the most likely interval in which x falls among those that have the same *coset index.* A model often used to represent the dependency between x and y is that $y = x + n$ where n is random noise, with a unimodal zero mean distribution. In this case a minimum distance rule can be used to decide the quantization interval where x is. Then, the quantized value of x is recovered as the centroid of this interval (e.g. $\hat{x} = \triangle$ in Fig.5.1). The formal algorithm is given as follows:

- Given an integer number $P = 2^R$, and quantization index $\triangle$, x is encoded as:

  $$p = \left[\frac{x}{\triangle}\right] \; mod \; P, \tag{5.1}$$

  where *mod* stands for the modulo operation that returns a number $p = 0, ..., P - 1$ and the brackets indicate the rounding operation.

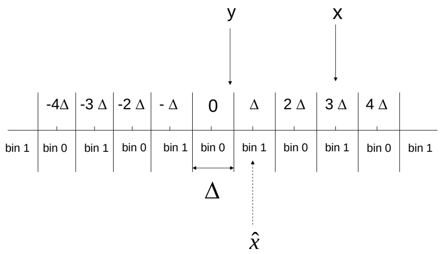


Figure 5.1: An example of nested lattice codes in 1-D

- The coset index p is transmitted to the decoder error free.
- Using p and side information y, the decoder reconstructs x as [25]:

$$\hat{x} \;=\; \left[\frac{\frac{y}{\triangle} - p}{P}\right] P\triangle + p\triangle \tag{5.2}$$

Since the rate is fixed by the communication constraint, the only parameter to choose in the encoder/decoder pair specified in eqs.(5.1) and (5.2) is the quantization step $\triangle$. The choice of the quantization step is non trivial due to its complex effect on the actual distortion that one obtains with a finite nested lattice code, such as our scalar quantizer. In fact, the decoding rule in (5.2) can produce errors having much larger range than the quantization step $\triangle$ and the expression of the error variance is a non linear function of the rate and the step size. The error variance expression for this scheme is given in [24]. It is valid under the assumptions of jointly Gaussian random variables x and y and high resolution. The encoded data x and side information y are such that $y = x + n$ where n is also a Gaussian random variable and independent of x. In the scalar case, the expression is:

$$E\{(x - \hat{x})^2\} \;=\; \Phi(R, \triangle, \sigma_n) \tag{5.3}$$

$$= 2^{2R}\triangle^2 \sum_{k=0}^{\infty}(2k+1)Q\big(\frac{(k+\frac{1}{2})2^R\triangle}{\sigma_n}\big) + \frac{\triangle^2}{12}$$

where σ_n is the variance of n, $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\frac{t^2}{2}}dt$, and $\triangle > 0$. Although the expression is non linear function of the quantization interval $\triangle$, it is convex therefore it has a minimum. Moreover, it can be minimized numerically for a given choice of rate and σ_n with respect to $\triangle$.

Considering Gaussian random states, we propose to use the methodology in (5.1) and (5.2) with a step size chosen to minimize (5.3) to encode the states $z_i(k)$ making use of the side information at the decoder side. The strategy can be generalized easily to other distributions by modifying (5.3) appropriately. Next, the *broadcast* and the *peer to peer* strategies are discussed in the given order.

5.2 Broadcast WZ Coding

In this case, there is a single encoder and several receivers with heterogenous side information. Recently, Feder et al. considered this problem for discrete sources in [26]. The authors make an analogy between the *degraded broadcast channel capacity* problem and *the broadcast Slepian Wolf* problem and provide achievable broadcast rates under *static* and *streaming* scenarios[5]. In our paper we use a simpler albeit suboptimum encoder: a single encoding scheme optimizing the quantization interval length $\triangle$ to have minimum average error variance at the receivers.

Let us denote the neighbor set of sensor i as N_i. For each receiver $j \in N_i$, we can map quantities x and $y = x + n$ in the previous section as follows:

$$z_i(k) \quad \rightarrow \quad x \tag{5.4}$$

[5] At the moment no extension exists for the WZ result and the scheme proposed in [26] requires a rather complex procedure.

$$\begin{aligned} \hat{z}_{ji}(k) &= \sum_{l=1}^{p} b_{ji}(l)\tilde{z}_{ji}(k-l) + \sum_{l=0}^{p} c_{ji}(l) z_j(k-l) && (5.5) \\ \hat{z}_{ji}(k) &\rightarrow y && (5.6) \end{aligned}$$

where $b_{ji}(l)$ and $c_{ji}(l)$ are the linear estimation coefficients, $\hat{z}_{ji}(k)$ is the linear estimate of $z_i(k)$ and $\tilde{z}_{ji}(k-l)$ is the reconstruction of $z_i(k-l)$ at sensor j.

Note that, although there is only one encoder used for all neighbors, the reconstruction of the state value $\tilde{z}_{ji}(k-l)$ (and its quantization noise) is different at each receiver due to the heterogonous side information $\hat{z}_{ji}(k)$. To compute the side information the decoder uses the supervector $\gamma_{ji}(k)$:

$$\gamma_{ji}(k) = [z_j(k)\ z_j(k-1) \dots z_j(k-p)\ \tilde{z}_{ji}(k-1) \dots \tilde{z}_{ji}(k-p)]^T, \qquad (5.7)$$

and the covariance matrix and cross-correlation vector:

$$\begin{aligned} M_{ji}(k) &= E\{\gamma_{ji}(k)\gamma_{ji}^T(k)\} && (5.8) \\ v_{ji}(k) &= E\{z_i(k)\gamma_{ji}^T(k)\}. && (5.9) \end{aligned}$$

Then, the side information in (5.5) can be written in a compact form as:

$$\hat{z}_{ji}(k) = v_{ji}^T(k) M_{ji}^{-1}(k)\gamma_{ji}(k). \qquad (5.10)$$

The reader should note that $M_{ji}(k)$ is a $(2p+1)\times(2p+1)$ matrix for a given order p. The upper left $(p+1)\times(p+1)$ block of the matrix contains cross correlations of the states $k, \dots, k-p$ at sensor j and can be calculated by (A.2). The lower right $p \times p$ block is the covariance of the noisy reconstructions of sensor i and is calculated via the recursion shown in Table 4.1. The upper right and the lower left blocks of the matrix are cross correlations between reconstructions of sensor i and states of sensor j and are derived in (4.14). The necessary equations to derive $v_{ji}(k)$ vector are (4.14) and (4.15).

In order to choose the quantizer parameters, both encoder and decoders need to know the statistics of $d_{ji}(k) = z_i(k) - \hat{z}_{ji}(k)$ where $d_{ji}(k)$ is the estimation error of $z_i(k)$ and it is used as the *side information noise* (i.e. $d_{ji}(k) \rightarrow n$). The variance of $d_{ji}(k)$ is necessary to derive $\triangle$:

$$\begin{aligned} \sigma^2_{d_{ji}}(k) &= E\{(z_i(k) - \hat{z}_{ji}(k))^2\} \\ &= VAR[z_i(k)] - v^T_{ji}(k) M^{-1}_{ji}(k) v_{ji}(k) \qquad (5.11) \end{aligned}$$

Using (5.11) we can derive:

$$\triangle_i(k) = argmin_{\triangle > 0} \sum_{j \in N_i} \Phi(R, \triangle, \sigma_{d_{ji}(k)}) \qquad (5.12)$$

which is the common quantization interval length minimizing the average error variance over all $j \in N_i$. Using (5.4), (5.6) and the $\triangle_i(k)$ in (5.12), the encoder i computes the coset index and broadcasts to its neighbors. Upon receiving the coset index error free, each neighbor reconstructs $z_i(k)$ as in (5.2).

