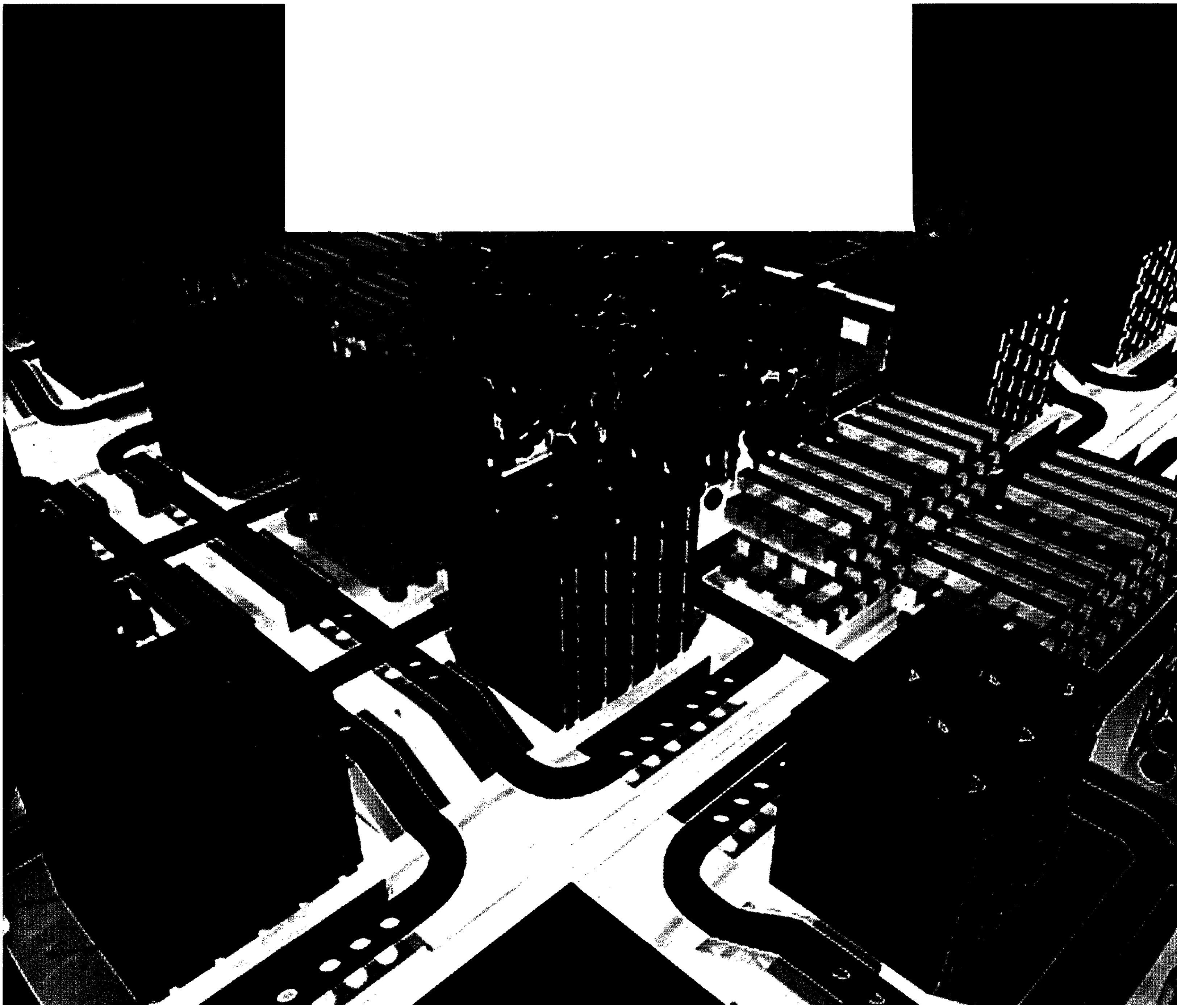
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

The RESEARCH LABORATORY *of* ELECTRONICS



Successive Structuring of Source Coding Algorithms for Data Fusion, Buffering, and Distribution in Networks

By: Stark Christiaan Draper

RLE Technical Report No. 661

June 2002

Successive Structuring of Source Coding Algorithms for Data Fusion, Buffering, and Distribution in Networks

Stark Christiaan Draper

B.S. Electrical Engineering, B.A. History
Stanford University

S.M. Electrical Engineering and Computer Science
Massachusetts Institute of Technology

Submitted to the Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering and Computer Science

June 2002

© 2002 Massachusetts Institute of Technology
All Rights Reserved

Author: ______________________________
Department of Electrical Engineering and Computer Science
24 May 2002

Certified by: ______________________________
Gregory W. Wornell
Professor of Electrical Engineering
Thesis Supervisor

Accepted by: ______________________________
Arthur C. Smith
Professor of Electrical Engineering
Chairman, Department Committee on Graduate Studies

Successive Structuring of Source Coding Algorithms for Data Fusion, Buffering, and Distribution in Networks

Stark Christiaan Draper

Submitted to the Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology on 24 May 2002 in partial fulfillment of
the requirements for the degree of Doctor of Philosophy

Abstract

Numerous opportunities to improve network performance present themselves when we make communication networks aware of the characteristics of the data content they are handling. In this thesis, we design such content-aware algorithms that span traditional network layers and are successively structured, focusing on problems of data fusion, buffering, and distribution. The successive structuring of these algorithms provides the flexibility needed to deal with the distributed processing, the heterogeneous sources of information, and the uncertain operating conditions that typify many networks.

We investigate the broad interactions between estimation and communication in the context of data fusion in tree-structured sensor networks. We show how to decompose any general tree into serial (pipeline) and parallel (hub-and-spoke) networks. We develop successive coding strategies for these prototype sensor networks based on generalized Wyner-Ziv coding. We extend Wyner-Ziv source coding with side information to "noisy" encoder observations and develop the associated rate-distortion function. We show how to approach the serial and parallel network configurations as cascades of noisy Wyner-Ziv stages. This approach leads to convenient iterative (achievable) distortion-rate expressions for quadratic-Gaussian scenarios. Under a sum-rate constraint, the parallel network is equivalent to what is referred to as the CEO problem. We connect our work to those earlier results. We further develop channel coding strategies for certain classes of relay channels.

We also explore the interactions between source coding and queue management in problems of buffering and distributing distortion-tolerant data. We formulate a general queuing model relevant to numerous communication scenarios, and develop a bound on the performance of any algorithm. We design an adaptive buffer-control algorithm for use in dynamic environments and under finite memory limitations; its performance closely approximates the bound. Our design uses multiresolution source codes that exploit the data's distortion-tolerance in minimizing end-to-end distortion. Compared to traditional approaches, the performance gains of the adaptive algorithm are significant – improving distortion, delay, and overall system robustness.

Thesis Supervisor: Gregory W. Wornell
Title: Professor of Electrical Engineering

Acknowledgments

I have had the good fortune to interact with and learn from some amazing people at M.I.T. At the top of this heap is my adviser Professor Greg Wornell, my teacher, mentor, collaborator, and friend. I feel extremely fortunate to have observed and learned from Greg's unique combination of sharp technical insight, broad perspectives, quiet wisdom, and wit. His continuing confidence in our work was critical, and I look forward to our future collaborations.

I would like to thank my thesis readers, Professors Robert Gallager and Alan Willsky, for their suggestions and thoughtful criticisms that make their mark throughout the thesis.

Chance (or fate?) served up the fortuitous meeting with Professor Alan Oppenheim that inspired me to come to M.I.T. In the ensuing years I have interacted with Al as a student, teaching assistant, and colleague. As a student I witnessed Al's grace in the classroom, and as a teaching-assistant I learned that such grace is achieved through a lot of hard work. Outside the classroom I continue to benefit from Al's vision, interest in his students, and his friendship.

To my colleagues in DSPG and LIDS, past and present, I would like to thank you for the hallway chats, coffee runs to Tosci's, and serious discussions at the Muddy. Thanks to the faculty and students of 6.962 for the opportunity to help create a great new learning environment. Professor David Forney played a crucial role in encouraging and establishing this new class. I deeply enjoyed working with Dave in that context, and learning from his engineering insight. Certainly, without the valuable collaborations and camaraderie of Richard Barron, Aaron Cohen, Yonina Eldar, Nick Laneman, and Emin Martinian, my experiences at M.I.T. would have been far less rich and rewarding.

The generous financial support from an Intel Corporation Fellowship; Hewlett-Packard under the MIT/HP Alliance; the National Science Foundation under Grant No. CCR-073520; the Microcircuits Advanced Research Corporation under the Center for Circuits, Systems, and Software; Microsoft Research; and Texas Instruments under the Leadership Universities Program; are gratefully acknowledged.

Some influences pre-date my arrival at M.I.T. For teaching me how to write (which seems particularly relevant at the moment), for supervising my first thesis, and for so many other life experiences, I want to express my gratitude to my history advisers at Stanford, Professors (Uncle) Wayne Vucinich and Norman Naimark. My parents provided continuous encouragement and, fortunately for me, an additional technical perspective. Finally, a loving thanks to my fianceé Michelle Ransom for aiding the cause and taking a chance on a disheveled refugee. I look forward to our many years of living, loving, and adventuring together.

For my grandfathers
Petraq and Charles

who inspired my curiosity
in history and science

Contents

Abstract 3

Acknowledgments 4

1 Introduction 13
- 1.1 Data Fusion in Sensor Networks 14
- 1.2 Buffering and Distribution of Distortion-Tolerant Data 17
- 1.3 Thesis Outline 19

2 Background: Using Side Information 21
- 2.1 Source Coding and Side Information 21
  - 2.1.1 Scalar Quantization with Side Information 22
  - 2.1.2 Lossless Vector Source Coding with Side Information 24
  - 2.1.3 Lossy Vector Source Coding with Side Information 26
  - 2.1.4 Quadratic-Gaussian Case 28
- 2.2 Channel Coding with Side Information 30
- 2.3 Dualities between Source and Channel Coding Strategies 32
- 2.4 Chapter Summary 33

3 Side Information Problems with Noisy Encoder Observations 35
- 3.1 Noisy Source Coding with Side Information 35
  - 3.1.1 Binary-Hamming Case 37
  - 3.1.2 Quadratic-Gaussian Case 38
- 3.2 Noisy Channel Coding with Side Information 39
  - 3.2.1 Quadratic-Gaussian Case 41
- 3.3 Geometric Pictures 42
  - 3.3.1 Noisy Wyner-Ziv Coding 42
  - 3.3.2 Noisy Information Embedding 43
- 3.4 Chapter Summary 44

4 Successively Structured Data Fusion Algorithms for Sensor Networks 47
- 4.1 System Model: Sensor Network, Finite-Rate Communications 47

4.2 Literature Review: CEO Problems . . . 51
4.3 Estimate-and-Quantize: A Network Layered Approach . . . 52
4.3.1 Parallel Network . . . 52
4.3.2 Serial Network . . . 53
4.3.3 Comparison . . . 54
4.3.4 Kalman Filtering with Rate Constraints . . . 54
4.4 Inter-Layer Approaches that use Decoder Side Information . . . 56
4.5 Serial CEO Problem: Quadratic-Gaussian Case . . . 57
4.5.1 Iterative Distortion-Rate Expression . . . 58
4.5.2 Design Implications . . . 60
4.5.3 Growing the Bit Pipe Logarithmically . . . 61
4.6 Parallel CEO Problem: Quadratic-Gaussian Case . . . 62
4.6.1 Iterative Distortion-Rate Expression . . . 62
4.6.2 Design Implications . . . 64
4.6.3 Special Cases and Connections to Earlier CEO Results . . . 65
4.7 Application to the Relay Channel . . . 71
4.8 Chapter Summary . . . 73

5 Queuing with Distortion-Control **75**
5.1 Introduction . . . 75
5.2 Fundamentals . . . 76
5.3 System Model: Finite-Memory Buffers, Multiresolution Source Codes . . 77
5.3.1 Lower Bound on Distortion — All Algorithms . . . 79
5.4 Algorithm Design . . . 79
5.4.1 Baseline Algorithm: A Network Layered Approach . . . 80
5.4.2 Inter-Layer Adaptive Algorithms with Distortion-Control . . . 82
5.5 Algorithm Analysis . . . 87
5.5.1 Steady State Performance of Baseline Algorithm . . . 87
5.5.2 Steady State Performance of Adaptive Algorithm . . . 90
5.6 Comparison of Algorithms . . . 90
5.6.1 Memory-Constrained Region . . . 92
5.6.2 Communication-Constrained Region . . . 94
5.6.3 Section Summary . . . 95
5.7 Design Implications . . . 97
5.8 Chapter Summary . . . 98

6 Conclusions **101**
6.1 Future Work . . . 102

A Derivations: Noisy Wyner-Ziv Coding **105**
A.1 Finite-Alphabet Rate-Distortion Function . . . 105
A.1.1 Convexity . . . 106
A.1.2 Converse . . . 107

A.1.3 Achievability . . . 107
A.2 Binary-Hamming Case . . . 109
A.3 Quadratic-Gaussian Case . . . 113

B Derivations: Noisy Information Embedding 117
B.1 Finite-Alphabet Capacity Expression . . . 117
B.1.1 Converse . . . 118
B.1.2 Achievability . . . 118
B.2 Quadratic-Gaussian Case . . . 119

C The Serial Markov Lemma 123
C.1 Introduction . . . 123
C.2 $\epsilon(n)$-Strong Typicality . . . 124
C.3 Approach 1: The Dithered Encoding Rule . . . 126
C.3.1 Proof Set-up . . . 127
C.3.2 Convergence of the Mean $E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right]$ . . . 128
C.3.3 Bounding the Growth of $\mathrm{var}(N(x_0, z_0; \mathbf{x}, \mathbf{z}))$ . . . 128
C.3.4 Mean-Square Convergence . . . 130
C.4 Approach 2: Viswanathan-Berger (VB) Codes . . . 131
C.4.1 Introduction to VB Codes . . . 131
C.4.2 Proof Set-up . . . 132
C.4.3 Convergence of Mean $E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right]$ . . . 133
C.4.4 Bounding the Growth of $\mathrm{var}(N(x_0, z_0; \mathbf{x}, \mathbf{z}))$ . . . 134
C.4.5 Mean-Square Convergence . . . 137

D Derivations: CEO Problems 139
D.1 Serial CEO Problem . . . 139
D.1.1 Achievability . . . 139
D.1.2 Quadratic-Gaussian Case . . . 140
D.2 Parallel CEO Problem . . . 141
D.2.1 Achievability . . . 141
D.2.2 Quadratic-Gaussian Case . . . 144
D.3 Parallel CEO Problem with Large Numbers of Agents . . . 144

List of Figures 148

List of Tables 153

Notation 155

Bibliography 158

Chapter 1

Introduction

Layered architectures underlie the design of many communication networks. Layering is a form of hierarchical modularity that allows the processes at each layer to view the functionality of lower layers as a black box with a defined set of inputs and outputs. Within this design paradigm, each layer's functional modules can be designed relatively independently of the other layers, constrained only to standard inter-layer interfaces. This greatly simplifies overall system design. Not surprisingly, however, such design simplicity comes at a price.

In this thesis, we investigate network applications where substantial performance gains can be realized by designing algorithms that work across traditional network layers. In particular, we focus on designing algorithms for data fusion, buffering, and content distribution in networks. We approach them as problems of joint source and channel coding. To understand better the motivations for the designs we present, we next consider how these problems would be approached in a layered architecture.

In a layered network architecture the source and channel coding aspects of these problems are separated. In many cases the transmission of any information source is decomposed into two stages. The first stage removes redundancy by *source coding* the information signal into a bit stream. The second stage *channel codes* the resulting bit stream to introduce structured redundancy that can correct for transmission errors. Source coding is generally carried out in the application layer at the top of the network protocol stack, and channel coding in the physical layer at the bottom. This functional separation is shown in Fig. 1.1. In certain point-to-point communication problems, and in certain limiting regimes (such as no constraints on decoding delay), employing such a decomposition does not necessarily incur any loss in performance. However, in many situations, such as the network situations we consider, this is not the case. For these problems, overall system performance can be improved by designing algorithms that work across traditional networking layers, requiring the joint design of source and channel codes. We term these "inter-layer" algorithms.