Once all the neighbors' information is received and decoded, the sensor i performs the following state update:

$$x_i(k) = (1 - \epsilon \sum_{j=1}^{n} a_{ij}) x_i(k-1) + \sum_{j=1}^{n} a_{ij} \tilde{z}_{ij}(k-1) \qquad (5.13)$$

To write the network equations we introduce $\tilde{Z}(k)$ as a $n \times n$ matrix whose ij entry is reconstruction of $z_j(k)$ by $z_i(k)$. The network equation is given as:

$$x(k) = (I - \epsilon diag(A\mathbf{1})) x(k-1) + (A \odot \tilde{Z}(k-1))\mathbf{1}. \qquad (5.14)$$

where $\odot$ represents entry by entry matrix multiplication and $\mathbf{1}$ is all ones column vector.

Once all sensors update their current states as in (5.13), the new step requires knowing $M_{ji}(k+1)$, and $v_{ji}(k+1)$ defined in (5.8) and (5.9). As mentioned before, these quantities can be written in terms of the state and the noise covariances and calculated recursively as indicated in Section 3.1.1. The only difference compared to Section 3.1.1 is that because each term $\tilde{z}_{ij}(k-1)$ has its own specific quantization error $w_{ij}(k-1)$ associated, the update of the covariance matrix of the states has a different form in this case compared to (4.15). By eqs. (2.5) and (5.14):

$$z(k) = Wz(k-1) + \beta(k-1) \tag{5.15}$$

where $[\beta(k-1)]_i = \epsilon \sum_{j=1}^{n} a_{ij} w_{ij}(k-1)$. Then:

$$\begin{aligned} E\{z(k)z^T(k)\} &= WE\{z(k-1)z^T(k-1)\}W^T \\ &+ E\{\beta(k-1)\beta^T(k-1)\} \end{aligned} \tag{5.16}$$

where, approximating the $w_{ij}(k-1)$ as being spatially uncorrelated:

$$[E\{\beta(k-1)\beta^T(k-1)\}]_{ii} = \epsilon^2 \sum_{j=1}^{n} a_{ij}^2 VAR[w_{ij}(k-1)]$$

and the non-diagonal entries are 0. We note that

$$VAR[w_{ji}(k)] = \Phi(R, \triangle_i(k), \sigma_{d_{ji}(k)}) \tag{5.17}$$

for a given i, j pair, where $\triangle_i(k)$ is given in (5.12).

The procedure described in this section is followed for all k iteratively.

5.3 Asymptotic Rate Behavior

Lemma 5. *The nodes converge to a consensus if and only if* $\sigma^2_{w_{ji}}(k) \to 0 \ \forall \ (j,i)$ *pair.*

Proof. Define $\sigma^2_{w_{ji}}(k) \triangleq VAR[w_{ji}(k)]$. Proof is straightforward from Lemma 1. □

Lemma 6. *Given finite number of sensors ($n < \infty$) and bounded A matrix ($a_{ij} < \infty;\ \forall\ i,j\ \in\ \{1,\ldots,n\}$), mean squared deviation from the initial average is bounded if and only if $lim_{k\to\infty}\sum_{t=0}^{k-1}\sigma^2_{w_{ji}}(t)$ converges $\forall\ i,j \in \{1,\ldots,n\}$.*

Proof. Proof is very similar to of Lemma 2 and omitted. □

Lemma 7. *The nodes converge to a consensus if and only if $\sigma^2_{d_{ji}}(k) \to 0\ \forall\ (i,j)$ pair.*

Proof. Forward: If $\sigma^2_{d_{ji}}(k) \to 0\ \ \forall\ j$, then $\triangle_i(k) \to 0$ in (5.12), since expression becomes a quadratic function of $\triangle$ which is minimized as $\triangle \to 0$. If $\triangle_i(k) \to 0$, then $\sigma^2_{w_{ji(k)}} \to 0$ by (5.17).

Reverse: If the nodes converge, then by Lemma 5 $\sigma^2_{w_{ji}}(k) \to 0$. Then by the proof of Lemma 3, $\sigma^2_{d_{ji}}(k) \to 0$. The similarity is due to the fact that $d_{ji}(k)$ corresponds to the linear predictor error ($d_i(k)$) in Lemma 3. □

Lemma 8. *All rate allocations that allow the node values to converge to a common value will eventually converge to zero rate under WZ coding scheme.*

Proof. By Lemma 5 and Lemma 7, the nodes converge if and only if $\sigma^2_{w_{ji(k)}}$ and $\sigma^2_{d_{ji(k)}} \to 0$. By (5.3), $R_i(k)$ can be chosen as any non-negative number. Then, $R_i(k) \to 0\ \ \forall i \in \{1,\ldots,n\}$. □

5.4 Peer to Peer WZ

This strategy utilizes a different encoder for each neighbor. For each sender-receiver pair (i,j), we define the message to be sent and the side information as in (5.4) and (5.6). We follow the same procedure (5.7-5.11) to calculate predictor coefficients and prediction error. In this case, the optimum quantization interval is given as:

$$\triangle_{ji}(k) = argmin_{\triangle>0}\Phi(R,\triangle,\sigma_{d_{ji(k)}}) \tag{5.18}$$

instead of (5.12). The sensor i encodes $z_i(k)$ as in (5.1) with $\triangle = \triangle_{ji}(k)$. The sensor j reconstructs $z_i(k)$ as $\tilde{z}_{ji}(k)$ via (5.2). The state updates are given as in (5.14). The only difference in this scheme compared to the previous one is that each encoder has a different step size and hence (5.17) should be replaced by

$$VAR[w_{ji}(k)] = \Phi(R, \triangle_{ji}(k), \sigma_{d_{ji}(k)}), \tag{5.19}$$

for a given i, j pair, where $\triangle_{ji}(k)$ is given in (5.18).

Chapter 6

Dense Sensor Networks

6.1 Relaxed Conditions on the Global Knowledge of the Network Connectivity

In this section, we give a discussion about how to relax global knowledge of the network graph under dense sensor network, therefore how to increase the scalability of the proposed schemes. In particular, our derivations will be given for predictive coding method since WZ analysis is quite similar.

Assume that a group of sensors is distributed uniformly over a 2-dimensional square with unit area. In general the area of interest is much larger than unit square, i.e. battlefield or forest, but it can be rescaled for analysis purposes. Transform the square such that it wraps around in both x and y directions. Such a network architecture is called 2-dimensional unit torus and has been used to model connectivity graph for wireless sensor networks ([27],[28]). The main motivation behind the approach is to remove boundary effects. An example with 16 nodes and 2-D is given in Fig. 6.1. On a unit torus nodes 2,4,5 and 13 lie on a radius-r circle whose center is node 1. We assume that a noise-free link exits between any two nodes that are at a range less than connectivity radius r ($[A]_{ij} = [A]_{ji} = 1$ if $d_{ij} < r$).

Remark 4. *For a network with n nodes distributed uniformly on a 2-dimensional unit torus with connectivity radius r ($G^2(n, r)$), the degree of every node is $O(r^2 n)$ for large n.*

The proof of the Remark 4 is straightforward from *Law of Large Numbers.* Con-

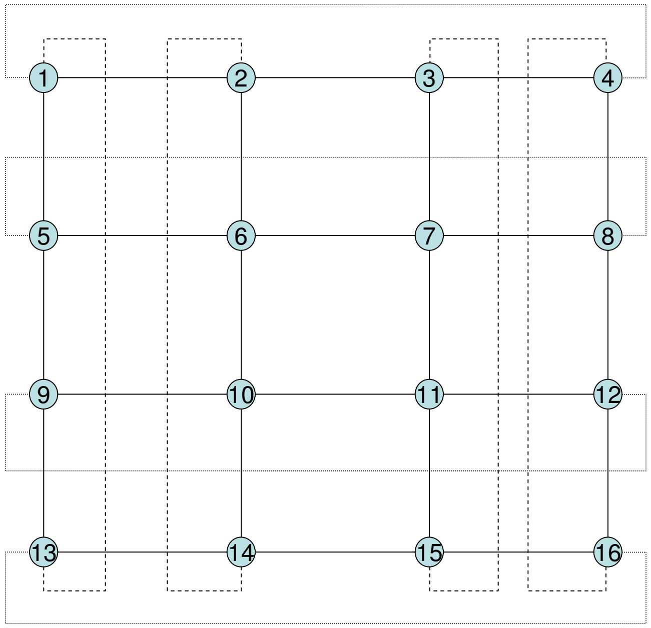


Figure 6.1: Grid representation of a regular network

sequently, for large n each node in the network has the same number of neighbors with high probability and the connectivity graph of the network becomes *regular* yielding more robust mathematical representation. We argue that for large n, connectivity matrix of the network can be reconstructed by each node independently with constraint that the underlying graph is regular.

Definition 1. *A graph isomorphism is a one-to-one and onto mapping between the vertices of the graphs G_1 and G_2 ($F : G_1 \rightarrow G_1$) such that any two vertices u_2 and v_2 are adjacent if and only if u_1 and v_1 are adjacent with relationship $u_2 = f(u_1)$ and $v_2 = f(v_1)$. In other words, given two undirected graphs G_1 and G_2 with adjacency matrices A_1 and A_2, G_1 and G_2 are isomorphic if and only if there exists a permutation matrix P such that $PA_1P^{-1} = A_2$.*

Definition 2. *A matrix P is defined as a permutation matrix if its entries are*

$\{0,1\}$ and, each row and column have exactly one entry as 1. Moreover P matrices are orthogonal, i.e. $PP^T = I$.

Lemma 9. *Given two undirected graphs with $A_1 = PA_2P^{-1}$, the covariance matrices of the consensus algorithm at step k have the following relationship under the uniform rate constraint:*

$$\Sigma_1(k) = P\Sigma_2(k)P^{-1}$$

Proof. By Definition 2, if $A_1 = PA_2P^{-1}$ then $W_1 = PW_2P^{-1}$. We use induction for the rest of the proof. Since G_2 is transformed into a new (isomorphic) graph, the nodes of the graph are renumbered while the edges stay fixed. Then $\Sigma_1(0) = P\Sigma_2(0)P^{-1}$ due to renumbering of the nodes. It follows that $\Gamma_1(0) = P\Gamma_2(0)P^{-1}$ where $\Gamma_i(k)$ is the noise variance matrix of graph i at step k. This is due the fact that initial noise variances are functions of the individual state variances (i.e. $[\Gamma_1(0)]_{ii} = f([\Sigma_1(0)]_{ii})$). We show that relationship holds for $k = 0$. Now, we assume that it holds for $t \in \{1, \dots, k\}$. Then, we show that it holds for $t = k+1$. By Table 1 and equation (4.14):

$$\begin{aligned} E_1\{\tilde{z}(k-l)\tilde{z}^T(k-m)\} &= PE_2\{\tilde{z}(k-l)\tilde{z}(k-m)\}P^{-1} \\ E_1\{z(k)\tilde{z}^T(k-m)\} &= PE_2\{z(k)\tilde{z}^T(k-m)\}P^{-1} \end{aligned}$$

In the above equations we use the fact that $(PBP^{-1})^m = PB^mP^{-1}$. Then,

$$[E_1\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}]_{ii} = [E_2\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}]_{jj}$$

$$[E_1\{z(k)\tilde{z}^T(k-m)\}]_{ii} = [E_2\{z(k)\tilde{z}^T(k-m)\}]_{jj}$$

where $i = f(j)$ and f is the isomorphic mapping from G_2 to G_1. Then, $[\Gamma_1(k)]_{ii} = [\Gamma_2(k)]_{jj}$. Since Γ is a diagonal matrix,

$$[\Gamma_1(k)] = P[\Gamma_2(k)]P^{-1} \tag{6.1}$$

By the update equation;

$$\begin{aligned}
\Sigma_1(k+1) &= W_1\Sigma_1(k)W_1 + \epsilon^2 A_1\Gamma_1(k)A_1 \\
&= PW_2P^{-1}P\Sigma_2(k)P^{-1}PW_2P^{-1} + \epsilon^2 PA_2P^{-1}P\Gamma 2(k)P^{-1}PA_2P^{-1} \qquad (6.2) \\
&= P(W_2\Sigma_2(k)W_2 + \epsilon^2 A_2\Gamma_2(k)A_2)P^{-1} \\
&= P\Sigma_2(k+1)P^{-1}
\end{aligned}$$

where in (6.2) we use (6.1) and induction hypothesis. □

The covariance matrix of the states of a graph at different time instances is the main performance indicator of the system since noise variances are functions of this particular matrix. By Lemma 9, we have shown that if two graphs are isomorphic, the proposed consensus algorithm on these networks will have the same performance under the constraint that same quantization rates are utilized.

Since dense sensor networks can be represented as regular graphs for sufficiently large n, and independent numbering of such a graph results in adjacency matrices which are isomorphic; by Lemma 9, there is no loss of performance with respect to the case where W is known globally.

6.2 The Asymptotic Behavior of Predictor Coefficients

In this section, we show that predictor coefficients ($a_i^{(k)}$) become the same for all sensors i for a given iteration k, therefore complexity of the scheme does not depend on number of neighbors but the predictor order and number of maximum iterations.

A brief background on block circulant matrices with circulant blocks is given, which will be used for analysis purposes.

Definition 3. *A matrix C is a circulant matrix of order n if it is a square matrix of the form:*

$$C = \begin{pmatrix} c_1 & c_2 & \dots & c_n \\ c_n & c_1 & \dots & c_{n-1} \\ \dots & \dots & \dots & \dots \\ c_2 & c_3 & \dots & c_1 \end{pmatrix}$$

Definition 4. *Let $C_1, C_2, \dots, C_n$ be circulant matrices with order n. Then B matrix is block circulant matrices with circulant blocks if:*

$$B = \begin{pmatrix} C_1 & C_2 & \dots & C_n \\ C_n & C_1 & \dots & C_{n-1} \\ \dots & \dots & \dots & \dots \\ C_2 & C_3 & \dots & C_1 \end{pmatrix}$$

Remark 5. *Diagonal entries of block circulant matrices with circulant blocks are equal. Moreover, block circulant matrices with circulant blocks form a commutative algebra, i.e. if A, B are such matrices, so are $A + B$ and $A.B$.*

The connectivity graph of regular grid in 2-dimensions is block circulant with circulant blocks. Then, connectivity graph of a sufficiently dense sensor network becomes block circulant.

Lemma 10. *For a sufficiently dense sensor network where sensors' initial observations are identically distributed, and quantization rate allocation is uniform among the sensors, LMMSE coefficients will be exactly the same for all sensor under predictive coding scheme.*

Proof. By Remark 4, sufficiently dense sensor network can be mapped into a 2-dimensional regular grid whose connectivity matrix is block circulant with circulant blocks. LMMSE coefficients for each sensor were given in (4.8) as:

$$a_i^{(k)} = v_{z_i(k)}^T M_{\tilde{z}_i(k-1)}^{-1}, \tag{6.3}$$

where, for $l, m = 1, \ldots, p \leq k$:

$$\begin{aligned} [M_{\tilde{z}_i(k-1)}]_{lm} &= E\{\tilde{z}_i(k-l)\tilde{z}_i(k-m)\}, & (6.4) \\ \left[v_{z_i(k)}\right]_m &= E\{z_i(k)\tilde{z}_i(k-m)\}. & (6.5) \end{aligned}$$

Furthermore, these quantities could be obtained by taking ii element of the cross-covariance matrices for a given k and l:

$$\begin{aligned} E\{\tilde{z}_i(k-l)\tilde{z}_i(k-m)\} &= \left[E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}\right]_{ii} & (6.6) \\ E\{z_i(k)\tilde{z}_i(k-m)\} &= \left[E\{z(k)\tilde{z}^T(k-m)\}\right]_{ii} & (6.7) \end{aligned}$$

Therefore, necessary and sufficient condition for $a_i^{(k)} = a_j^{(k)}$ for all $\{i, j\}$ pairs and recursion index k is that

$$\begin{aligned} \left[E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}\right]_{ii} &= \left[E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}\right]_{jj} \\ \left[E\{z(k)\tilde{z}^T(k-m)\}\right]_{ii} &= \left[E\{z(k)\tilde{z}^T(k-m)\}\right]_{jj} \end{aligned}$$

$\forall\{i, j\} \in \{1, \ldots n\}$, $l, m \in \{1, \ldots, p\}$ and $k \in \{1, \ldots\}$.