In this thesis we design inter-layered approaches to data fusion, buffering, and distribution. Our designs share two important perspectives: they are "content-aware" and "successively structured". While in layered architectures source coding converts any information source into an undifferentiated stream of bits, in inter-layered designs the network is able to exploit more detailed understanding of the characteristics of the

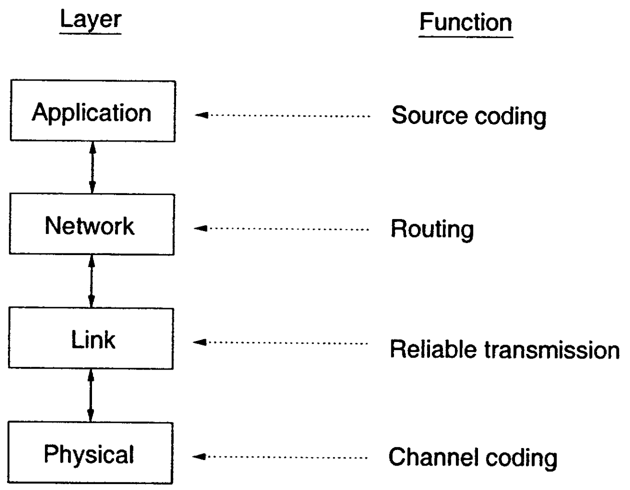


Figure 1.1. An abbreviated diagram of the protocol stack in a traditional layered architecture, focusing on the layers relevant to the work in this thesis.

data it is handling. When networks have this more detailed knowledge, we term them "content-aware". The second perspective – successive structuring of algorithms – is useful because of the distributed nature of networks. As the information sources in a network are distributed, it is necessary to design algorithms that work without having access to all the information sources in any location. Successively structured algorithms work well in such contexts. They have the added advantage that they are flexible enough to deal with the uncertain operating conditions and heterogeneous sources of information that typify many networks. In the rest of this chapter, we illustrate the utility of these perspectives by introducing the problems of study and describing the characteristics of networks that make these problems challenging.

■ 1.1 Data Fusion in Sensor Networks

Consider the sensor network depicted in Fig. 1.2. The black node represents the n-length random source vector $\mathsf{x} = x^n$ that is observed at a number of sensor nodes (represented by open circles). In addition to making a measurement, each node can communicate to one other sensor node at a finite rate. For example, node 1 measure y_1 and communicates nR_1 bits (a rate of R_1 bits per observation sample) to node 3. The goal of the network is to get a particular node the best possible approximation of the source signal. This node is termed the "CEO" – the Chief Estimation Officer. In keeping with CEO terminology we often refer to sensor nodes as "agents". The tools of classical estimation theory cannot be applied directly to this problem because the observations are not co-located. The distributed nature of the data turns this into a joint problem of estimation and communication. By focusing on successively structured algorithms, our results will effectively generalize classic sequential estimation problems

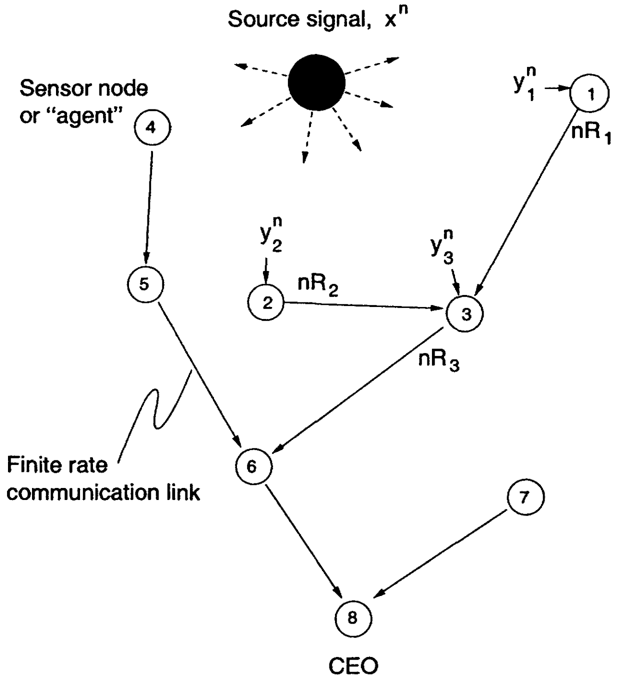


Figure 1.2. A general sensor network with finite rate links and tree-structured communications.

(such at the Kalman Filter) to finite rate communication constraints.

As in Fig. 1.2, our focus will be on tree-structured communication strategies. This means that all information flow is uni-directional, from the outermost nodes (the "leaves" of the tree – nodes 1,2,4,7 in the figure) to the CEO (the "root" of the tree – node 8 in the figure). We concentrate on tree-structured communications because they are more easily analyzed than more general communications that include loops. Loops increase the complexity of the data fusion problem because care must be taken not to double-count information at the end of each loop. The complications introduced by loops arise in other estimation problems. One example is in inference using graphical models [44, 77, 85]. In these problems, probabilistic dependencies between random variables are encapsulated by the topology of a graph. If the dependencies are structured like a tree, then iterative algorithms such as Belief Propagation [49] are guaranteed to converge to the correct inferences. Such convergence is not guaranteed for graphs with cycles. A second example is in some information theoretic problems of communication with feedback, such as the multiple-access channel with feedback [48, 23]. In these problems, loop-like information dependencies are introduced by the feedback, often making these problems difficult to analyze.

The model for data fusion in networks we have introduced resonates in a number of research communities. Many researchers have looked into problems of detection with distributed sensors (see, e.g., [69, 73, 76, 11] and the references therein). In terms of Fig. 1.2, the CEO's (or perhaps the "Chief Decision Officer's") job would be to make a decision about the source, rather than an estimate of it. One early piece of work that illustrates the failure of the separation principle in this network context is by Tenney and Sandell [69]. They discuss a binary hypothesis testing problem with two sensors. Under each hypothesis, the sensors observe independent, but identically distributed, Gaussian random variables; the mean of the observations is determined by the true hypothesis. Each sensor is able to send a single bit of information to the decision maker. The authors show that even in this simple case, the decision rules at each sensor need to be optimized jointly rather than individually, making this a joint detection-communication problem.

Much of the work in the distributed detection literature concentrates on scalar observations ($n = 1$) and finite *sized* codebooks. Researchers in the information theory community have also investigated detection problems under vector observations ($n > 1$) and finite *rate* codebooks [61, 1, 90, 39]. A related area of research in information theory is multiterminal source coding. In these problems, the CEO's job is not to make a decision based on $\mathbf{x}$, but rather to estimate all the observations $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_L$. This vein of research was initiated by Slepian and Wolf [67] for the lossless encoding of a distributed pair of correlated source signals. Their elegant solution motivated many extensions, both lossless [82, 40] and lossy [74, 8, 87]. To our knowledge, the full solution to the latter remains unsolved.

The difference between multiterminal source coding and CEO data fusion problems is that in the latter, the CEO is interested only in the source signal $\mathbf{x}$. The CEO has no specific interest in the sensor observations $\mathbf{y}_1, \ldots, \mathbf{y}_L$ other than in how they help in the estimation of $\mathbf{x}$. Therefore, CEO problems are particular instances of more general problems of estimation under communication constraints. As we will discuss later on in more depth, one characteristic of CEO problems is that there is no constraint on decoding delay, so large block sizes can be exploited during encoding and decoding. In more general problems of estimation under communication constraints there may be constraints on decoding delay. This is the case in situations where source estimates are being used for feedback control [68, 58]. A second characteristic of CEO problems is that all agents observe noisy versions of the same constant source $\mathbf{x}$. In more general estimation problems the source may actually evolve as a function of l. In such situations the CEO's objective would be to make a sequence of estimates, $\hat{\mathbf{x}}[1], \ldots, \hat{\mathbf{x}}[L]$. Clearly, the field of estimation under communication constraints is quite rich.

Problems similar to the CEO problems we present arise in the context of ad hoc sensor networks [70, 43, 37, 38] as well as in more strict network coding contexts [53, 51]. The introduction of CEO data fusion problems to the information theory community occurred relatively recently [9]. The original finite-alphabet CEO problem was extended to continuous-alphabets in [75, 45]. In all of these papers [9, 75, 45] the network configuration considered consisted of a set of agents that independently communicated to the CEO in a single step (e.g., nodes 6,7, and 8 in Fig. 1.2). The scenario depicted in Fig. 1.2 that we consider in this thesis is an extension of this model to general tree configurations.

One of the main contributions of this thesis is in developing connections between (a) the data fusion model proposed in the CEO literature [9, 75, 45], and (b) the coding ideas proposed in the source coding with side information literature [81, 2, 84, 83]. Bringing these two veins of work together yields some attractive successive coding structures for the coupled problems of estimation and communications for sensor networks on trees. In Chapter 2 we set the stage for our discussion of data fusion problems by presenting the ideas and insights from the coding with side information literature in some depth.

To complete this introduction of the data fusion problem, we now describe in broad terms some of the questions addressed in this thesis. Consider again the sensor network model of Fig. 1.2. At node 3 there are three sources of information: (a) the node's own observation $\mathbf{y}_3$, (b) a message m_1 from node 1, and (c) a message m_2 from node 2. The observation $\mathbf{y}_3$ may be continuous or discrete, depending on the scenario, but the messages are always discrete indices $m_1 \in \{1, 2, \ldots, 2^{nR_1}\}$ and $m_2 \in \{1, 2, \ldots, 2^{nR_2}\}$. One question we investigate is how to fuse these heterogeneous sources of information together. We will see that because we allow length-n block encoders and decoders, joint design of the source encoding, communication, and data fusion steps will yield substantial performance gains over decoupled designs.

We also discuss the design implications our results have on choosing the configuration of network communications. For instance, it may be better for node 2 to communicate to node 5, rather than to node 3. We determine some design rules to help choose the best communications tree leading to the best estimate. On the other hand, we may be given a fixed communications tree, but have the flexibility to assign resources (such as rate) differentially to the various nodes of the tree. We determine some design rules to help make such resource assignments.

1.2 Buffering and Distribution of Distortion-Tolerant Data

For problems of data buffering and distribution, we can again design network algorithms that work across protocol layers. We focus on networks handling distortion-tolerant data, i.e., data that is useful at a range of fidelity levels (such as audio, image or video data). This characteristic contrasts with the distortion-intolerance of data that must be communicated losslessly (such as executable programs). There is a wide range of problems – such as data caching, routing, and congestion control – where ideas of buffering and distributing apply. Furthermore, there is a wide range of applications – such as data fusion in sensor networks or multimedia content distribution on the Internet – where the data being handled is largely distortion-tolerant. Pairing these problems and applications leads to the ideas of queuing with distortion-control that we develop herein.

The content buffering and distribution protocols we design operate at the network layer of Fig. 1.1 and require that multiresolution source codes [35, 36, 26, 66] are used at the application layer. We use the ordered information structure of multiresolution

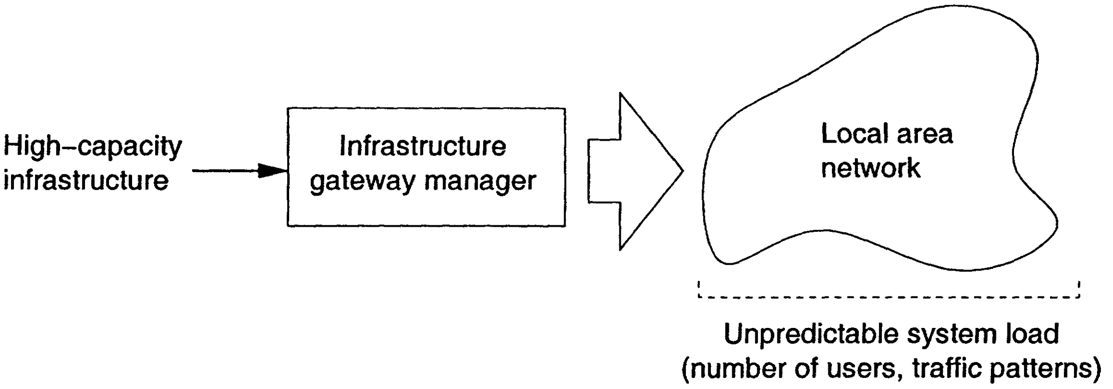

High-capacity infrastructure
Infrastructure gateway manager
Local area network
Unpredictable system load (number of users, traffic patterns)

Figure 1.3. Possible applications of the buffering and distribution protocols developed in this thesis include infrastructure gateway managers. An objective is to design algorithms that can adapt to unpredictable system loads caused by users entering and exiting the network, as well as by unpredictable traffic patterns.

codes to design a pair of network-layer-level priority storage and transmission protocols. Multiresolution source codes have been proposed before as a natural approach to the coding of multimedia content for networks [3, 54, 56]. When the link layer of Fig. 1.1 is particularly unreliable or delay-prone, the non-order structure of multiple descriptions source codes [78, 47, 29] has been proposed [33, 55] as an alternative to multiresolution codes where subcodes must be decoded in a particular order.

A central contribution of this thesis is the joining of multiresolution source coding ideas with queuing theoretic models of networks. This enables us to model situations such as the one illustrated in Fig. 1.3. Here a finite memory infrastructure gateway manager connects two heterogeneous networks. On the left is a high-capacity network connecting information sources to the gateway. On the right is a shared communication medium such as a wireless channel or a local area network. The number of users on the shared medium may be time-varying and traffic levels may be unpredictable, making fluctuating demands on the local area network. We use multiresolution source coding ideas to develop protocols that are robust to unpredictable fluctuations in system load and we quantify this robustness. A major benefit of the protocols developed is that buffer overflows are avoided in a dynamic fashion.

Earlier researchers [27, 41, 71, 72] have looked at the related problem of controlling the source quantization rate based on the state of the buffer to avoid overflows and minimize average distortion. Our work differs from theirs because, by using multiresolution source codes, we can effectively change the source quantization rate *long after the source is quantized* by deleting least-significant description as a function of the state of the buffer. Some related ideas that have been developed in packet scheduling contexts for wireless communications can be found in [50].

Using queuing models we derive a lower-bound on the average end-to-end distortion that can be achieved by any buffering protocol. We develop baseline protocols that obey the separation principle, as well as adaptive protocols that work across traditional network layers. We show that the adaptive protocols are much more robust to uncertainty in queue arrival and departure statistics than are the baseline protocols. Furthermore, their performance closely approximates the performance bound.

■ 1.3 Thesis Outline

Chapter 2, Background: Using Side Information discusses how to employ side information in network communication problems. We motivate the problems, and discuss both the source coding and channel coding versions. We focus on the finite-alphabet and quadratic-Gaussian rate distortion and capacity results. In the quadratic-Gaussian case we present geometric derivations of the rate-distortion and capacity expressions. We conclude the chapter with a discussion of the dualities between source and channel coding with side information.

In **Chapter 3, Side Information Problems with Noisy Encoder Observations** we quantify the effect that noisy encoder observations have on the rate-distortion and capacity results derived in Chapter 2. We generalize Wyner-Ziv source coding and present the rate-distortion function for finite-alphabet sources and arbitrary distortion measures. We further evaluate the rate-distortion function for the binary-Hamming and quadratic-Gaussian cases. Similarly, we develop the capacity expression for channel coding with side information in the context of information embedding. We present the capacity expression for finite-alphabet sources and arbitrary distortion measures and evaluate this expression for the quadratic-Gaussian case. For the quadratic-Gaussian cases, we present the geometric derivations, allowing comparison with the analogous geometric pictures of Chapter 2.

Chapter 4, Successively Structured Data Fusion Algorithms for Sensor Networks develops successive coding strategies for sensor network problems based on generalized Wyner-Ziv coding. We present a general model for sensor networks on trees. We then show how to decompose any general tree into sets of two prototype network configurations: serial (pipeline) and parallel (hub-and-spoke). We first develop "estimate-and-quantize" strategies that are appropriate for use in layered network architectures. The second approach is based on viewing both serial and parallel problems as cascades of noisy Wyner-Ziv stages and results in an inter-layer algorithm.

By interpreting these sensor networks as side information problems we are able to develop approaches to both the parallel and serial configurations that are, in a sense, dual. We analyze these strategies in the case of a finite number of agents and find convenient iterative (achievable) distortion-rate expressions for quadratic-Gaussian scenarios. Under a sum-rate constraint, the parallel network is equivalent to the CEO problem of information theory, and we connect our work to those earlier results. Using our approach, we thoroughly analyze the two-sensor and infinite-sensor cases of the CEO problem, proving that the rate-distortion bound is attained in both. Combining our techniques for the serial and parallel configurations provides a good coding strategy for arbitrary sensor trees. Based on this work, we further develop coding strategies for relay channel communications.

In **Chapter 5, Queuing with Distortion-Control** we wrap a signal processing interface around a basic memory unit to produce buffering systems for distortion-tolerant data that attempt to minimize end-user distortion. We design two interfaces: a baseline interface that is appropriate for use in a layered network architecture, and an adaptive interface that works across traditional network protocol layers. The adaptive interface uses the ordered information structure of multiresolution source codes to alleviate network congestion adaptively. This is accomplished by reducing the fidelity at which data is stored in a controlled manner, avoiding uncontrolled data loss due to buffer overflows. Compared to the traditional baseline approach, the performance gains of the adaptive algorithm are significant – impacting distortion, delay, and overall system robustness – and closely approximate a bound on the performance of any algorithm.

Chapter 6, Conclusions discusses the contributions of this thesis and extensions of the work.

Chapter 2

Background: Using Side Information

Much of communication theory has been developed for point-to-point communication. In there scenarios there is only a single source of information at each processing stage: the message at the encoder, and the received signal at the decoder. With this as the default scenario, other sources of useful information available at either the encoder or the decoder are called "side" information. In this chapter we describe how source and channel coding strategies can be designed to exploit certain types of side information.

In Section 2.1 we consider the problem of Wyner-Ziv source coding with side information. We present the basic problem and set it in the context of other distributed source coding problems, before describing Wyner and Ziv's elegant solution and developing geometric interpretations for the quadratic-Gaussian case. In Section 2.2 we consider Gel'fand and Pinsker's channel coding dual of Wyner-Ziv source coding. We comment on the particular relevance of these ideas to information embedding and watermarking problems, and develop geometric interpretations for Costa's solution to the quadratic-Gaussian version of the problem. In Section 2.3 we discuss the duality relationships between the source and channel coding problems of this chapter. Finally, we close the chapter in Section 2.4 with a summary of the results we present.

■ 2.1 Source Coding and Side Information

We begin our discussion of source coding with side information by placing it in a large class of distributed source coding problems. Consider the following model for distributed source coding problems, depicted in Fig. 2.1: a pair of length-n random source vectors $\mathbf{x}$ and $\mathbf{y}$ are observed at two separate encoders. The source vectors are generated in a pairwise independently identically distributed (pairwise i.i.d.) manner, $p_{\mathrm{x,y}}(\mathbf{x},\mathbf{y}) = \prod_{i=1}^{n} p_{\mathrm{x,y}}(x_i, y_i)$, and the joint statistics are known throughout the system. The encoder for $\mathbf{y}$ encodes $\mathbf{y}$ into the message m_y at rate R_y bits per source sample. Depending on whether switches (a) is open, or closed, the $\mathbf{x}$ encoder may, or may not, know m_y when $\mathbf{x}$ is encoded into m_x. Similarly, depending on whether switches (b) is open, or closed, the $\mathbf{x}$ decoder may, or may not, know m_y when m_x is decoded. The decoder's objective is to reproduce $\mathbf{x}$ and $\mathbf{y}$ to within (usually average) distortions d_x and d_y, where d_x and d_y are the average distortions given by two (possibly different) distortion measures, $E\left[D_x(\mathbf{x},\hat{\mathbf{x}})\right]$ and $E\left[D_y(\mathbf{y},\hat{\mathbf{y}})\right]$, respectively. Depending on the positions of the (a) and

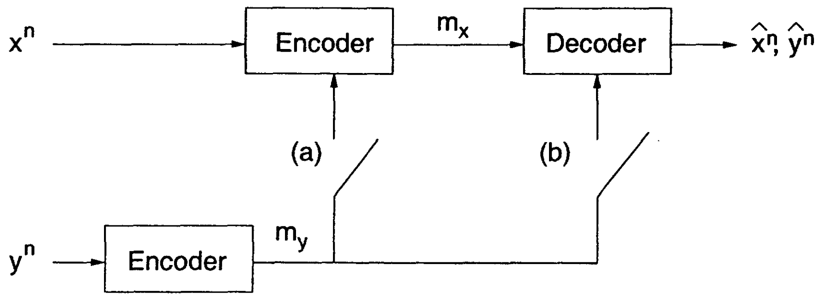


Figure 2.1. Distributed source coding with two switches, (a) and (b).

Problem name	Rate R_y	(a)	(b)	d_x	d_y	References
Lossless source coding of x	0	○	○	0	∞	[63]
Lossy source coding of x	0	○	○	≥ 0	∞	[65, 28, 7]
Slepian-Wolf source coding	$\leq H(y)$	○	●	0	0	[67, 10, 24]
Multiterminal lossy coding	$\leq H(y)$	○	●	≥ 0	≥ 0	[8, 74, 87]
Conditional rate-distortion theory	$\geq H(y)$	●	●	≥ 0	∞	[34]
Lossless coding with side info.	$\leq H(y)$	○	●	0	∞	[82, 2, 10, 24]
Lossy coding with side info.	$\geq H(y)$	○	●	≥ 0	∞	[84, 10, 24]

Table 2.1. Relationship between problems and parameters of Fig. 2.1. Column 2 indicates whether the y-encoder can losslessly communicate y to the decoder ($R_y \geq H(y)$). This is equivalent to the decoder observing y directly. If $R_y \leq H(y)$ the y-encoder cannot losslessly communicate y to the decoder. In the first two problems there is no y to encode. Columns 3 and 4 indicate whether the respective switches are open (○) or closed (●). Columns 5 and 6 indicate the distortion requirements on the decoded source estimates. A non-zero value indicates a rate distortion (lossy) problem while $d_y = \infty$ means that the decoder is not concerned with estimating y in this problem.

(b) switches, the rate R_y, and the decoder's particular goal, a number of interesting problems arise, some of which we list in Table 2.1.

The last two problems in Table 2.1, those concerning source coding with side information, are the most relevant to this thesis. In these problems, the decoder wants to make the best approximation of **x** possible, and is not concerned with estimating **y**; **y** is therefore termed side information. We present a solution to the lossless version of the problem in Section 2.1.2, and present Wyner and Ziv's solution to the lossy version in Section 2.1.3. Before discussing these information theoretic solutions, we develop intuition through a scalar example.

■ 2.1.1 Scalar Quantization with Side Information

Consider the following scalar version of the source coding with side information problem. The scalar source x is a zero-mean Gaussian random variable with variance σ_x^2, i.e., $x \sim \mathcal{N}(0, \sigma_x^2)$. The decoder receives message m_x and measures side information y. The side

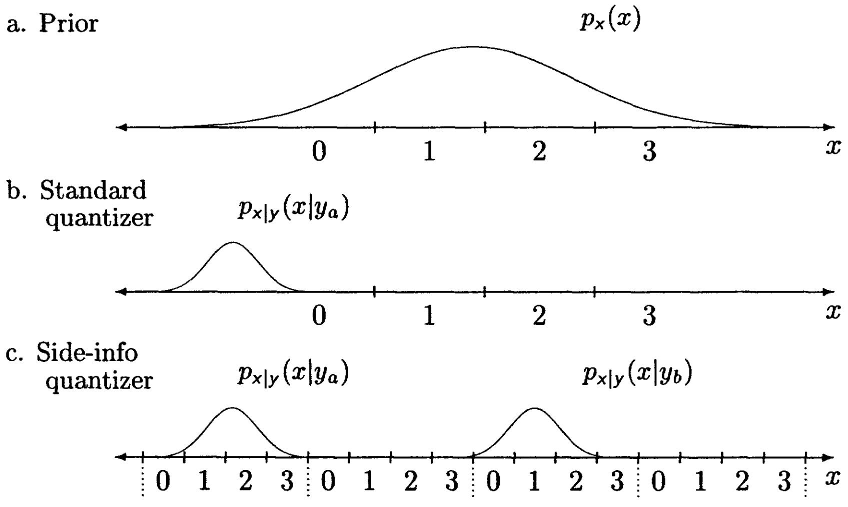


Figure 2.2. Design intuition for scalar quantizers with decoder side information. Fig. 2.2-a plots the prior $p_x(x)$ versus the quantization regions of a standard two-bit quantizer. Fig. 2.2-b plots the posterior $p_{x|y}(x|y_a)$ for the source x given the side information is y_a. Most of the posterior distribution is supported within the single quantization region labeled 0. Fig. 2.2-c plots two possible posteriors for x versus the quantization regions of a non-standard scalar quantizer. In this design the side information y is used to resolve the ambiguity in quantizer region. For example, if x was quantizer to 0, and we observed y_a we would guess that x is in the left-most region labeled 0. If, on the other hand y_b were measured, then we would guess that x is in the third region from the left labeled 0.

information is related to x by an additive noise channel $y = x + v$ where $v \sim \mathcal{N}(0, \sigma_v^2)$. The decoder's goal is to produce the source approximation $\hat{x}$ that minimizes the mean-squared distortion (or mean-squared error) $d_x = E\left[|x - \hat{x}|^2\right]$.

First consider the performance of a standard scalar quantizer in this situation. Figure 2.2-a plots the prior for x, versus the quantization regions of a two-bit scalar quantizer that might have been designed, e.g., using the Lloyd-Max algorithm [32]. The encoder sends the index $m_x \in \{0, 1, 2, 3\}$ that corresponds to the region in which x is measured. The decoder maps the received index to a source reconstruction $\hat{x}$, typically the conditional mean of the quantization region $\int xp(x|m)dx$. The resolution of this type of quantizer is limited by the width of the quantization regions.

Figure 2.2-b helps demonstrate why this approach is suboptimal for quantization systems that have decoder side information. A possible posterior distribution $p(x|y_a)$ for x given side information observations y_a is plotted. In this scenario most of the posterior probability is located in the quantization region labeled 0. It is very likely that x is also located in this region. We assume it was and that the decoder therefore received index 0 from the encoder. Receiving index 0 confirms that x was in this quantization region, but we were already quite sure of that from $p(x|y_a)$. Therefore, the index m_x does not tell us much that the side information did not already, and can pretty much be ignored. This is somewhat unsatisfying, however, as the quantizer measures the source perfectly, while the side information y is a *noisy* observation of x. We want to design a system that takes advantage of the encoder's clean observation of x.

In Fig. 2.2-c we show a set of quantization regions that can be used to exploit the clean encoder observation. The source x is again encoded into an index in the set $\{0, 1, 2, 3\}$ depending on which quantization region x is located in. However, there are now sixteen quantization regions, and only four indices. Because there are more quantization regions than indices, we must reuse indices when labeling the regions. The result at the decoder is a non-unique mapping from received index to quantization region. This ambiguity can be resolved by using the side information.

To see how to use the side information in resolving the ambiguity, consider the two possible posteriors $p(x|y_a)$ and $p(x|y_b)$ indicated in Fig. 2.2-c. Say that the decoder again receives index $m_x = 0$ from the encoder. If the side information was measured to be y_a, then the left-most region labeled 0 would be most likely. However, if the measurement was y_b, the third region from the left labeled 0 would be most likely. Through this algorithm we can use the side information to resolve the ambiguity in labeling and determine which of the similarly lapelled regions is the one in which the source was located. In effect, by using the side information we were able to double our quantization rate from two bits (four region) to four (sixteen regions), while keeping the communication rate fixed at two bits. There were two bits of uncertainty which were resolved through the side information.

The scalar design shown in Fig. 2.2-c uses a periodic quantizer. Because of the non-infinite spacing between similarly-labeled quantization regions there is always a non-zero probability that the decoder will identify the incorrect quantization region. If the spacing is increased this probability decreases. However, as the spacing is increased, then the performance gains from the periodic quantizer decrease. Good designs balance the probability of decoder error with the gains made when the decoding errors are not made. The design and analysis of these quantization systems is carried out in [4].

However, when considering higher dimensional generalization of this scalar system, the probability of identifying the incorrect quantization region can be driven to zero asymptotically as the length of the source signal grows to infinity. This asymptotic vector version of the problem is known as Wyner-Ziv source coding with side information. And, as we will see, the solution to the Wyner-Ziv problem, the solution to which displays the same type of periodic quantizer structure as shown in Fig. 2.2.

2.1.2 Lossless Vector Source Coding with Side Information

As a prelude to developing the information theoretic results for the vector generalization of the scalar problem of Section 2.1.1, we first develop the problem of lossless source coding with side information. In this case, the decoder's objective is to reconstruct the source perfectly $\hat{\mathbf{x}} = \mathbf{x}$ with probability approaching one as the source block length n approaches infinity. This is the second-to-last problem in Table 2.1 where switch (a) is open (∘)and switch (b) is closed (•).

Theorem 1 *[81, 2] Let a pair of sources* $\mathbf{x}$ *and* $\mathbf{y}$ *jointly distributed pairwise i.i.d.,* $p_{\mathbf{x},\mathbf{y}}(\mathbf{x},\mathbf{y}) = \prod_{i=1}^{n} p_{x,y}(x_i, y_i)$*, be given.*

If (a) $\mathbf{x}$ *is encoded at rate* R_x*,*
(b) $\mathbf{y}$ *is encoded at rate* R_y*,*

then a sequence of length-n block encoder-decoder pairs can be designed such that the probability that $\hat{\mathbf{x}} \neq \mathbf{x}$ *can be made arbitrarily small as n grows to infinity, if and only if there exists an auxiliary random variable u such that:*

(i) $R_x \geq H(x|u)$*, and*

(ii) $R_y \geq I(y;u)$*,*

(iii) x,y,u forms a Markov chain, $x \leftrightarrow y \leftrightarrow u$.

This problem has two encoders and therefore two codebooks. We next summarize the achievability half of the proof. For the $\mathbf{x}$ encoder, randomly and uniformly assign each typical sequence $\mathbf{x}$ an index from $\{1, 2, \ldots, 2^{nR_x}\}$.[1] These typical sequences are the sequences we want to be able to identify at the decoder. For the $\mathbf{y}$ encoder, generate a codebook with 2^{nR_y} codewords $\mathbf{u}(j)$, $j \in \{1, 2, \ldots 2^{nR_y}\}$, where each $\mathbf{u}$ is generated in an i.i.d. manner according to $p(\mathbf{u}) = \prod_{i=1}^{n} u_i$.

The $\mathbf{x}$ encoder sends to the decoder the index m_x of the set in which the realized source sequence $\mathbf{x}$ lies. This subset contains many typical $\mathbf{x}$ sequences (about $2^{n(H(x)-R_x)} \simeq 2^{nI(x;u)}$ of them). The $\mathbf{y}$ encoder looks through all the codewords, $\mathbf{u}(1), \ldots, \mathbf{u}(2^{nR_y})$ for one jointly typical with $\mathbf{y}$. It sends the corresponding index m_y to the decoder. The side information m_y is used to select which typical source sequence within subset m_x was the realized sequence.[2] This selection is done via joint typicality arguments, and relies on the Markov Lemma [8]

As a point of reference, consider what happens if we choose the auxiliary random variable $u = y$. Then we get $R_x \geq H(x|y)$, $R_y \geq I(y;y) = H(y)$. In this situation $\mathbf{y}$ can be transmitted losslessly and only the residual randomness in $\mathbf{x}$ needs be sent. This is a special case of Slepian-Wolf coding [67]. Because we do not care about decoding $\mathbf{y}$ in the side information problem, we can generally save rate by not communicating $\mathbf{y}$ to the decoder perfectly. The side information $\mathbf{u}$ is a function of $\mathbf{y}$ and so by the Data Processing Inequality can have no more information about $\mathbf{x}$ than does $\mathbf{y}$. This gives us a lower bound on R_x, i.e. $R_x \simeq H(x|u) \geq H(x|y)$.

---

[1] This is often called "random binning" where the set of sequences associated with each index is called a "bin" of sequences.