By hypothesis, adjacency matrix of the graph (A), initial covariance matrix of the states and noises $(\Sigma(0), \Upsilon(0))$ are block circulant with circulant blocks. Then, $W = I - \epsilon(diag(A) - A)$ is also block circulant. Since block circulant matrices form a commutative algebra, consecutive state and noise covariance matrices are also block circulant given that quantization rate allocation is uniform among sensors. If W,A, $\Sigma(k)$ and $\Upsilon(k)$ are block circulant for all $k \in \mathbb{N}$, then $\left[E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}\right]$ and $E\{z_i(k)\tilde{z}_i(k-m)\}$ are also block circulant $\forall \;\; l, m \in \{1, \ldots, p\}$. Since diagonal entries of a block circulant matrix are equal, $a_i^{(k)} = a_j^{(k)}$ for all $\{i, j\}$ pairs and recursion index k. □

By Lemma 10, we have shown that the prediction coefficients at each sensor will be the same therefore the number of neighbors of a given sensor is not an indicative of the complexity. In other words, the scheme is asymptotically scalable in terms of the number of coefficients computed and stored at each iteration.

To sum up, in Section 6.1 we have shown that each sensor can independently label the network and reconstruct the adjacency matrix without losing any performance with respect to the case where each node has access to the same adjacency matrix. In Section 6.2, we have proved that prediction coefficients are the same for all nodes, which makes the scheme more robust from scalability point of view.

Chapter 7

Simulation Results

7.1 Performance Analysis

In this chapter, we show the numerical performance of the *broadcast* algorithms proposed, illustrating their convergence characteristics in the network scenario defined next. *Peer to peer* schemes are not included in the section, since their performance is naturally better than *broadcast*. For simulation purposes, several random geometric graphs $G(n, r)$ are generated with nodes uniformly distributed over a unit square. According to the definition of $G(n, r)$, there exists a link between any two nodes if their range is less than r. We assume that there are no channel errors between two nodes that are connected. Moreover, we assume that matrix A in (2.1) is such that:

$$a_{ij} = \left\{ \begin{array}{ll} a_{ij} = 1, & \text{if there is an edge between node } i \text{ and } j \\ a_{ij} = 0, & \text{if } i = j \text{ or } i \neq j \text{ and there is no edge} \end{array} \right\} \tag{7.1}$$

A sample random network with ten nodes and connectivity 0.4 is shown in Fig.7.1. Nodes are represented by vertexes, and link are represented by edges. First, we investigate the behavior of the quantization rate and compare it among three different schemes: broadcast, Wyner-Ziv and simple coding. Fig.7.2 shows the behavior of the average number of quantization bits per sensor over 60 iterations. The simulation parameters are $r = 0.4$, $n = 10$. The data at each sensor is initialized as zero mean unit variance Gaussian random variable. Moreover, the initial transmission rate is chosen as 0. Then, at each iteration the quantization rates are chosen such that $\frac{\sigma^2_{w_i}(k)}{\sigma^2_{w_i}(k-1)} = \left(\frac{k-1}{k}\right)^{\beta}$. It is true that for any choice of $\beta > 1$, quantization noise will form a convergent p-series, thus nodes will achieve

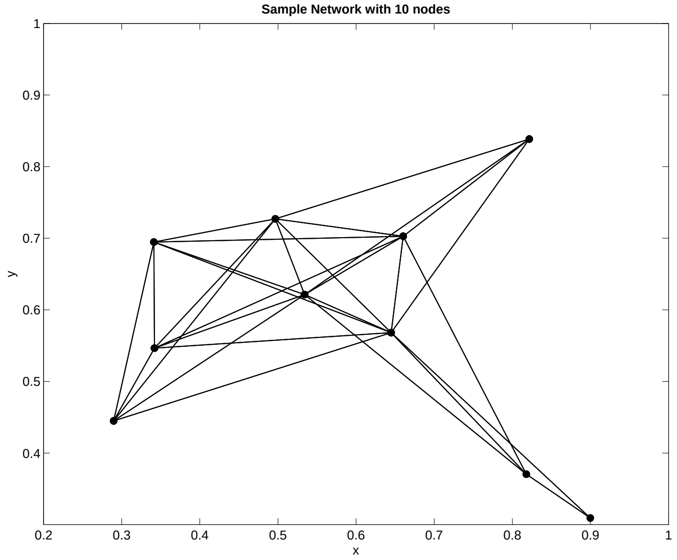


Figure 7.1: A sample random network with $r = 0.4$, $n = 10$

bounded convergence. For simulation purposes, β is chosen to be 5 in Fig.7.2. Under these conditions, simple coding requires a non-decreasing rate, and it tends to converge to a non-zero rate. On the other hand, our proposed schemes require smaller rates after a certain number of iterations and rates converge to 0 as number of iterations go to infinity. For this specific simulation, WZ requires greater rates in the beginning, but converges to 0 faster than predictive coding. We would like to emphasize that the average number of quantization bits per sensor per iteration converges to 0 in the proposed schemes, whereas in the simple coding method, it will converge to a non-zero constant. Fig.7.3 shows the evolution of the sensor states as the algorithm iterates under different methods. We assume that we are allowed to transmit at a rate which is equal to the channel capacity at each iteration($R = 5$ bits per sensor). All algorithms are initialized by the same zero mean unit variance Gaussian random data. There are 10 nodes in the network, and

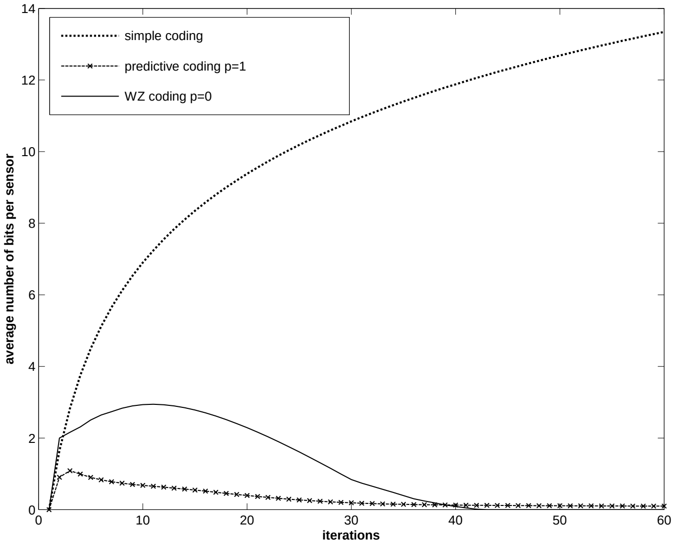
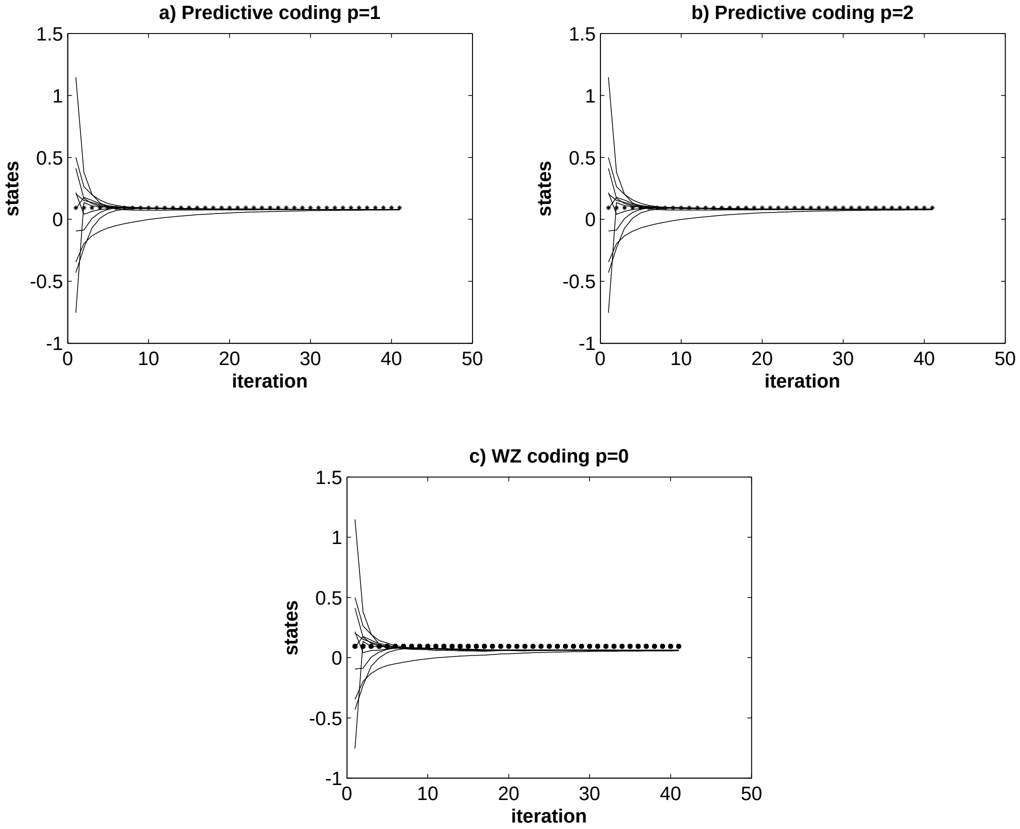