[2] Note that $R_y \geq I(y;u) \geq I(x;u)$ by the Markov property and the Data Processing Inequality. Therefore it is reasonable to believe that the side information rate is high enough to do the intra-subset selection since $R_x + R_y \geq H(x|u) + I(y;u) \geq H(x|u) + I(x;u) = H(x)$.

2.1.3 Lossy Vector Source Coding with Side Information

In the problem of lossy source coding with side information, the design objective is to minimize the transmission rate needed to guarantee that the source decoder can approximate the source to within average distortion d_x, i.e., $E[D_x(\mathbf{x}, \hat{\mathbf{x}})] \leq d_x$ with probability approaching one as the source vector length n approaches infinity. This is the last problem in Table 2.1. Generally, is assumed to be observed at the decoder directly, rather than through a finite-rate encoder.

Theorem 2 *[84] Let a pair of jointly distributed random source vectors* $(\mathbf{x}, \mathbf{y})$ *and a distortion measure* $D_x(\cdot,\cdot)$ *be given such that*

(a) $p_{\mathbf{x},\mathbf{y}}(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{n} p_{x,y}(x_i, y_i)$, *and*

(b) $D_{\mathbf{x}}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{n}\sum_{i=1}^{n} D_x(x_i, \hat{x}_i)$.

Then a sequence of length-n block encoder-decoder pairs can be designed such that if $\mathbf{x}$ *is encoded at rate* R_x*, the source* x *can be recovered to within average distortion* d_x *with arbitrarily small probability of failure as n grown to infinity, if and only if*

$$R_x > R^{\mathrm{WZ}}(d_x) = \min_{p_{u|y}(u|y) \in \mathcal{U}} [I(x; u) - I(y; u)], \tag{2.1}$$

where the set $\mathcal{U}$ *consists all posteriors* $p(u|y)$ *relating the auxiliary random variables* u *to* y *that satisfy the following two conditions:*

(i) $x \leftrightarrow y \leftrightarrow u$, *and*

(ii) $E[D_x(x, f(y, u))] \leq d_x$ *for some memoryless function* $f : \mathcal{Y} \times \mathcal{U} \to \hat{\mathcal{X}}$.

We use the notation $R^{\mathrm{WZ}}(d_x)$ to refer to the Wyner-Ziv rate distortion function. This differentiates it from the conditional rate distortion function of Gray [34], denoted by $R_{\mathbf{x}|\mathbf{y}}(d_x)$, which is the rate-distortion function when the side information $\mathbf{y}$ is measured at both encoder and decoder.

The encoding technique in the rate distortion case is similar to the lossless case, except that now we bin the $\mathbf{x}$ sequences instead of the $\mathbf{y}$. This is because the side information $\mathbf{y}$ is known at the decoder and we only need an approximation $\hat{\mathbf{x}}$ to $\mathbf{x}$ at the decoder. To encode the source we first generate a codebook of about $2^{nI(x;u)}$ sequences $\mathbf{u}$ according to $p_{\mathbf{u}}(\mathbf{u}) = \prod_{i=1}^{n} p_u(u_i)$. These are the $\mathbf{u}$-sequences we want to be able to identify at the decoder using the side information $\mathbf{y}$. Next, the codebook sequences are assigned randomly and uniformly to $2^{nR_x} \leq 2^{nI(x;u)}$ subsets (or bins). To encode we first fine the $\mathbf{u}$ sequence that is jointly typical with $\mathbf{x}$ according to $p_{u,x}(u, x)$. The bin index in which this $\mathbf{u}$ lies, say set i, is transmitted to the receiver at rate R_x. This bin contains many $\mathbf{u}$ sequences (about $2^{n(I(x;u)-R_x)} \simeq 2^{nI(y;u)}$ of them). The side information is used to select which $\mathbf{u}$-sequence within set i is the one that is jointly typical with $\mathbf{x}$. This selection is done via joint typicality arguments, and the Markov

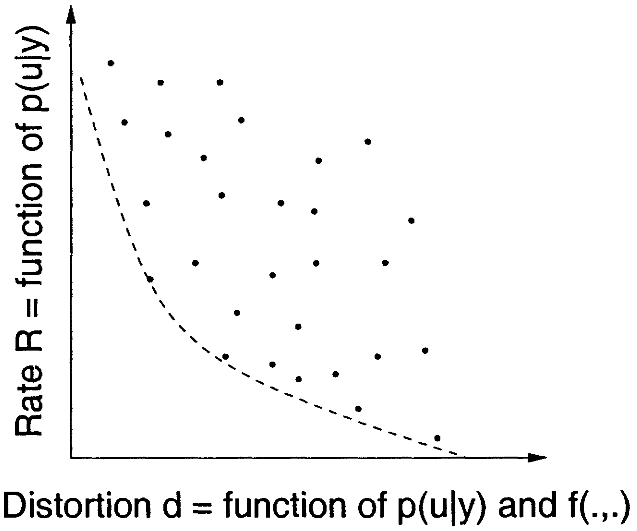


Figure 2.3. Each choice of $p(u|y)$ and $f(\cdot,\cdot)$ defines one achievable rate-distortion point (d,R). The rate-distortion function $R(d)$ is the lower convex envelope of all achievable points and is indicated by the dashed curve.

Lemma. After correct identification of **u**, the final step is is to fuse together **u** with the side information **y**, producing $\hat{x}_i = f(u_i, y_i)$.

Knowing how the coding theorem works, we now go back to parse the statement of Thm. 2. The theorem breaks into two steps: decoding and data fusion. The limit on the rate R_x, given by the difference of mutual informations (2.1), makes sure our rate is high enough to perform the selection of the correct **u** from the set i. The Markov condition (i) guarantees that the selection works by the Markov lemma. Assuming that the correct **u** is identified, we turn to the data fusion step. The second condition (ii) guarantees that we can find a fusion function f that satisfies the distortion constraint. In general, for each choice of $p(u|y)$ and $f(\cdot,\cdot)$ we get a rate and a distortion defining a point on the achievable rate-distortion graph plotted in Fig. 2.3. The rate is a function of the conditional probability $p(u|y)$ chosen, and the distortion is a function of $p(u|y)$ as well as the data fusion function $f(\cdot,\cdot)$. The rate-distortion function is the lower convex envelope of all that points that can be so defined, indicated by the dashed curve in the figure.

We complete this section by pointing out the connection between the rate distortion function $R(d) = \min[I(x;u) - I(y;u)]$ of Thm. 2 and the scalar example, of Fig. 2.2-c. In the scalar example the communication rate (analogous to $R(d)$ in Thm. 2) was two bits per sample, the quantization rate (analogous to $I(x;u)$) was four bits per sample, and the resolution rate (analogous to $I(y;u)$) was two bits per sample. Setting the analogous terms equal gives $2 = R(d) = I(x;u) - I(y;u) = 4 - 2 = 2$. For work on Slepian-Wolf and Wyner-Ziv implementations see [80] and [4, 52, 62, 86, 88], respectively.

2.1.4 Quadratic-Gaussian Case

Wyner determined the rate distortion region for i.i.d. Gaussian sources under a mean-square distortion measure [83], commonly referred to as the 'quadratic-Gaussian' case. For a pairwise i.i.d. jointly Gaussian zero-mean source where $x \sim \mathcal{N}(0, \sigma_x^2)$ and $y \sim \mathcal{N}(0, \sigma_y^2)$,

$$R^{\mathrm{WZ}}(d_x) = \begin{cases} \frac{1}{2}\log\frac{\sigma_{x|y}^2}{d_x}, & 0 \le d_x \le \sigma_{x|y}^2 \\ 0, & \sigma_{x|y}^2 < d_x \end{cases} \tag{2.2}$$

where $\sigma_{x|y}^2$ is the minimum mean-squared estimation error:

$$\sigma_{x|y}^2 = E\left[(x - \hat{x}(y))^2\right] = E\left[(x - E[x|y])^2\right] = E\left[x^2\right] - \frac{E[xy]^2}{E[y^2]}. \tag{2.3}$$

Interestingly, in this case $R^{\mathrm{WZ}}(d_x) = R_{x|y}(d_x)$, Gray's conditional rate distortion function [34] where $\mathbf{y}$ is known at both encoder and decoder. This equality is probably related to the fact that the posterior variance $\sigma_{x|y}^2$ is independent of the realization of $\mathbf{y}$. This conjecture seems related to the results of [19, 89] which show that in the channel coding dual that we discuss in Section 2.2, it is not the host statistics, but rather the channel statistics that matter. In the source coding context then it is not the statistical prior $p(x)$ that matters, but rather the posterior $p(x|y)$.

We now show how, in the low-distortion regime, the rate distortion function for this problem (2.2) can be derived from geometric sphere-packing arguments [5]. The minimum mean-squared error estimate given the side information is $E[\mathbf{x}|\mathbf{y}]$; the associated estimation error is $\sigma_{x|y}^2$, which can be achieved without using the encoded message. Therefore, before using the message from the encoder, the decoder can determine that the true vector source $\mathbf{x}$ lies within an uncertainty ball of radius $\sqrt{n(\sigma_{x|y}^2 + \epsilon_1)}$ centered at its side information based source estimate $E[\mathbf{x}|\mathbf{y}]$. This ball of uncertainty, indicated by the dotted circle in Fig. 2.4, is centered around the tip of a vector indicating $E[\mathbf{x}|\mathbf{y}]$, the side information based source estimate. Within the ball we pack spherical quantization regions of radius $\sqrt{n(d - \epsilon_2)}$. At the encoder we map $\mathbf{x}$ to the label of the quantization region in which it falls. However, these labels are not unique: just as in the scalar example, many quantization regions are assigned the same label. If we can resolve the ambiguity in the labeling using the side information, then we can determine the small $\sqrt{n(d - \epsilon_2)}$-radius ball in which $\mathbf{x}$ is located, satisfying the distortion constraint. The ambiguity in the labeling can always be resolved without error as long as within any large sphere of radius $\sqrt{n(\sigma_{x|y}^2 + \epsilon_1)}$, wherever centered, no two quantization regions share the same label. In the random coding proof outlined in Section 2.1.3, this is the same as finding the unique codeword in set i that is jointly typical with the side information sequence. Thus, the set of codewords in each set in the random coding proof correspond to the set of quantization regions that share the same label in Fig. 2.4 or Fig. 2.2-c.

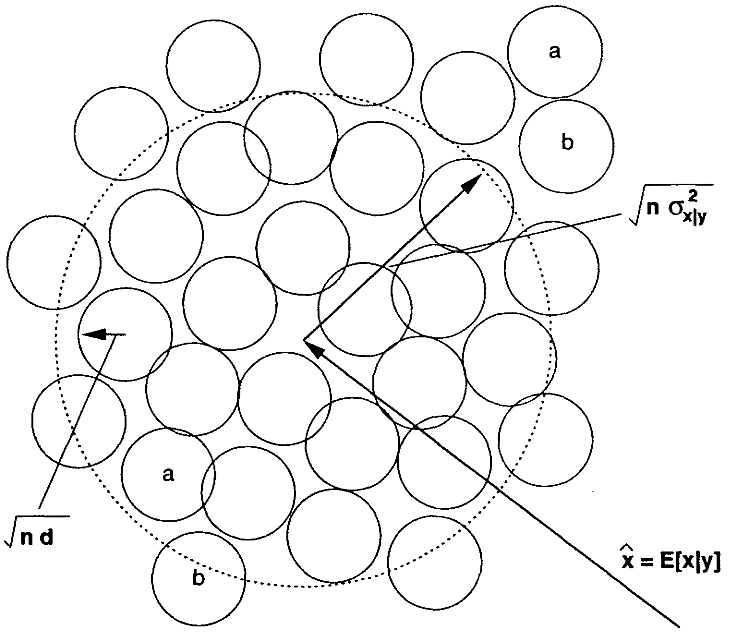


Figure 2.4. The rate distortion function for Wyner-Ziv source coding with side information in the low distortion-to-noise ($d/\sigma^2_{x|y}$) regime can be found via sphere covering arguments. The dotted circle correspond to the source uncertainty given the side information, $\sigma^2_{x|y}$. The solid circles correspond to the radius-$\sqrt{nd}$ quantization regions at the encoder. The labeling of these quantization regions is not unique, e.g., two are labeled 'a', and two 'b' in the picture.

To determine a lower bound on the rate distortion function we determine the minimum number of spheres of radius $\sqrt{nd}$ required to cover the large dotted sphere. This number is lower bounded by the ratio of volumes:

$$\begin{aligned} M &\geq \frac{\kappa(n)\left(\sqrt{n(\sigma^2_{x|y}+\epsilon_1)}\right)^n}{\kappa(n)\left(\sqrt{n(d-\epsilon_2)}\right)^n} \\ R = \frac{1}{n}\log_2 M &\geq \frac{1}{2}\log\left[\frac{\sigma^2_{x|y}+\epsilon_1}{d-\epsilon_2}\right] > \frac{1}{2}\log\left[\frac{\sigma^2_{x|y}}{d}\right] \end{aligned} \tag{2.4}$$

where $\kappa(n)$ is a coefficient that is a function of the dimension[3], and where (2.4) equals (2.2).

The rate distortion region is also known for discrete binary-symmetric sources with Hamming distortion, the "binary-Hamming" case. In this case $R^{\mathrm{WZ}}(d_x)$ is generally strictly greater than the conditional rate-distortion function $R_{x|y}(d_x)$. This example tells that that generally $R^{\mathrm{WZ}}(d_x) \neq R_{x|y}(d_x)$. In the quadratic-Gaussian case the rate distortion bound can be achieved via nested lattice quantizers [4, 88, 89]. In the discrete

---

[3] For example, $\kappa(1) = 2$, $\kappa(2) = \pi$, and $\kappa(3) = 4\pi/3$.

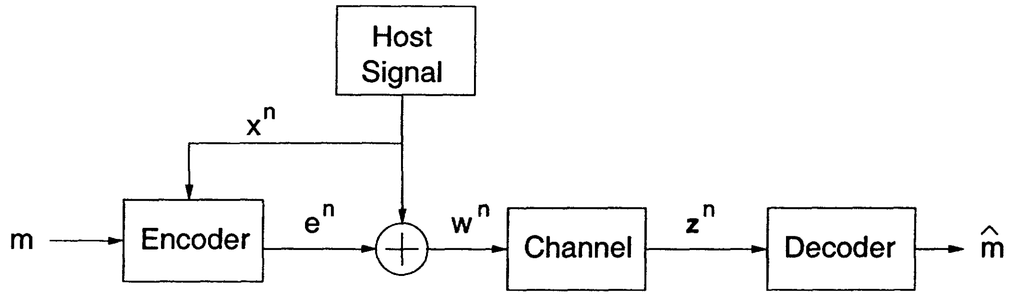


Figure 2.5. Channel coding with state side information at the encoder.

binary-symmetric Hamming case, the rate distortion limit can be achieved via nested linear codes [62].

■ 2.2 Channel Coding with Side Information

Side information can also be exploited in channel coding problems. The channel coding equivalent of the Wyner-Ziv source coding problem discussed in Section 2.1 is illustrated in Fig. 2.5. In this case the side information is knowledge about the state of the channel, available at the encoder, but not the decoder. The version of the channel coding with side information problem depicted in Fig. 2.5 is particularly relevant to problems of information embedding and watermarking.

The basic information embedding problem is to 'embed' (or hide) the message m in the host signal $\mathbf{x}$ robustly, and without causing too much distortion [14]. More exactly, the i.i.d. host signal $\mathbf{x}$ is known non-causally to the encoder, but not to the decoder. As a function of m and $\mathbf{x}$ the encoder produces an embedding signal $\mathbf{e}$ that is added to the host $\mathbf{x}$ producing the channel input $\mathbf{w}$. The decoder measures $\mathbf{z}$ which is related to $\mathbf{w}$ by the memoryless channel law $p(z_i|w_i)$. We want to be able to decode m from the $\mathbf{z}$ reliably so that the probability that the decoded message $\hat{m}$ is not equal to the transmitted message m can be made to converge to zero as the block length n approaches infinity. Furthermore, we do not want the embedding signal to reduce the host fidelity too much, so we place an average distortion constraint $E\left[D(\mathbf{x}, \mathbf{w})\right]$ between the host and the channel input.

The general problem of channel coding with side information, of which this is a particular instance, was first investigate by Shannon [64] in the case where the host is known causally. Gel'fand and Pinsker [31] and later Costa [20] developed the capacity expression for the non-causal case, of which our information embedding scenario is an example. More recently, connections have been made between channel coding with side information and multi-antenna array communications [12].

In [5] the authors extend the results of [31, 20] to the information embedding situation and show the following theorem.

Theorem 3 *Let a random source* $\mathbf{x}$*, a distortion measure* $D(\cdot, \cdot)$*, and a memoryless* *channel law* $p(z|w)$ *be given such that*

(a) $p_{\mathsf{x}}(\mathbf{x}) = \prod_{i=1}^{n} p_{\mathsf{x}}(x_i)$,

(b) $D(\mathbf{x}, \mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} D(x_i, w_i)$,

where $\mathbf{w} = \mathbf{x} + \mathbf{e}$ *is the channel input and* $\mathbf{e}$ *is the embedding signal. Then, a sequence of length-*n *encoder-decoder pairs can be designed such that a message of rate* R *can be communicated to the decoder with an arbitrarily small probability of decoding error as* n *grows to infinity while satisfying the average distortion constraint* d *if and only if*

$$R < C_{\mathrm{I}}^{\mathrm{IE}}(d) = \max_{p_{u|x}(u|x)\in\mathcal{U}} [I(z; u) - I(x; u)] \tag{2.5}$$

where the set $\mathcal{U}$ *consists of all posteriors* $p(u|x)$ *relating the auxiliary random variable* u *to* x *that satisfy the two conditions:*

(i) $p(u|x, e, w, z) = p(u|x)$,

(ii) $E[D(x, w)] \leq d$ *where* $w = x + e$ *and* $e = f(u, y)$ *for some memoryless function* $f : \mathcal{U} \times \mathcal{X} \to \mathcal{E}$.

In the case where $\mathbf{x}$ is an i.i.d. Gaussian vector, the channel is an additive white Gaussian noise channel with noise variance N, and the distortion constraint is mean-squared distortion, i.e., $\frac{1}{n}E\left[\|\mathbf{x} - \mathbf{w}\|^2\right] \leq d$, the capacity of this system is [20, 14]

$$C = \frac{1}{2}\log\left(1 + \frac{d}{N}\right). \tag{2.6}$$

This is the same capacity as if the host were known at both the encoder and receiver and so could be subtracted out. This is analogous to the result of Section 2.1.4 where the Wyner-Ziv rate distortion equaled the conditional rate distortion function. Costa named this scenario 'writing on dirty paper' where the state $\mathbf{x}$ is the 'paper' that is 'dirty' since $\mathbf{x} \neq 0$.

Similar to Wyner-Ziv source coding, the channel capacity in this quadratic-Gaussian case can be derived geometrically. In Fig. 2.6 we diagram the sphere-packing arguments we step through next. The vector labeled $\mathbf{x}$ in the figure indicates the host signal known at the encoder. The distortion constraint d means that the codeword (vectors) we transmit must lie in a sphere of radius $\sqrt{n(d - \epsilon_1)}$ centered around the host signal $\mathbf{x}$. The variance-N channel noise enlarges this sphere to radius $\sqrt{n(d + N - \epsilon_1)}$. Furthermore, so that the channel noise does not cause a decoding error, the codewords must lie in non-overlapping spheres of radius at least $\sqrt{n(N + \epsilon_2)}$. If M is the number of codewords we transmit, we upper bound M as

$$\begin{aligned} M &\leq \frac{\kappa(n)\left(\sqrt{n(d + N - \epsilon_1)}\right)^n}{\kappa(n)\left(\sqrt{n(N + \epsilon_2)}\right)^n} \\ C = \frac{1}{n}\log_2 M &\leq \frac{1}{2}\log\left[\frac{d + N - \epsilon_1}{N + \epsilon_2}\right] < \frac{1}{2}\log\left[1 + \frac{d}{N}\right]. \end{aligned}$$

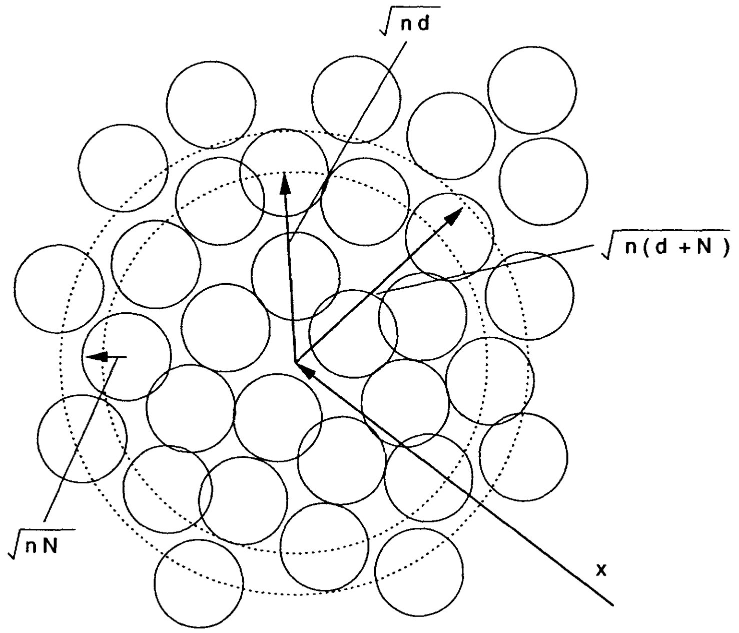


Figure 2.6. Capacity for information embedding region for a Gaussian i.i.d. host of variance σ_x^2 can be found in the high distortion-to-noise d/N regime via sphere packing arguments. The inner dotted circle correspond to the power constraint, the source cannot be moved further than this without violating the distortion constraint. The outer circle indicates the extra distortion incurred by the host due to channel noise. The solid circles correspond the the embedding messages, which must be spaced at least $\sqrt{nN}$ apart to be immune to the channel noise.

In [19, 89] the authors show that in the information embedding model, only the additive noise need be Gaussian for the capacity to be $0.5\log(1+d/N)$. The host $\mathbf{x}$ that is known at the encoder can be more general. This is similar to the discussion of the posterior $p(x|y)$ in the Wyner-Ziv problem discussed at the end of Section 2.1.4.

■ 2.3 Dualities between Source and Channel Coding Strategies

Recently a number of authors (see, e.g., [17, 5, 16, 6] and the references therein) have commented on the duality between the coding with side information problems presented in this chapter. In [6] the authors describe a useful notion of functional duality where by a good Wyner-Ziv encoder makes a good information embedding decoder, and vice-versa. We can most easily understand this duality in quadratic-Gaussian case by referring to the sphere-packing pictures of the Wyner-Ziv problem,Fig. 2.4, and of the information embedding problem, Fig. 2.6.

In the Wyner-Ziv problem the encoder maps from the region in which the source $\mathbf{x}$ is located to a bin number m. In the information embedding problem the decoder maps from the region in which the received signal $\mathbf{y}$ lies to a bin number m. These turn out to be identical tasks. Conversely, in the Wyner-Ziv problem the decoder picks out the quantization vector $\mathbf{u}(s)$ from the uncertainty sphere of radius roughly $\sqrt{n\sigma^2_{\mathsf{x}|\mathsf{y}}}$ that surrounds the source estimate $E[\mathsf{x}|\mathsf{y}]$ by referring to the bin index m. In the information embedding problem the encoder picks out the codeword $\mathbf{w}$ from the distortion sphere of radius roughly $\sqrt{nd}$ that surrounds the host signal $\mathbf{x}$ by referring to the message to be embedding m. In both the Wyner-Ziv and information embedding problems, Fig. 2.4 and Fig. 2.6 tell the whole store in the asymptotically small distortion and high distortion-to-noise regions, respectively. In intermediate regions a second step is needed for both. In the Wyner-Ziv problem this is the data fusion step specified by the function $f(\mathbf{y}_0, \mathbf{u}(s))$, and in the information embedding problem this step often goes by the name of "distortion-compensation" see, e.g., [14].

Finally, in [15] Chiang and Cover consider generalizations of coding with side information problems of this chapter. They consider the case where there are different, jointly distributed side information vectors observed at the encoder and decoder. In terms of Fig. 2.1, switches (a) and (b) are both closed, but connected to two *different* sources of 'state' information. Because the side informations are difference, their result also generalize the conditional rate-distortion theory of Gray [34]. For source coding problems they derive,

$$R(d) = \min_{p(u|x,s_{\mathrm{enc}}),p(\hat{x}|u,s_{\mathrm{dec}})} [I(u; s_{\mathrm{enc}}, x) - I(u; s_{\mathrm{dec}})],$$

where $\mathbf{s}_{\mathrm{enc}}$ and $\mathbf{s}_{\mathrm{dec}}$ are the i.i.d. side information vectors known at the encoder and decoder, respectively. Chiang and Cover show that the channel coding dual is

$$C = \max_{p(u,x,s_{\mathrm{enc}})} [I(u; s_{\mathrm{dec}}, y) - I(u; s_{\mathrm{enc}})]$$

where, in their work, Chiang and Cover do not consider a distortion constraint.

■ 2.4 Chapter Summary

In this chapter we review the coding with side information literature. We particularly focus on the Wyner-Ziv problem, setting the problem in the wider context of distributed source coding, and developing the basic intuition behind Wyner and Ziv's solution through a scalar example. We informally present the achievability proofs for both the lossless and lossy source coding with side information problems. For the latter case, we further discuss the solution in the quadratic-Gaussian case and show how to derive the rate-distortion function through sphere-packing arguments. We next turn to the channel coding dual introduced by Gel'fand and Pinsker and present that problem in the context of information embedding applications. We discuss the capacity expression in the quadratic-Gaussian case first investigated by Costa, and show how to re-derive his results for information embedding through sphere packing arguments. We end the chapter by discussing the dual natures of the source and channel coding with side information problems and present Chiang and Cover's unifying approach to these problems.

,

Chapter 3

Side Information Problems with Noisy Encoder Observations

In this chapter we investigate the effect that noisy encoder observations have on the rate distortion and capacity expression of source and channel coding with side information. We first generalize Wyner-Ziv source coding approach to deal with noisy encoder observations. The resulting model can be applied to a number of practical scenarios, such as multiple-microphone problems in acoustic applications and transcoding for hybrid digital-analog radio. In Chapter 4 we use the results of this chapter to approach the more complex sensor network configurations discussed in the Introduction.

We also generalize the information embedding problem of Section 2.2 to noisy encoder observations in the context of information embedding. After developing the capacity expression we show that a separation theorem holds in the quadratic-Gaussian case. In the context of information embedding this theorem tells us first to estimate the host, and then design the embedding signal as if the estimate were the actual host; estimation uncertainty acts as extra channel noise. While this channel coding generalization plays less of a role in the remainder of the thesis, it is potentially applicable to, for example, the multiple access channel with feedback [48, 23].

In Section 3.1 we develop the noisy encoder generalization of the Wyner-Ziv source coding with side information problem. We present the rate distortion function for finite-alphabet sources with an arbitrary distortion measure. We further evaluate the resultant expression for the binary-Hamming and quadratic-Gaussian cases. In Section 3.2 we develop the dual generalization for information embedding. We present the capacity expression for finite-alphabet channels with an arbitrary distortion measure, and evaluate it for the quadratic-Gaussian case. Finally, in Section 3.3 we develop the sphere-packing derivations of the quadratic-Gaussian rate-distortion and capacity expressions. We conclude the chapter in Section 3.4 with a summary of our results.

■ 3.1 Noisy Source Coding with Side Information

The model for source coding with side information that we now introduce is quite similar to the Wyner-Ziv model of Section 2.1.3. The distinguishing feature is the addition of a memoryless channel between source and encoder. This results in the slight

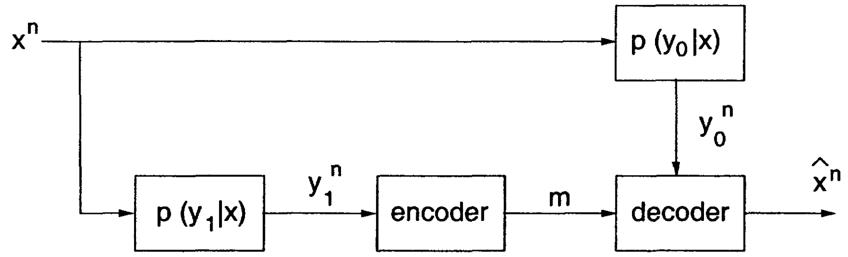


Figure 3.1. Wyner-Ziv source coding with noisy encoder observations. The signals $\mathbf{x}$, $\mathbf{y}_0$, $\mathbf{y}_1$ and m are, respectively, the source, side information, encoder observation, and message.

generalization we need to use this system as a building block for the sensor networks considered in Chapter 4. Because the two models and coding approaches are quite similar we term this problem, and the resulting solution, "noisy" Wyner-Ziv coding.

Fig. 3.1 depicts the source coding with side information scenario of interest. The length-n i.i.d. source vector $\mathbf{x}$ is observed via two memoryless channel laws $p(y_1|x)$ and $p(y_0|x)$ at the encoder and decoder, respectively. Based on its observation $\mathbf{y}_1$, the encoder transmits a message m over a rate-constrained channel to the decoder. The decoder produces $\hat{\mathbf{x}}$, an estimate of the source $\mathbf{x}$, as a function of m and its side information $\mathbf{y}_0$. This scenario differs from Wyner-Ziv source coding because $\mathbf{x}$ is not uniquely determinable from the encoder measurement $\mathbf{y}_1$.

The rate distortion function for source-coding with side information and noisy or "imperfect" encoder observations is denoted $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$. This is a tight lower bound on the rate needed to guarantee that $\frac{1}{n}E\left[\sum_{i=1}^{n} D(x_i, \hat{x}_i)\right]$ can be made arbitrarily close to d for a sufficiently long block length n. The derivation can be viewed as generalizing the results in [84, 15] to accommodate the lack of direct source observations via application of the Markov Lemma [8]. The rate distortion function for finite-alphabet sources is derived in Appendix A.

Theorem 4 *Let a triple of random source and observation vectors,* $(\mathbf{x}, \mathbf{y}_0, \mathbf{y}_1)$ *and a distortion measure* $D(\cdot,\cdot)$ *be given such that:*

(a) $p_{\mathbf{x},\mathbf{y}_0,\mathbf{y}_1}(\mathbf{x}, \tilde{\mathbf{y}}, \bar{\mathbf{y}}) = \prod_{i=1}^{n} p_{\mathbf{x}}(x_i) p_{\mathbf{y}_0|\mathbf{x}}(\tilde{y}_i|x_i) p_{\mathbf{y}_1|\mathbf{x}}(\bar{y}_i|x_i)$

(b) $D(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{n}\sum_{i=1}^{n} D(x_i, \hat{x}_i)$.

Then a sequence of length-n block encoder-decoder pairs can be designed such that if $\mathbf{y}_1$ *is encoded at rate R,* $\mathbf{x}$ *can be recovered to within average distortion d with arbitrarily small probability of failure as n grows to infinity if and only if*

$$R \geq R_{\mathrm{I}}^{\mathrm{WZ}}(d) = \min_{p_{u|y_1}(u|y_1) \in \mathcal{U}} [I(y_1; u) - I(y_0; u)], \tag{3.1}$$

where the set $\mathcal{U}$ *consists of all posteriors relating the random variable* u *to the encoder observation* $\mathbf{y}_1$ *that satisfy the following conditions:*

(i) $x \leftrightarrow y_1 \leftrightarrow u$,

(ii) $y_0 \leftrightarrow x \leftrightarrow u$,

(iii) $E[D(x, f(y_0, u))] \leq d$ *for some memoryless function* $f : \mathcal{Y}_0 \times \mathcal{U} \rightarrow \hat{\mathcal{X}}$.