Figure 7.2: Rate demand for consensus($r = 0.4$, $n = 10$,iteration$= 60$)

the connectivity radius is 0.5. Figs.7.3a and b show predictive coding with order 1, and 2. We do not observe any significant improvement by increasing the prediction order p. In fact, for $p >= 2$ the covariance matrix becomes badly scaled giving rise to numerical errors in the computation of the prediction coefficients. Fig.7.3 c shows WZ coding with $p = 0$. All three methods converge to a consensus which is very close to the initial average (indicated with $*$). Fig.7.4 shows the behavior of the average mean squared deviation from the initial average per iteration, defined as:

$$MSE(k) = \frac{1}{n}\sum_{i=1}^{n}\left(z_i(k) - \frac{1}{n}\sum_{i=1}^{n} z_i(0)\right)^2 .$$

In other words, $MSE(k)$ is the average mean squared distance of the states at iteration k from the initial mean. The algorithms are simulated through 1000 monte carlo initial state values, for a maximum of 400 iterations of the average consensus algorithm and averaged over 13 different random geometric graphs. There are 10



a) Predictive coding p=1

b) Predictive coding p=2

c) WZ coding p=0

Figure 7.3: State evolution($n = 10$,$R = 5$,$r = 0.5$)

network nodes and the connectivity radius is uniformly distributed in $(0, 1)$ interval. Fig.7.4 shows that quantizing without taking into account of the increasing correlations among the states (i.e. simple coding) results in increasing and unbounded error variance as discussed by Boyd et al.[16](dotted line). Instead, it appears that if we utilize the correlation structure of the data exchanged as in our paper, the error variance converges to a finite value (solid line), irrespective of the iteration number. Fig.7.5 shows the rate-distortion performance of predictive coding($p = 1$) and WZ ($p = 0$). The results are averaged over 1000 monte carlo

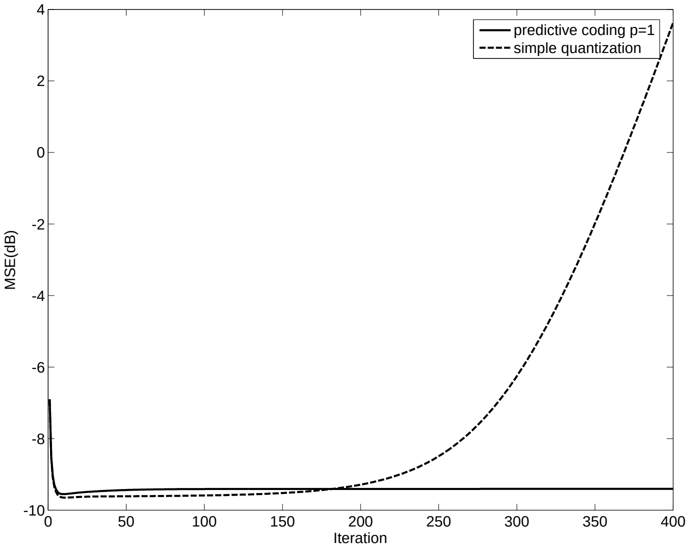


Figure 7.4: MSE versus iteration($n = 10$,$R = 0.1$)

runs under a single random network topology and 50 iterations. Rates represent average number of bits per sensor per iteration. We note that given rate-distortion performances are only valid for 50 iterations, since as the number of iterations increase average transmission rates are going to decrease, and vanish as iterations go to infinity. But we are aware that in practical communication systems one can only have a finite number of iterations. Thus, we provide a sample rate-distortion performance for one such scenario. While it is clear that the predictive coding method performs better, the WZ scheme also has reasonably good performance. This result is promising in the following way: One may argue that due to memory and lifetime constraints of battery powered wireless networks, storing and processing previous values may not be feasible. In this case, one can set $p = 0$ in the WZ coding strategy (no previous values but current states only reaching reasonably good performance).

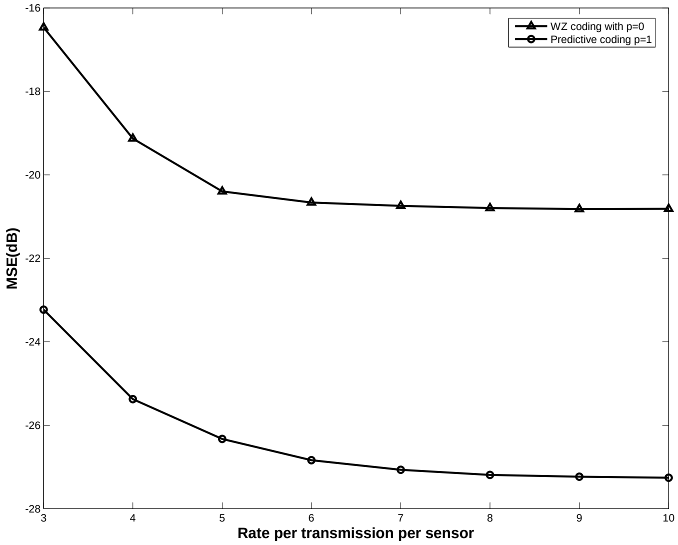


Figure 7.5: MSE versus iteration($n = 10$,$R = 0.1$)

7.2 An Example Practical Application

In this section, we consider a practical application of consensus algorithm. Assume that there are n agents on the field that can only communicate locally with their neighbors. They travel at the same speed but towards different directions. The aim of these agents is to move into the average of the initial direction. In this scenario, adjacency matrix of the network is dynamic since as the agents move outside of the connectivity radius of some neighbors, connectivity will break down. We note that the agents may be troops in a battlefield trying to move to the same direction, UAV's flying in a formation, or satellites trying to beam into a single point.