In the spirit of [15] we can use (3.1) to generate a number of earlier results. Setting $\Pr(y_1 = x | x = x) = 1$ yields the perfect observation case investigated by Wyner and Ziv [84]. Setting the side information to zero ($y_0 = 0$) means that $I(y_0; u) = 0$, which gives us the noisy quantization results developed in [25, 79]

■ 3.1.1 Binary-Hamming Case

In Appendix A.2 we determine $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ for the case of discrete binary-symmetric sources under a Hamming distortion measure. In this case $\mathbf{x}$ is a sequence of i.i.d. Bernoulli random variables: $\Pr(x_i = 1) = p$ and $\Pr(x_i = 0) = 1 - p$. The variables $\mathbf{y}_0$ and $\mathbf{y}_1$ are observations of $\mathbf{x}$ through independent binary-symmetric channels with cross-over probabilities p_0 and p_1, respectively. This results in posterior distributions $p(y_{0,i} \neq x_i) = p_0$ and $p(y_{1,i} \neq x_i) = p_1$ where we have used $y_{0,i}$ and $y_{1,i}$ to denote the ith samples of the observations $\mathbf{y}_0$ and $\mathbf{y}_1$, respectively. To present our results we slightly abuse notation and use $H(p)$ to denote the entropy rate of a Bernoulli random variable x where $\Pr(x = 1) = p$. Using this notation $H(x) = H(p) = -p\log(p) - (1-p)\log(1-p)$. We also use use $*$ to denote binary convolution, i.e., $p * q = p(1-q) + q(1-p)$.

The noisy Wyner-Ziv rate distortion function for this case is derived in Appendix A.2, and depicted graphically in Fig. 3.2. It helps to keep this figure in mind while considering the following analytic expressions for that curve. The rate-distortion function for this case is the lower convex envelope of the function

$$g(d) = H(p_0 * d) - H\left(\frac{d - p_1}{1 - 2p_1}\right), \qquad p_1 \leq d < p_0, \tag{3.2}$$

and the point $(0, p_0)$. The point $(0, p_0)$ can be achieved at zero rate by simply using the side information as the source estimate. The convex combination of $g(d)$ and $(0, p_0)$ results in the rate distortion function

$$R_{\mathrm{I}}^{\mathrm{WZ}}(d) = \begin{cases} \text{unachievable}, & \text{if} d < \min\{p_0, p_1\}, \\ g(d), & \text{if } \min\{p_0, p_1\} \leq d \leq \tilde{d} \\ g(\tilde{d})\left(1 - \frac{d - \tilde{d}}{p_0 - \tilde{d}}\right), & \text{if } \tilde{d} \leq d \leq p_0, \\ 0, & \text{if } d \geq p_0, \end{cases} \tag{3.3}$$

where $\tilde{d}$ is the solution to the equation

$$\frac{d}{ds} g(s)\bigg|_{s=\tilde{d}} = \frac{g(\tilde{d})}{\tilde{d} - p_0}, \tag{3.4}$$

and s is a dummy variable.

Given our discussion of the Wyner-Ziv coding technique in Section 2.1.3, the results of this section can be understood relatively simply. The encoder observation $\mathbf{y}_1$ is vector quantized and then binned. The bin index is sent to the decoder. The side information $\mathbf{y}_0$ is used to pick out the correct quantization vector from the specified bin. At this point the decoder has two pieces of information: the identified quantizer codeword $\mathbf{u}(s)$ and the side information $\mathbf{y}_0$. Both are binary sequences. The decoder must decide how to fuse these two sequences together. While in other scenarios, such as the quadratic-Gaussian, the two pieces of information can be fused together softly, in the binary-Hamming case hard decisions are optimal. The source estimate $\hat{\mathbf{x}}$ is set equal to either $\mathbf{u}(s)$ or $\mathbf{y}_0$, depending on which is more reliable. If $\hat{\mathbf{x}} = \mathbf{u}(s)$, then the side information $\mathbf{y}_0$ is used only in the decoding step, and not in the data fusion step. If $\hat{\mathbf{x}} = \mathbf{y}_0$, then the transmitted bin index is not used in the data fusion step, and so is best not to send in the first place.

From the preceding discussion we can determine the limits of the rate-distortion function. When $d = p_0$, the rate should be zero since the is the channel cross-over probability relating the side information to the source and can be achieved at zero rate by setting $\hat{\mathbf{x}} = \mathbf{y}_0$. On the other hand, the rate-distortion function should go to infinity when $d = \min\{p_0, p_1\}$. This follows because even if we transmit $\mathbf{y}_1$ to the decoder losslessly, the decoder's optimal strategy is simply to set $\hat{\mathbf{x}} = \mathbf{y}_0$ or $\hat{\mathbf{x}} = \mathbf{y}_1$, whichever has a lower cross-over probability and therefore better approximates the source. These limit on the rate-distortion function are reflected in Fig. 3.2.

3.1.2 Quadratic-Gaussian Case

In Appendix A.3 we develop $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ for the quadratic-Gaussian case where the sequences $\mathbf{y}_0$ and $\mathbf{y}_1$ are observations of the i.i.d. Gaussian source vector $\mathbf{x}$ through additive white Gaussian noise channels: $y_{0,i} = x_i + v_{0,i}$, $y_{1,i} = x_i + v_{1,i}$, where $v_{j,i} \sim \mathcal{N}(0, N_j)$ and the two noise sources are independent of each other and of the source. For this problem

$$R_{\mathrm{I}}^{\mathrm{WZ}}(d) = \frac{1}{2} \log \left[\frac{\sigma^2_{x|y_0} - \sigma^2_{x|y_0,y_1}}{d - \sigma^2_{x|y_0,y_1}} \right], \tag{3.5}$$

where $\sigma^2_{x|y_0,y_1} \le d \le \sigma^2_{x|y_0}$ and $\sigma^2_{x|y_0}$ is the minimum mean-squared estimation error in x given y_0, while $\sigma^2_{x|y_0,y_1}$ is similarly defined given both y_0 and y_1.

Investigating some limiting cases to develop intuition, we have that if the encoder noise $\mathbf{v}_1$ equals zero, then $\sigma^2_{x|y_0,y_1} = 0$ and (3.5) is the regular quadratic-Gaussian Wyner-Ziv rate distortion function presented in (2.2). On the other hand, if the side information is absent (or, equivalently the variance of $\mathbf{v}_0$ approaches infinity) then $\sigma^2_{x|y_0} = \sigma^2_x$ and $\sigma^2_{x|y_0,y_1} = \sigma^2_{x|y_1}$. Under these conditions the model is identical to that investigate in [25, 79] for the case of quantization in noise, and (3.5) is equal to the rate-distortion function developed therein.

In determining capacity we must make optimal choices of the $p(u|y_1)$ and f discussed in Thm. 4. The test channel that specifies the relationship between y_1 and u is developed

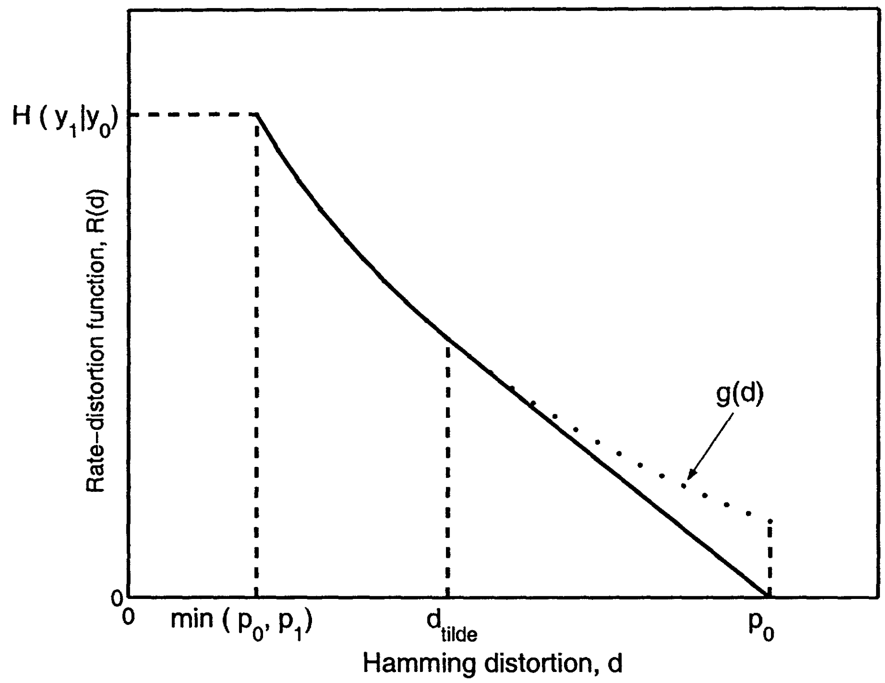


Figure 3.2. The noisy Wyner-Ziv rate distortion function, $R_1^{\mathrm{WZ}}(d)$, for the binary Hamming case is shown with the solid line. The function $g(d)$ is shown by the dotted curve, and the point $\tilde{d}$ is indicated by d_{tilde}.

in Appendix A.3. In brief, the auxiliary random variable $u = \alpha y_1 + e$ where $e \sim \mathcal{N}(0, \alpha d^*)$ is independent of y_0 and y_1. Optimal choices of α and d^* are $\alpha = \sigma^2_{x|y_0} - d/(\sigma^2_{x|y_0} + N_1)$ and $d^* = d - \sigma^2_{x|y_0,y_1}$. In addition, the data fusion function f is

$$f(y_0, u) = \hat{x} = \frac{d}{N_0} y_0 + \left[1 + \frac{N_1}{\sigma^2_{x|y_0}}\right] u. \tag{3.6}$$

Fig. 3.3 illustrates the noisy Wyner-Ziv rate distortion function for the quadratic-Gaussian case. As in the binary-Hamming case the minimal achievable distortion is generally bounded away from zero. The bound is given by the minimal achievable distortion $\sigma^2_{x|y_0,y_1}$ given both observations $\mathbf{y}_0$ and $\mathbf{y}_1$, which constitute a sufficient statistic for estimating of $\mathbf{x}$.

■ 3.2 Noisy Channel Coding with Side Information

In this section we generalize the model of information embedding discussed in Section 2.2 to the situation of noisy host observations, and derive the limits of reliable communication. We term this "writing on dirty paper wearing foggy glasses" since, in the quadratic-Gaussian case, it is a generalization of Costa's work "writing on dirty

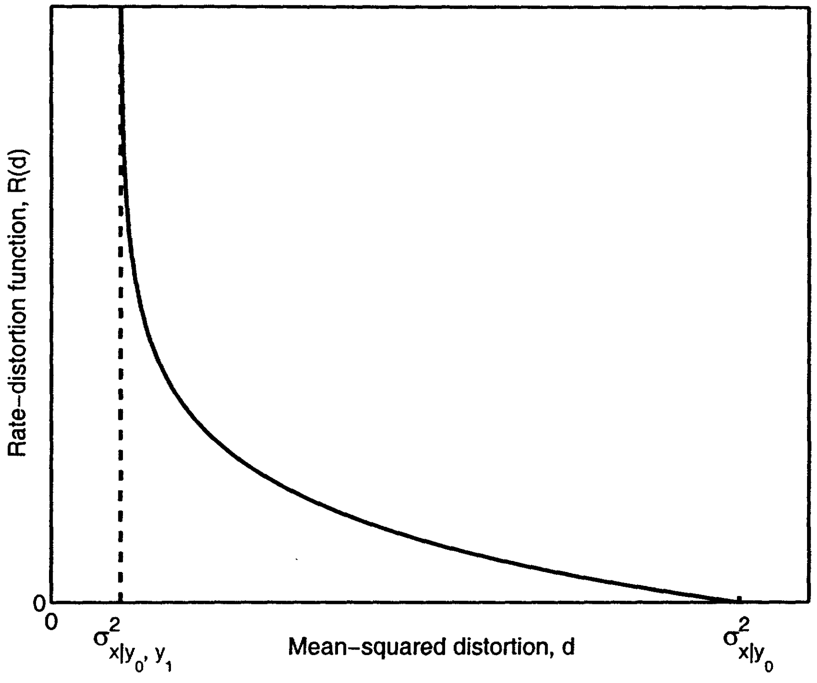


Figure 3.3. The noisy Wyner-Ziv rate distortion function, $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$, for the quadratic-Gaussian case.

paper" [20] to imperfect state information (i.e., as if the state was viewed through foggy glasses).

Fig. 3.4 depicts the scenario of interest. The length-n random vector $\mathbf{y}$ is the encoder's observation, related to the i.i.d. host $\mathbf{x}$ by the memoryless channel law $p(y|x)$. The message to be embedded is m, and the output of the encoder is the embedding signal $\mathbf{e}$. This signal is added to the host $\mathbf{x}$ producing composite signal $\mathbf{w}$. A average distortion constraint $E\left[\frac{1}{n}\sum_{i=1}^{n} D(x_i, w_i)\right]$ is placed between the host and composite signals. Finally, the decoder observes $\mathbf{z}$ which is an observation of $\mathbf{w}$ via the memoryless channel given by $p(z|w)$. The information embedding capacity with noisy or "imperfect" host information is denoted $C_{\mathrm{I}}^{\mathrm{IE}}(d)$. The derivation can be viewed as generalizing earlier results to accommodate imperfect observations, just as [5] generalizes [31] to accommodate a distortion constraint.

Theorem 5 *Let a random pair of sources* $(\mathbf{x}, \mathbf{y})$*, a distortion measure* $D(\cdot,\cdot)$*, and a memoryless channel law* $p(z|w)$ *be given such that*

(a) $p_{\mathbf{x},\mathbf{y}}(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{n} p_{x,y}(x_i, y_i)$,

(b) $D(\mathbf{x}, \mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} D(x_i, w_i)$,

where $\mathbf{w} = \mathbf{x} + \mathbf{e}$ *is the channel input and* $\mathbf{e}$ *is the embedding signal. Then, a sequence of length-n encoder-decoder pairs can be designed such that a message of rate* R *can be*

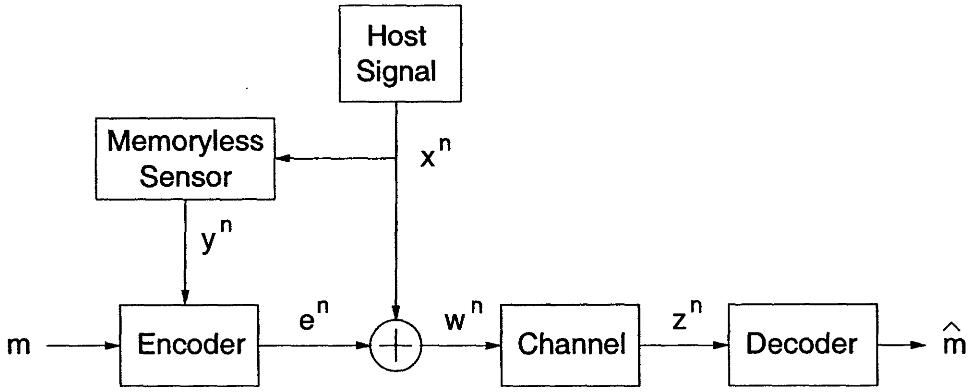


Figure 3.4. Information embedding with noisy host observations. The signals x, y, m, e, w and z are, respectively, the host, host observation, message, embedding signal, composite signal, and channel output.

communicated to the decoder with an arbitrarily small probability of decoding error as n grows to infinity while satisfying the average distortion constraint d if and only if

$$R < C_{\mathrm{I}}^{\mathrm{IE}}(d) = \max_{p_{u|y}(u|y)\in\mathcal{U}} [I(z; u) - I(u; y)] \tag{3.7}$$

where the set $\mathcal{U}$ consists of all posteriors $p(u|y)$ relating the auxiliary random variable u to the host information y that satisfy the two conditions:

(i) $p(u|x, y, e, w, z) = p(u|y)$.

(ii) $E[D(x, w)] \leq d$ *where* $w = x + e$ *and* $e = f(u, y)$ *for some memoryless function* $f : \mathcal{U} \times \mathcal{Y} \rightarrow \mathcal{E}$.

3.2.1 Quadratic-Gaussian Case

In Appendix B.2 we develop $C_{\mathrm{I}}^{\mathrm{IE}}(d)$ for the quadratic-Gaussian case. In this case the vector $\mathbf{y}$ is the encoder's observation of the length-n i.i.d. Gaussian host vector $\mathbf{x}$ through an additive white Gaussian noise channel $\mathbf{y} = \mathbf{x} + \mathbf{v}_0$ where $\mathbf{v}_0 \sim \mathcal{N}(0, N_0\mathbf{I})$. As a function of $\mathbf{y}$ and m, the encoder produces the embedding signal $\mathbf{e}$ giving the channel input $\mathbf{w} = \mathbf{x} + \mathbf{e}$ where the distortion constraint is $E\left[\frac{1}{n}\sum_{i=1}^{n}(x_i - w_i)^2\right] \leq d$. The communication channel is an additive white Gaussian noise channel, $\mathbf{z} = \mathbf{w} + \mathbf{v}_1$ where $\mathbf{v}_1 \sim \mathcal{N}(0, N_1\mathbf{I})$. For this channel, the capacity is

$$C_{\mathrm{I}}^{\mathrm{IE}}(d) = \frac{1}{2}\log\left[1 + \frac{d}{\sigma_{x|y}^2 + N_1}\right]. \tag{3.8}$$

The two terms in the denominator of (3.8) correspond to the two sources of uncertainty. The first term is the mean-squared estimation error in the host estimate $\hat{\mathbf{x}} = E[\mathbf{x}|\mathbf{y}]$. The second term is the uncertainty caused by the channel noise. When the host estimation error $\sigma^2_{x|y}$ is zero (3.8) reduces to the capacity of Costa's problem discussed in (2.6).

Eq. (3.8) also tells us that, unlike for noisy Wyner-Ziv source coding, in the case of Gaussian measurements, a separation theorem applies to noisy information embedding. Without loss in performance the encoder can be realized as the cascade of minimum mean-squared error estimation of the host $\mathbf{x}$ from the observation $\mathbf{y}$ followed by the generation of an embedding signal as if the estimate were the true state. Since the minimum mean-squared estimation error acts as extra channel noise, this must be taken into account if distortion compensation is used in the information embedder [14].

The test channel used to determine capacity is similar to the channel used by Costa in [20]. The auxiliary random variable $u = \alpha y + e$, where e is the embedding signal, $e \sim \mathcal{N}(0, d)$, that is independent of y. The input to the channel is $w = x + e$. Optimization over α yields $\alpha = d/((d + N_0 + N_1) + N_0(d + N_1)/\sigma^2_x)$ which gives the capacity (3.8).[1]

3.3 Geometric Pictures

Just as in Chapter 2, in the quadratic-Gaussian case the rate distortion and capacity results of Section 3.1 and 3.2 can be derived via sphere packing arguments. Such pictures have proven useful in the design of nested lattice codes [5, 89].

3.3.1 Noisy Wyner-Ziv Coding

Fig. 3.5 illustrates how the noisy Wyner-Ziv rate-distortion function in the quadratic-Gaussian case (3.5) can be derived as a sphere covering problem. Based solely on its side information, the decoder can achieve a minimum mean-squared estimation error equal to $\sigma^2_{x|y_0}$. Therefore, before using the message from the encoder, the decoder can determine that the true vector source $\mathbf{x}$ lies within an uncertainty ball of radius $\sqrt{n(\sigma^2_{x|y_0} + \epsilon_1)}$ centered at its source estimate $E[\mathbf{x}|\mathbf{y}_0]$. This large sphere of uncertainty is indicated by the dotted circle in Fig. 3.5.

We now determine the size of the smaller spheres to be packed within the large dotted sphere of uncertainty. It is tempting to guess that the smaller sphere have radius $\sqrt{n(d^* - \epsilon_2)} = \sqrt{n(d - \sigma^2_{x|y_0,y_1} - \epsilon_2)}$, the size of the quantization regions. This is not correct because of the extra uncertainty at the encoder caused by the noisy measurements. To take into account this extra uncertainty we must consider how much the encoded message, once decoded, will contribute to resolving the uncertainty in $\mathbf{x}$. The reconstruction function (3.6) tells us that we must scale $\mathbf{u}$ by $(1+N_1/\sigma^2_{x|y_0})$ when producing $\hat{\mathbf{x}}$. The minimum number of spheres of radius $\sqrt{n(1 + N_1/\sigma^2_{x|y_0})(d - \sigma^2_{x|y_0,y_1} - \epsilon_2)}$

---

[1] A variant of the embedding problem is when the encoder observes a noisy version of the host and must also embed in some function of that noisy observation. For this case the capacity is $C(d) = \frac{1}{2}\log[1 + (d - \sigma^2_{x|y})/N_1]$, where $d - \sigma^2_{x|y} \geq 0$, which is achieved by embedding in the minimum mean-squared error estimate.

needed to cover the large sphere of uncertainty is lower bounded by the ratio of volumes:

$$M \geq \frac{\kappa(n)\left(\sqrt{n(\sigma^2_{\mathsf{x}|\mathsf{y}_0}+\epsilon_1)}\right)^n}{\kappa(n)\left(\sqrt{n(1+N_1/\sigma^2_{\mathsf{x}|\mathsf{y}_0})(d-\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}-\epsilon_2)}\right)^n}, \tag{3.9}$$

This gives a lower-bound on the rate, as a function of d:

$$\begin{aligned}
R(d) &= \frac{1}{n}\log M \geq \frac{1}{2}\log\left[\frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0}+\epsilon_1}{(1+N_1/\sigma^2_{\mathsf{x}|\mathsf{y}_0})(d-\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}-\epsilon_2)}\right] && (3.10)\\
&> \frac{1}{2}\log\left[\left(1-\frac{N_1}{\sigma^2_{\mathsf{x}|\mathsf{y}_0}+N_1}\right)\frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0}}{d-\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}}\right]\\
&= \frac{1}{2}\log\left[\left(1-\frac{1}{\sigma^2_{\mathsf{x}|\mathsf{y}_0}}\left(\frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0}N_1}{\sigma^2_{\mathsf{x}|\mathsf{y}_0}+N_1}\right)\right)\frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0}}{d-\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}}\right]\\
&= \frac{1}{2}\log\left[\left(1-\frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}}{\sigma^2_{\mathsf{x}|\mathsf{y}_0}}\right)\frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0}}{d-\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}}\right] && (3.11)\\
&= \frac{1}{2}\log\left[\frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0}-\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}}{d-\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1}}\right],
\end{aligned}$$

where (3.10) follows by substituting in (3.9) for M, and (3.11) follows from $\sigma^2_{\mathsf{x}|\mathsf{y}_0,\mathsf{y}_1} = \frac{\sigma^2_{\mathsf{x}|\mathsf{y}_0}N_1}{\sigma^2_{\mathsf{x}|\mathsf{y}_0}+N_1}$.

3.3.2 Noisy Information Embedding

Fig. 3.6 illustrates how the noisy information embedding function in the quadratic-Gaussian case (3.8) can be derived as a sphere-packing problem. The host signal, x, lies somewhere in a region of radius $\sqrt{n(\sigma^2_{\mathsf{x}|\mathsf{y}}+\epsilon_1)}$ centered around the host estimate $\hat{\mathsf{x}} = E[\mathsf{x}|\mathsf{y}]$. The distortion constraint restricts the composite signal to be contained in a sphere of radius $\sqrt{n(d-\epsilon_2)}$ centered around the host x. Putting these two components together implies that the composite signal lies in a sphere of radius $\sqrt{n(\sigma^2_{\mathsf{x}|\mathsf{y}}+d+\epsilon_1-\epsilon_2)}$ centered at $\hat{\mathsf{x}}$. The channel noise adds extra uncertainty, leaving the channel output in a sphere of radius $\sqrt{n(\sigma^2_{\mathsf{x}|\mathsf{y}}+d+N_1+\epsilon_1-\epsilon_2+\epsilon_3)}$ centered at $\hat{\mathsf{x}}$. Reliable decoding requires the codewords to lie in disjoint spheres of radius $\sqrt{n(\sigma^2_{\mathsf{x}|\mathsf{y}}+N_1+epsilon_1+\epsilon_3+\epsilon_4)}$. The first term, $\sigma^2_{\mathsf{x}|\mathsf{y}}$, ensures that errors are not introduced by the encoder's uncertainty about the host signal. The second, N_1, ensured that the channel noise does not introduce errors.

In Fig. 3.6 the three dotted spheres illustrate the uncertainty introduced by host estimation error, embedding distortion, and channel noise. The solid circles indicate

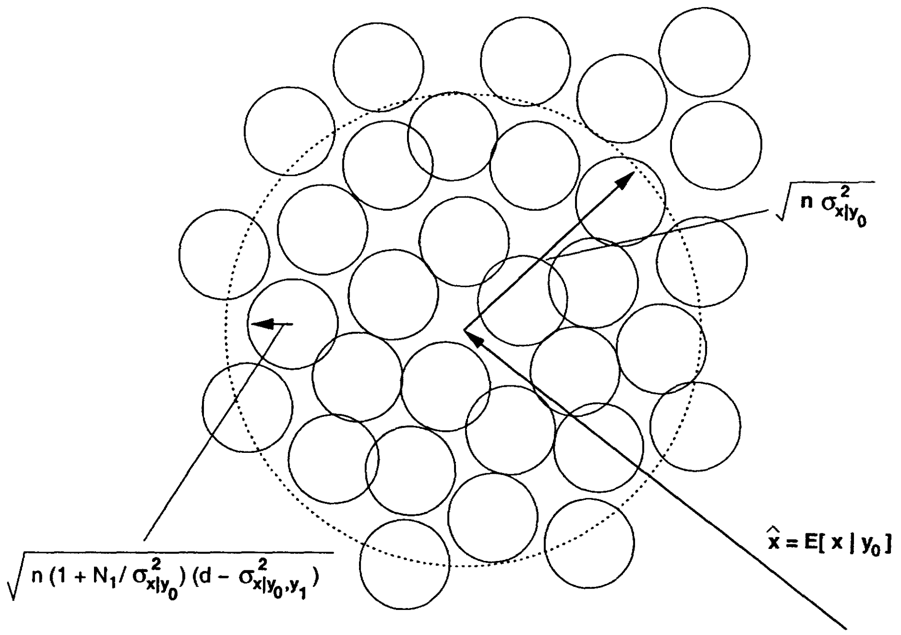

$\sqrt{n\,\sigma^2_{x|y_0}}$
$\hat{x} = E[x \mid y_0]$
$\sqrt{n\,(1 + N_1/\sigma^2_{x|y_0})\,(d - \sigma^2_{x|y_0,y_1})}$

Figure 3.5. The rate distortion function for noisy Wyner-Ziv source coding can be found via sphere covering arguments. The dotted circle correspond to the source uncertainty given the side information, $\sigma^2_{x|y}$. The solid circles correspond to the quantization regions at the encoder $(d - \sigma^2_{x|y_0,y_1})$ that must be scaled up by $(1 + N_1/\sigma^2_{x|y_0,y_1})$ because of the extra uncertainty at the encoder caused by the noisy observations.

codeword spheres. The ratio of volumes of the largest dotted circle to that of the solid circles gives an upper bound on the number of uniquely decodable codewords. The maximum number of codewords M that can be transmitted reliably is upper bounded by the ratio of volumes:

$$M \leq \frac{\kappa(n)\left(\sqrt{n(\sigma^2_{x|y} + d + N_1 + \epsilon_1 + \epsilon_3 - \epsilon_2)}\right)^n}{\kappa(n)\left(\sqrt{n(\sigma^2_{x|y} + N_1 + \epsilon_1 + \epsilon_3 + \epsilon_4)}\right)^n} < \left(1 + \frac{d}{\sigma^2_{x|y} + N_1}\right)^{n/2},$$

where $d \geq 0$. This gives an upper-bound on the rate,

$$R = \frac{1}{n}\log M < \frac{1}{2}\log\left(1 + \frac{d}{\sigma^2_{x|y} + N_1}\right),$$

which is equal to the noisy information embedding capacity (3.8) derived in Section 3.2.1.

■ 3.4 Chapter Summary

In this chapter we generalize the coding with side information problem of presented in Chapter 2 to noisy encoder observations. We first do this for the Wyner-Ziv prob-

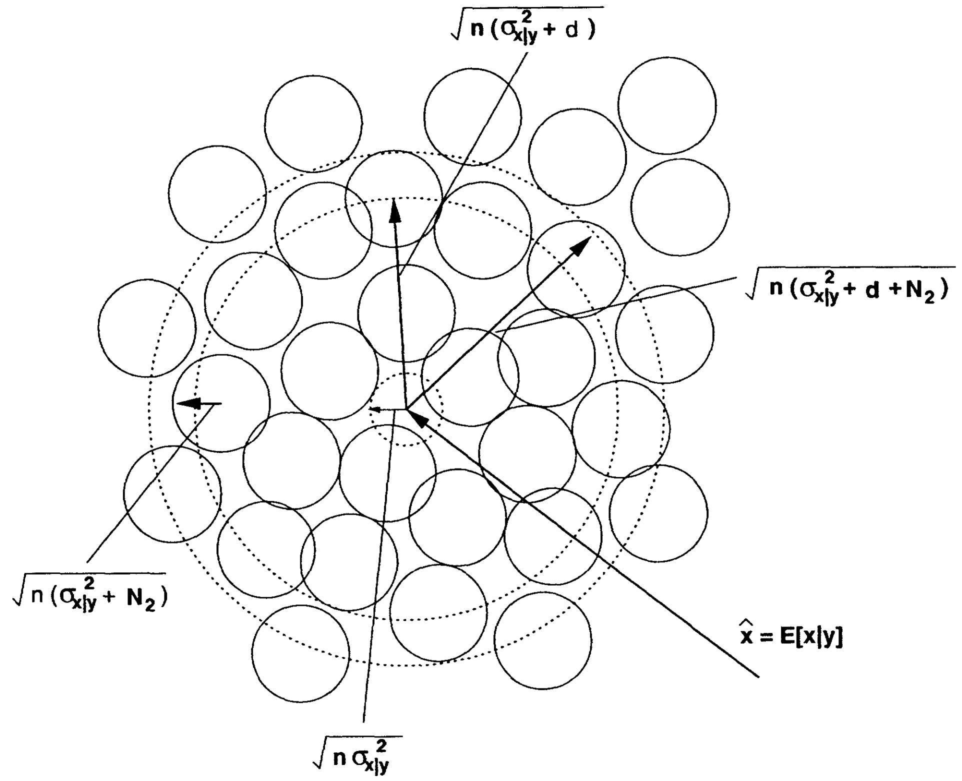

$\sqrt{n(\sigma_{x|y}^2 + d)}$
$\sqrt{n(\sigma_{x|y}^2 + d + N_2)}$
$\sqrt{n(\sigma_{x|y}^2 + N_2)}$
$\hat{x} = E[x|y]$
$\sqrt{n\sigma_{x|y}^2}$

Figure 3.6. The capacity for noisy information embedding can be derived from sphere packing arguments. The three concentric dotted circles correspond to: 1) uncertainty in state, $\sigma_{x|y}^2$, 2) state uncertainty + allowable introduced distortion, d, and 3) state uncertainty + distortion + channel noise, N_1. We cover the largest sphere with smaller (solid) spheres that correspond to codewords. Each small sphere is of radius $\sqrt{n(\sigma_{x|y}^2 + N_1)}$ to ensure reliable decoding.

lem and develop the rate-distortion function for finite-alphabet sources with arbitrary distortion measures. We then evaluate this function for the binary-Hamming and quadratic-Gaussian cases and discuss the resulting expressions. We then generalize the information embedding problem to noise host observations and develop the capacity expression in the finite-alphabet and arbitrary distortion measure case. We evaluate this function for the quadratic-Gaussian case. This analysis tells us that a separation theorem applies in the quadratic-Gaussian case: we can first to estimate the host and then to design our embedding signal as if the estimate were the true host, estimation error counts as extra channel noise. Finally, we show how the rate-distortion and capacity expressions for each problem can be derived through sphere-packing arguments in the quadratic-Gaussian case.

Chapter 4

Successively Structured Data Fusion Algorithms for Sensor Networks

In this chapter we develop source coding strategies for sensor networks. We base these strategies on the generalizations of source coding with side information to noisy encoder observations developed in Chapter 3. We consider "CEO" problems where the network goal is to provide a particular network node – the CEO – with the best possible estimate of the source under various rate constraints. We structure successive algorithms that are flexible enough to deal with the distributed nature of the data in sensor networks. We show that these algorithms can be used to achieve the rate-distortion function for some useful network configurations. The successively structured design also gives a new approach to relay channel communications.