We assume that there are total of 10 agents and they communicate at the rate of 5 bits per iteration. Fig. 7.6 shows the movement of the agents without any communication. Each agents initialize its direction independently and follows its

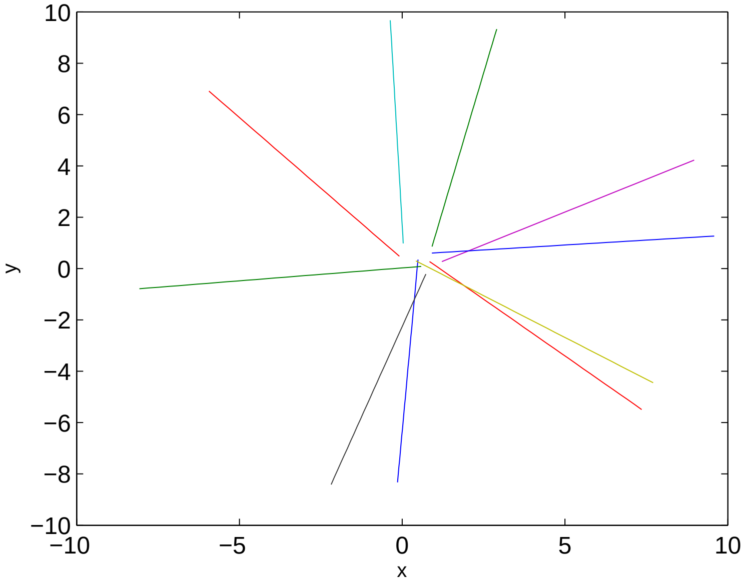


Figure 7.6: Agent movement without communication

own course. Fig. 7.7 shows the movement when the agents communicate with their neighbors. The connectivity radius is 0.3. Each node can hear the nodes which are closer than 0.3 units in Euclidean distance metric. It can be seen that nodes whose initial headings are towards *negative* directions, turn and follow a positive bearing. While seven agents converge to the same direction, three of them follow a different course. Connectivity radius is not large enough so three agents lose communication with the group and get lost. We can increase the connectivity radius to make sure that agents converge to the same heading before they lose communication. Fig. 7.8 is the result of the simulation when connectivity radius is 5 units. It may be seen that the agents follow the same heading which is very close to the average of the initial directions.

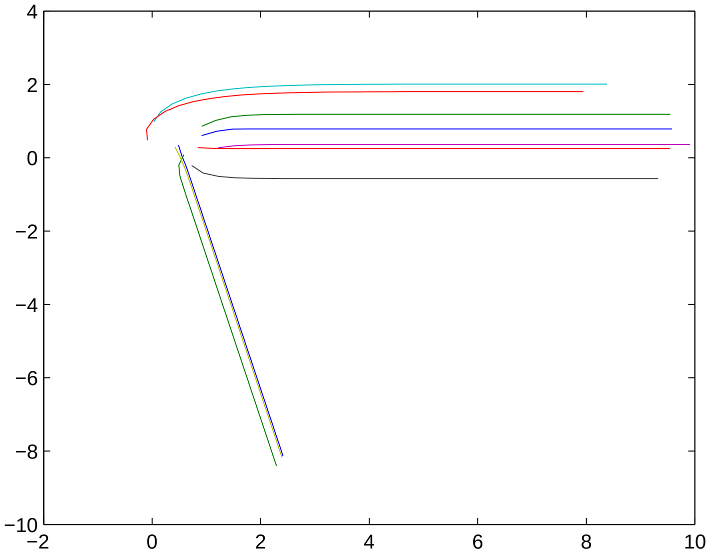


Figure 7.7: Agent movement with connectivity radius 0.3

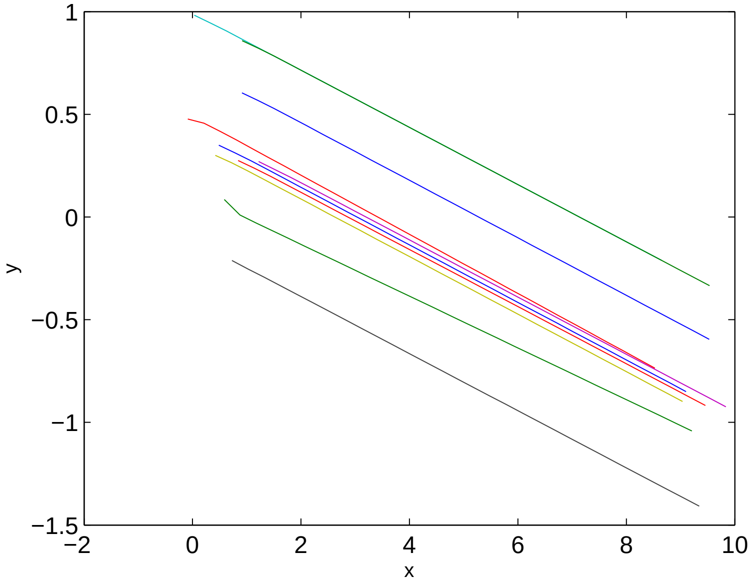


Figure 7.8: Agent movement with connectivity radius 5

Chapter 8

Discussion

We have explored conditions on the quantization noise variances which are required for convergence of the nodes in the *average consensus* problem. We have also studied bounded convergence,where the mean squared distance between the final node values and the initial mean of the nodes is bounded. We have proposed and investigated two source coding strategies which satisfy bounded convergence constraints with zero average rate asymptotically. We have given the mathematical framework for predictive coding, and nested lattice coding for both *peer to peer* and *broadcast* scenarios, providing the details on how to implement each strategy. Our most significant contribution is that we have shown that even with highly suboptimal encoding strategies using the temporal correlation and the ever increasing spatial correlation, bounded consensus can be achieved with zero asymptotic rate. In other words, our proposed schemes lead to a non increasing MSE divergence from the initial mean as the number of iterations increases. Hence, this decreases the sensitivity of the algorithm precision relative to the convergence speed of the algorithm itself. On the other hand this also highlights that the accuracy gained by increasing the number of iterations saturates with these simple approaches.

We analyze the behavior of the schemes under the dense sensor network assumption without boundary effects. We have shown that network connectivity matrix of such a network can be reconstructed by each node independently without any performance losses with respect to the case where each node has access to the same connectivity matrix. Moreover, we have shown that predictor coefficients at each sensor are the same and we have removed drawback mentioned in Remark

3. We conclude that the schemes presented are scalable in the context of wireless sensor networks.

Moreover, the comparison of our strategies leads to the interesting observation that predictive coding strategy performs better but WZ scheme has a promising performance even with $p = 0$.

Appendix A

Calculation of $E\{\tilde{z}_i(k-l)\tilde{z}_i(k-m)\}$

We have already known that $E\{\tilde{z}_i(k-l)\tilde{z}_i(k-m)\} = [E\{\tilde{z}(k-l)\tilde{z}(k-m)\}]_{ii}$. Covariance among noisy states at time instant l and time instant k can be written as:

$$\begin{aligned} E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\} &= E\{z(k-l)z^T(k-m)\} \\ &+ E\{z(k-l)w^T(k-m)\} \\ &+ E\{w(k-l)z^T(k-m)\} \\ &+ E\{w(k-l)w^T(k-m)\}. \end{aligned}$$

We also note that $z(k)$ vector can be written in terms of $z(k-1-q)$ as follows:

$$\begin{aligned} z(k) &= Wz(k-1) + \epsilon Aw(k-1) \\ &= W^{q+1}z(k-1-q) + \epsilon \sum_{j=0}^{q} W^j Aw(k-1-j). \end{aligned} \tag{A.1}$$

We will focus on the first term of covariance equation. Suppose $l \leq m$; by eq. A.1 correlation between states can be expressed as:

$$\begin{aligned} E\{z(k-l)z^T(k-m)\} &= E\{z(k-m+(m-l))z^T(k-m)\} \\ &= W^{m-l}E\{z(k-m)z^T(k-m)\} \\ +\epsilon \sum_{j=0}^{m-l-1} & W^j AE\{w(k-l-1-j)z^T(k-m)\} \\ &= W^{m-l}E\{z(k-m)z^T(k-m)\} \end{aligned} \tag{A.2}$$

We note that each element of the summation term is zero since vector $z(k-m)$ is independent of future noise vectors and both state and noise vectors have zero mean. In the case of $l > m$, state correlation can be written as;

$$E\{z(k-l)z^T(k-m)\} = E\{z(k-l)z^T(k-l)\}(W^{l-m})^T$$

Now, we will focus on the second equation. Suppose $l < m$. Then,

$$\begin{aligned} & E\{z(k-l)w^T(k-m)\} \\ = \; & E\{z(k-m+(m-l))w^T(k-m)\} \\ = \; & W^{m-l}E\{z(k-m)w^T(k-m)\} \\ + \; & \epsilon \sum_{j=0}^{m-l-1} W^j AE\{w(k-l-1-j)w^T(k-m)\} \\ = \; & \epsilon W^{m-l-1}AE\{w(k-m)w^T(k-m)\} \end{aligned}$$

We realize that $z(k-m)$ is independent of the noise $w(k-m)$, and noises vectors at different time instants are independent. In the case of $l \geq m$,

$$E\{z(k-l)w^T(k-m)\} \;=\; 0$$

By the same way, the third term can be given as

$$E\{w(k-l)z^T(k-m)\} \;=\; 0$$

if $l \leq m$. In the case of $l > m$,

$$\begin{aligned} & E\{w(k-l)z^T(k-m)\} = \\ & E\{w(k-m)w^T(k-m)\}\epsilon\big(W^{m-l-1}A\big)^T \end{aligned}$$

The last term, which is the covariance of the error terms, is zero unless $l = m$ since noises at different time instants are independent.

If we are to summarize the formulas above in three different cases; If $l = m$

$$\begin{aligned} E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\} \;=\; & E\{z(k-m)z^T(k-m)\} \\ + \; & E\{w(k-m)w^T(k-m)\} \end{aligned}$$

If $l < m$

$$\begin{aligned} E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\} \;=\; & W^{m-l}E\{z(k-m)z^T(k-m)\} \\ + \; & \epsilon W^{m-l-1}AE\{w(k-m)w^T(k-m)\} \end{aligned}$$

If $l > m$

$$\begin{aligned} E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\} &= E\{z(k-l)z^T(k-l)\}(W^{l-m})^T \\ &+ E\{w(k-l)w^T(k-l)\}\epsilon\big(W^{m-l-1}A\big)^T \end{aligned}$$

Moreover, $E\{\tilde{z}_i(k-l)\tilde{z}_i^T(k-m)\}$ can be found by taking ii entry of the $E\{\tilde{z}(k-l)\tilde{z}^T(k-m)\}$ matrix.

Appendix B

Calculation of $E\{z_i(k)\tilde{z}_i^T(k-m)\}$

In this section, we will calculate correlation between state vector at time k, and noise vector at time $k-m$. We know that $E\{z_i(k)\tilde{z}_i(k-m)^T\} = E\{z(k)\tilde{z}(k-m)\}_{ii}$. Then,

$$\begin{aligned} E\{z(k)\tilde{z}(k-m)^T\} &= E\{z(k)\big(z^T(k-m)+w^T(k-m)\big)\} \\ &= W^m E\{z(k-m)z^T(k-m)\} \\ &+ \epsilon W^{m-1} A E\{w(k-m)w^T(k-m)\} \end{aligned}$$

In the above derivation, we use the equations in Appendix A by setting $l=0$, and $m \geq 1$.

Appendix C

Derivation of the Separate Encoder

We start by redefining the equations given in Section 4.2. Assume that sensor i is to transmit sensor j where the link is bidirectional. Define a vector γ_{ji} as:

$$\gamma_{ji}(k) = [\tilde{z}_{ji}(k-1) \ \ldots \ \tilde{z}_{ji}(k-p) \ \tilde{z}_{ij}(k-1) \ \ldots \ \tilde{z}_{ij}(k-p)]^T \tag{C.1}$$

where $\tilde{z}_{ji}(k)$ is the reconstruction of $z_i(k)$ at sensor j, and $\tilde{z}_{ij}(k)$ is the reconstruction of $z_j(k)$ at sensor i at time k. By modifying (4.1), the predictor of $z_i(k)$ at the sensor j can be expressed as;

$$\hat{z}_{ji}(k) = a_{ji}^{(k)}\gamma_{ji}(k) \tag{C.2}$$

where $a_{ji}^{(k)} = v_{z_{ji}(k)}^T M_{\tilde{z}_{ji}(k-1)}^{-1}$. For $l, m \leq 2p$, we define:

$$[M_{\tilde{z}_{ji}(k-1)}]_{lm} = \left\{ \begin{array}{ll} E\{\tilde{z}_{ji}(k-l)\tilde{z}_{ji}(k-m)\} & l, m \leq p \\ E\{\tilde{z}_{ji}(k-l)\tilde{z}_{ij}(k-m \ mod \, p)\} & l \leq p < m \\ E\{\tilde{z}_{ij}(k-l \ mod \, p)\tilde{z}_{ji}(k-m)\} & m \leq p < l \\ E\{\tilde{z}_{ij}(k-l \ mod \, p)\tilde{z}_{ij}(k-m \ mod \, p)\} & m, l \geq p \end{array} \right\} \tag{C.3}$$

$$[v_{z_{ji}}(k)]_m = \left\{ \begin{array}{ll} E\{z_i\tilde{z}_{ji}(k-m)\} & m \leq p \\ E\{z_i\tilde{z}_{ij}(k-m \ mod \, p)\} & m > p \end{array} \right\} \tag{C.4}$$

C.1 Detailed Mathematical Analysis

We define $n^2 \times 1$ vector s as:

$$s(k) = Hz(k) + \omega(k) \tag{C.5}$$

Table C.1: M Matrix

	$[M_{\tilde{z}_{ji}(k-1)}]_{lm} =$
$l, m \leq p$	$E[s(k-l)s^T(k-m)]_{(i+(j-1)n, i+(j-1)n)}$
$l \leq p < m$	$E[s(k-l)s^T(k-m\ modp)]_{(i+(j-1)n, j+(i-1)n)}$
$m \leq p < l$	$E[s(k-l\ modp))s^T(k-m)]_{(j+(i-1)n, i+(i-1)n)}$
$m, l \geq p$	$E[s(k-l\ mod\ p)s^T(k-m\ mod\ p)]_{(j+(i-1)n, j+(i-1)n)}$

z(k) is the state vector, H is $n^2 \times n$ matrix of the form

$$H = \begin{bmatrix} I \\ I \\ \dots \\ I \end{bmatrix} \tag{C.6}$$

I is $n \times n$ identity matrix and,

$$\omega(k) = \begin{bmatrix} w_{11} \\ w_{12} \\ w_{13} \\ \dots \\ w_{1n} \\ w_{21} \\ \dots \\ w_{nn} \end{bmatrix} \tag{C.7}$$

where $w_{ji}(k)$ is the additive noise in the reconstruction of $z_i(k)$ at sensor j and has the characteristics which are discussed in Section 4.2. It is clear that $s(k)$ vector contains all possible state reconstructions. We can express the entries of $[M_{\tilde{z}_{ji}(k-1)}]$ in terms of $s(k)$ as in Table C.1. Now, we revisit the noisy recursion formula of the $z(k)$ vector. If we focus on the ith entry of the vector $z(k)$,

$$z_i(k) = \sum_{j=1}^{n} w_{ij} z_j(k-1) + \epsilon \sum_{j=1}^{n} a_{ij} w_{ij}(k-1) \tag{C.8}$$

Then, (4.13) can be modified as:

$$z(k) = Wz(k-1) + \epsilon\beta(k-1) \tag{C.9}$$

where

$$[\beta(k-1)]_i = \sum_{j=1}^{n} a_{ij} w_{ij}(k-1) \tag{C.10}$$

We note that state recursions have rather simple form also in this case. Moreover,

$$z(k) = W^{q+1} z(k-q-1) + \epsilon \sum_{j=0}^{q} W^j \beta(k-1-j) \tag{C.11}$$

C.2 Calculation of M matrix

To be able to calculate $[M_{\tilde{z}_{ji}(k-1)}]_{lm}$, we need to calculate $E[s(k-l)s^T(k-m)]$ for a given $l, m \leq p$ pair. By (C.5):

$$\begin{aligned} E\{s(k-l)s^T(k-m)\} &= HE\{z(k-l)z^T(k-m)\}H^T \\ &+ HE\{z(k-l)\omega^T(k-m)\} \\ &+ E\{\omega(k-l)z^T(k-m)\}H^T \\ &+ E\{\omega(k-l)\omega^T(k-m)\}. \end{aligned}$$

Let's focus on the first term. If $l \leq m$,

$$\begin{aligned} HE\{z(k-l)z^T(k-m)\}H^T &= HE\{W^{m-l}z(k-m)z^T(k-m)\}H^T \\ &+ HE\{\sum_{j=0}^{m-l-1} W^j \beta(k-l-1-j) z^T(k-m)\}H^T \\ &= HW^{m-l}E\{z(k-m)z^T(k-m)\}H^T \end{aligned}$$

since noise is independent of states. If $l \geq m$,

$$HE\{z(k-l)z^T(k-m)\}H^T = HE\{z(k-l)z^T(k-l)\}(W^{l-m})^T H^T$$

If we focus on the second term, for $l \geq m$

$$HE\{z(k-l)\omega^T(k-m)\} = 0$$

If, $l < m$, then

$$\begin{aligned} HE\{z(k-l)\omega^T(k-m)\} &= HE\{W^{m-l}z(k-m)... \\ &+ \epsilon \sum_{j=0}^{m-l-1} W^j \beta(k-l-1-j) \quad \omega^T(k-m)\} \\ &= \epsilon H W^{m-l-1} E\{\beta(k-m)\omega(k-m)\} \end{aligned}$$

where $E\{\beta(k-m)\omega(k-m)\}$ has a rather nice form, i.e.

$$[E\beta(k-m)\omega(k-m)]_{ij} = a_{ij}^2 VAR[w_{ij}(k)]$$

Third term is very similar to second term. If $l \leq m$

$$HE\{\omega(k-l)z^T(k-m)\} = 0$$

If $l > m$,

$$HE\{\omega(k-l)z^T(k-m)\} = E\{\beta(k-l)\omega(k-l)\}(\epsilon H W^{m-l-1})^T$$

The last term, $E\{\omega(k-l)\omega^T(k-m)\} = 0$ if $m \neq l$. If $m = l$,

$$E[\{\omega(k-l)\omega^T(k-l)\}]_{(i(n-1)+j, i(n-1)+j)} = VAR[w_{ij}(k-l)]$$

where non-diagonal entries are 0. Now, we will investigate $v_{z_{ji}}(k)$ vector. We can rewrite eq. C.4

$$[v_{z_{ji}}(k)]_m = \left\{ \begin{array}{ll} E[z(k)s^T(k-m)]_{(i,n(j-1)+i)} & m \leq p \\ E[z(k)s^T(k-m \ mod \ p)]_{(i,n(i-1)+j)} & m > p \end{array} \right\} \tag{C.12}$$

and

$$E[z(k)s^T(k-m)] = \epsilon H W^{m-1} E\{\beta(k-m)\omega(k-m)^T\}$$

We will finalize the section by calculating state variance matrix in a recursive fashion. Using eq. C.9,

$$
\begin{aligned}
E\{z(k)z^T(k)\} &= WE\{z(k-1)z^T(k-1)\}W^T && \text{(C.13)}\\
&+ \epsilon^2 E\{\beta(k-1)\beta^T(k-1)\} && \text{(C.14)}
\end{aligned}
$$

where

$$E[\beta(k-1)\beta^T(k-1)]_{ii} = \sum_{j=1}^{n} a_{ij}^2 VAR[w_{ij}(k-1)]$$

and non-diagonal entries are 0.

BIBLIOGRAPHY

[1] T. Vicsek, A. Czirok, E. B. Jacob, I. Cohen, and O. Schoche, "Novel type of phase transitions in a system of self-driven particles," *Phys. Rev. Lett.*, vol. 75, pp. 1226–1229, 1995.

[2] C. M. Breder, "Equations descriptive of fish schools and other animal aggregations," *Ecology*, pp. 361–370, 1954.

[3] A. Okubo, "Dynamical aspects of animal grouping: Swarms, schools, flocks, and herds," *Adv. Biophys*, vol. 22, pp. 1–94, 1986.

[4] A. Czirok, A. L. Barabadi, and T. Vicsek, "Collective motion of self-propelled particles: Kinetic phase transition in one dimension," *Phys. Rev. Lett.*, vol. 82, pp. 209–212, 1999.

[5] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. on Automatic Control*, vol. 48, pp. 988–1001, 2003.

[6] J. Lin and A. S. Morse, "The multi-agent rendezvous problem," in *IEEE Conf. on Decision and Control Proc.*, December 2003, pp. 1508–1513.

[7] A. Mostefaoui, M. Raynal, and F. Tronel, "The best of both worlds:a hybrid approach to solve consensus," in *International Conference on Dependable Systems Proc.*, June 2000, pp. 513–522.

[8] H. Seba, N. Badache, and A. Bouabdallah, "Solving the consensus problem in a dynamic group:an approach suitable for a mobile environment," in *International Symposium on Computers and Communications Proc.*, July 2002, pp. 327–332.

[9] W. Weigang, Y. J. C. Jiannong, and M. Raynal, "A hierarchical consensus protocol for mobile ad hoc networks," in *Euromicro International Conference on Parallel, Distributed, and Network-Based Processing Proc.*, Feb 2006, p. 9.

[10] T. Arai, E. Yoshida, and J. Ota, "Information diffusion by local communication of multiple mobile robots," in *International Conference on Systems, Man and Cybernetics Proc.*, October 1993, pp. 535 – 540.

[11] R. O. Saber and R. M. Murray, "Agreement problems in networks with directed graphs and switching topology," *California Institute of Technology Technical Report*, vol. CIT-CDS 03-005, 2003.

[12] W. Ren, R. W. Beard, and D. B. Kingston, "Multi-agent kalman consensus with relative uncertanity," in *American Control Conference Proc.*, OR, USA, June 2005, pp. 1865 – 1870.

[13] M. Rabbat, R. Nowak, and J. Bucklew, "Generalized consensus computation in networked systems with erasure links," in *IEEE Workshop on Signal Processing Advances in Wireless Communications Proc.*, June 2005, pp. 1088–1092.

[14] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Gossip algorithms: Design, analysis and applications," in *INFOCOM Proc.*, 2005.

[15] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, pp. 65–78, 2004.

[16] L. Xiao, S. Boyd, and S. Kim, "Distributed average consensus with least-mean-square deviation," in *Mathematical Theory of Networks and Systems Proc.*, Kyoto,Japan, July 2006, pp. 158–167.

[17] L. W. Beineke and R. J. Wilson, *Topics in Algebraic Graph Theory.* U.S.A.: Cambridge, 2004.

[18] B. Widrow, I. Kollar, and M. Liu, "Statistical theory of quantization," *IEEE Trans. on Instrumentation and Measurement*, vol. 45, pp. 353–361, 1996.

[19] A. Leon-Garcia and I. Widjaja, *Communication Networks.* U.S.A.: McGrawHill, 2000.

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

[21] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 471–480, July 1973.

[22] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (discus): design and construction," in *Data Compression Conference Proc.*, March 1999, pp. 158–167.

[23] R. Zamir and S. Shamai, "Nested linear/lattice codes for wyner-ziv encoding," in *IEEE Inform. Theory Workshop Proc.*, Killarney,Ireland, 1998, pp. 158–167.

[24] Z. Lin, S. Cheng, A. D. Liveris, and Z. Xiong, "Slepian-wolf coded nested quantization(swc-nq) for wyner-ziv coding: Performance analysis and code design," in *Data Compression Conference Proc.*, 2004, pp. 322–331.

[25] A. Vosughi and A. Scaglione, "Precoding and decoding paradigms for distributed data compression,," *IEEE Transactions on Signal Processing*, 2007.

[26] M. Feder and N. Shulman, "Source broadcasting with unknown amount of receiver side information," in *IEEE Proc. of Information Theory Workshop Proc.*, Banglore,India, 2005, pp. 127–130.

[27] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Transactions on Information Theory*, vol. 46, pp. 388–404, March 2000.

[28] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, “Mixing times for random walks on geometric random graphs,” in *Workshop on Analytic Algorithms and Combinatorics Proc.*, January 2005.