In Section 4.1 we present the probabilistic model of sensor networks that we will be using, and discuss two prototype sensor network problems that we term the parallel and serial CEO problem. In Section 4.2 we discuss earlier information theoretic results for these problems. In Section 4.3 we discuss approaches to these prototype network problems that obey a layered network architecture, while in Section 4.4 we explain how to refine these approaches through inter-layer optimization. In particular, we draw upon insights on the use of decoder side information from Chapter 3. We then present successively structured coding techniques for the two prototype problems that are, in a sense, dual. In Section 4.5 we analyze the resultant performance for the serial problem, and in Section 4.6 for the parallel problem. In Section 4.7 we apply the successive coding approaches to certain classes of relay channels. We close the chapter with a summary of our results in Section 4.8.

■ 4.1 System Model: Sensor Network, Finite-Rate Communications

In this section we describe the probabilistic model of sensor networks with which we work in this chapter. Fig. 4.1 illustrates anew the sensor network discussed in the Introduction. The black node represents the source signal that we want to estimate. The source x is modeled as a length-n independent identically distributed random vector $p_{\mathsf{x}}(\mathbf{x}) = \prod_{i=1}^{n} p_{\mathsf{x}}(x_i)$. Each open circle represents one of L sensor nodes ($L = 8$ in the figure). Node l measures y_l which is related to the source x by the memory-

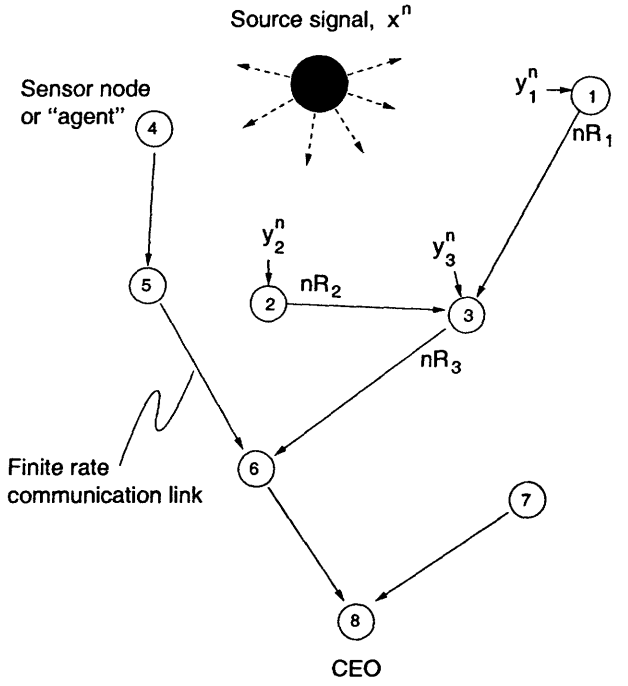


Figure 4.1. A general sensor network with finite-rate links and tree-structured communications.

less channel law $p_{\mathbf{y}_l|\mathbf{x}}(\mathbf{y}_l|\mathbf{x}) = \prod_{i=1}^{n} p_{y_l|x}(y_{l,i}|x_i)$. The channels are assumed independent, meaning that the joint distribution can be factored as $p_{\mathbf{x},\mathbf{y}_1,\ldots\mathbf{y}_L}(\mathbf{x},\mathbf{y}_1,\ldots,\mathbf{y}_L) = \prod_{i=1}^{n}\left[p_x(x_i)\prod_{l=1}^{L} p_{y_l|x}(y_{l,i}|x_i)\right]$. In other words, the observations $\mathbf{y}_1,\ldots\mathbf{y}_L$ are conditionally independent given the source $\mathbf{x}$. This important assumption is needed to make our proof techniques work. It is also a standard assumption in the distributed detection and estimation literature (see, e.g., [76, 75]). We next discuss why the model we have presented is not always well matched to sensor network problems.

Example: Model Matching. The model for sensor networks presented above has three major features: conditional independence of observations, memoryless source-observation relationships, and the same underlying source is observed by all sensors. The conditional independence assumption is not a good model for situations where, e.g., an interferer results in correlated noise processes across the sensor array. Such an interferer might also produce noise that is strongly correlated temporally. These situations are not well modeled by the memoryless assumption. Finally, the underlying source signals observed by the sensor nodes may differ. For example, in underwater sonar arrays, each node may observe a time-shifted version of the source signal. □

Together with an observation model we need a model for inter-node communication. We assume that each node is given a finite rate, in terms of bits (or nats) per observation sample, at which it can communicate reliably to one other node in the network. Depending on whether the communications structure is flexible, or fixed, each node may, or may not, get to choose to which node it communicates. In Fig. 4.1, for example, node 1 can communicate to node 3 at rate R_1. We choose this fixed-rate communication model rather than a probabilistic channel-law model so as to focus in on rate-distortion trade offs. This fixed-rate model simplifies out many interesting communication phenomena that have been investigated by other researchers. Interference between node-to-node communications that would be present in, e.g., a wireless scenario is one such phenomenon that is investigated in [38]. A second phenomenon that has been investigated [22, 59, 60] is to use source correlations to transmit correlated codewords that constructively interfere in the multiple-access channel to give, e.g., in the Gaussian multiple-access channel, a power gain.

Given the observation and communications models, we now describe the knowledge and resource constraints under which the algorithms that we design must operate. First, all nodes are assumed to know the full joint statistical description of the source and observations $p_{\mathbf{x},\mathbf{y}_1,\ldots,\mathbf{y}_L}(\mathbf{x},\mathbf{y}_1,\ldots,\mathbf{y}_L)$. Second, the nodes are assumed not to have any processing constraints. Third, the CEO is assumed not to have any delay constraints, so n can be very large. The first assumption is required so that, at a minimum, we know how the observations related to one another and can design data fusion techniques. In a deployed network such statistical knowledge may be gained through a training phase. As it turns out, assuming full joint statistical knowledge is somewhat more than we will always need, e.g., leaf nodes need to know less than nodes closer to the CEO. The latter two assumptions allows us to concentrate on the ultimate trade offs between estimation and communication rather than effects resulting from constraints on processing or delay.

The algorithms we develop work in a multi-step manner, sometimes referred to as "block-Markov" in the information theory literature. The start of communications between nodes is delayed until all sensors have observed their full vector of observations. At that point, communication starts at the leaves (nodes 1,2,4,7 in Fig. 4.1), which block-encode their observations and send messages through the communication tree toward the CEO (node 8 in the figure). Each non-leaf node (nodes 3,5,6 in Fig. 4.1) in the tree waits until it has received messages from all incoming branches (i.e., all branches except the one leading to the CEO). Once it has received all these messages, the node determines what message to send on toward the CEO. The CEO waits until he receives all incoming messages before he making the final source estimate. Readers familiar with the Belief Propagation [49] or the sum-product [44] algorithms will recognize this as a similar scheduling algorithm.

Decomposing a Tree into Parallel and Serial CEO Problems. Instead of directly addressing general tree-structured sensor network, we instead concentrate on two prototype networks. Any tree, such as the one depicted in Fig. 4.1, can be decomposed into a collection of smaller parallel and serial prototype networks. A parallel network has a hub-and-spoke structure such as is displayed by node groupings (1,2,3), (3,5,6) and (6,7,8) in Fig. 4.1. A serial network has a data-pipeline or chain structure, such as displayed by node grouping (4,5,6). These prototype parallel and serial networks are

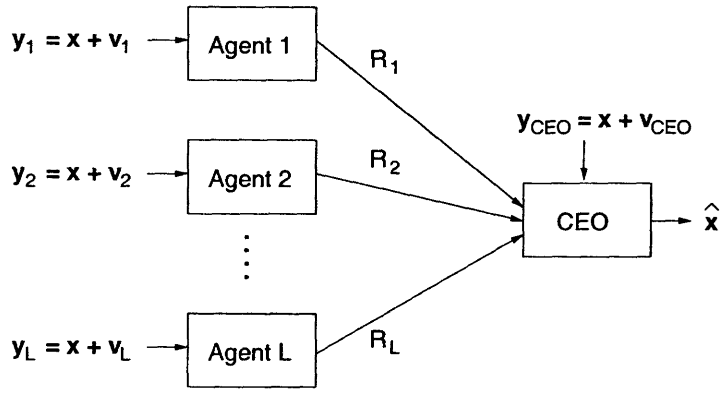


Figure 4.2. The parallel CEO problem with additive noise observations. The signals $\mathbf{x}$, $\mathbf{y}_l$, and $\mathbf{v}_l$ are the source, observation, and noise vectors, respectively. Agent l communicates to the CEO at rate R_l. The CEO fuses the data to produce a source estimate $\hat{\mathbf{x}}$. Generally the CEO has its own observation observation $\mathbf{y}_{\text{CEO}}$. If the CEO does not have an observation, set $\mathbf{y}_{\text{CEO}} = 0$.

the basic network configurations we consider. We term them the "parallel" and "serial" CEO problems, respectively.

In Fig. 4.2 we diagram the parallel CEO problem for additive noise observations. In the parallel problem the central data fusion site is the CEO. The parallel network configuration is a good model for situation where a number of agents are reporting in to a central estimation center or, alternately, when a single agent is reporting in at a succession of time steps. At time step one the agent observes $\mathbf{y}_1$, at time step two $\mathbf{y}_2$, and so on. In the current setting $\mathbf{x}$ is assumed constant, but in a more general setting the agent could observe a dynamically changing source, e.g., $\mathbf{y}_l = \mathbf{x}_l + \mathbf{v}_l$ in an additive noise scenario.

In Fig. 4.3 we diagram the serial CEO problem for additive noise observations. In the serial CEO problem the agents are ordered and transmit in turn — one to the next (i.e., in series) — over rate-constrained links. The last agent in the chain is the CEO. This is a good model for a sensor pipeline or, alternately for a single sensor making sequential vector-measurements of $\mathbf{x}$ over time. If the agent has a limited memory he can assign to storing $\hat{\mathbf{x}}_l$, we could model his memory limitations as a rate-constrained channel. In that case instead of sending the message to the next agent, he instead writes it to his memory, and reads it out to do data fusion at the next time step.

In Section 4.4 we will show how to decompose further each of the prototype problems into a sequence of basic data fusion encoding and decoding blocks based on the noisy Wyner-Ziv results. As we discuss in the next section, the parallel CEO problem has been investigated before, while the serial problem is new.

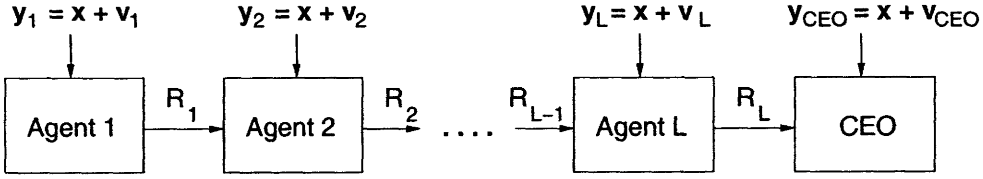


Figure 4.3. The serial CEO problem with additive noise observations. The signals $\mathbf{x}$, $\mathbf{y}_l$, and $\mathbf{v}_l$ are the source, observations, and noise vectors, respectively. Agent $l-1$ communicates to agent l at rate R_{l-1}. The CEO has his own observation $\mathbf{y}_{\text{CEO}}$.

■ 4.2 Literature Review: CEO Problems

The CEO problem introduced by Berger et. al. [9] falls into the class of sensor network data fusion problems that we have termed parallel CEO problems. The slight specialization in the original CEO problem is that the CEO does not have his own source observation. In the original CEO paper [9] finite-alphabet sources were considered. In subsequent work [75, 45] the quadratic-Gaussian problem was considered. This is the case where $\mathbf{x} \sim \mathcal{N}(0, \sigma_x^2 \mathbf{I})$, $\mathbf{y}_l = \mathbf{x} + \mathbf{v}_l$ where $\mathbf{v}_l \sim \mathcal{N}(0, N_l \mathbf{I})$, and the network goal is to minimize the mean-squared distortion between $\mathbf{x}$ and $\hat{\mathbf{x}}$, i.e., $d = E\left[\|\mathbf{x} - \hat{\mathbf{x}}\|^2\right]$.

In all three papers [9, 75, 45], the objective is to determine the minimal achievable distortion under a sum-rate constraint, $\sum_{l=1}^{L} R_l \leq \bar{R}$, as the number L of agents grows to infinity. The coding approach proposed in [75] is representative. It is a three step process. First, all agents block-encode their observations into messages using identical quantizers. They then transmit their messages to the CEO using Slepian-Wolf coding techniques which enables them to avoid sending redundant information. Finally, the CEO decodes all the messages jointly and makes a source estimate. This joint decoding structure contrasts with the successive decoding structure we propose in this thesis where agents' messages are decoded sequentially, increasing the fidelity of the estimate at each decoding step.

In [75] the authors show that under a sum-rate constraint the CEO's estimation error decreases at best as R^{-1}, even as the number of agents tends toward infinity. This contrasts with the case where the agents are allowed to convene and pool their data to jointly estimate the source before sending a joint message to the CEO. In this scenario an exponential decrease in estimation error as a function of R can be achieved. In [45] Oohama determined that the rate-distortion function for this asymptotic (in the number of agents) version of the quadratic-Gaussian parallel CEO problem is

$$R(d) = (\log_2 e)\frac{N}{2\sigma_x^2}\left[\frac{\sigma_x^2}{d} - 1\right] + \frac{1}{2}\log_2\left[\frac{\sigma_x^2}{d}\right], \qquad \text{(bits/sample)} \qquad (4.1)$$

where $0 \leq d \leq \sigma_x^2$. In our work we can reproduce this result using successively structured codes as discussed in Section 4.6.3. Furthermore, the successive coding structure we propose allows us to analyze the finite-L region. We derive achievable distortion-rate trade offs in this region and, for the case of two agents ($L = 2$), we are able to use Oohama's converse from [45] to show that our coding structure can be used to achieve the distortion-rate bound for this problem. We discuss the two agent case in Section 4.6.3.

In recent correspondence with Professor Oohama, we have learned that he has also been investigating the finite-L region. In [46] he has extended his earlier results of [45] to show that the rate-distortion function for L equal-SNR agents is

$$R_L(d) = -\frac{L}{2}\log\left[1 - \frac{2}{L}\frac{N}{2\sigma_{\mathsf{x}}^2}\left(\frac{\sigma_{\mathsf{x}}^2}{d} - 1\right)\right] + \frac{1}{2}\log_2\left[\frac{\sigma_{\mathsf{x}}^2}{d}\right], \qquad \text{(bits/sample)} \qquad (4.2)$$

where again $0 \leq d \leq \sigma_{\mathsf{x}}^2$. Our results and Oohama's new results appear to be the only results for the finite-L region. They match for the two agent case, and his also give the new rate-distortion bound for $2 < L < \infty$.

4.3 Estimate-and-Quantize: A Network Layered Approach

We now develop two basic approaches, one for each type of CEO problem. Both approaches are based on a simple estimate-and-quantize idea. These coding strategies can be used in a layered network architecture. Developing these approaches will help highlight the extra degrees of design freedom we use in our inter-layer approaches.

4.3.1 Parallel Network

First, consider the parallel CEO problem in a quadratic-Gaussian context where $\mathbf{x}$ is a white Gaussian source, $R_l = R$ for all l, and $\mathbf{y}_l = \mathbf{x} + \mathbf{v}_l$ where $\mathbf{v}_i \sim \mathcal{N}(0, N\mathbf{I})$, i.e., the agents have equal signal-to-noise rations (SNRs). A basic approach to this problem is: 1) Estimate the source at each agent $\hat{\mathbf{x}}_l = E\left[\mathbf{x}|\mathbf{y}_l\right]$. 2) Vector-quantize the result to get $\hat{\hat{\mathbf{x}}}_l$. 3) Transmit the corresponding quantizer index $m_l \in \{1, \ldots, 2^{nR_l}\}$ to the CEO. 4) Reconstruct the quantized estimates $\hat{\hat{\mathbf{x}}}_1, \ldots \hat{\hat{\mathbf{x}}}_L$ at the CEO. 5) Determine $\hat{\mathbf{x}}$ as a weighted average of the quantized estimates $\hat{\hat{\mathbf{x}}}_1, \ldots, \hat{\hat{\mathbf{x}}}_L$ and the CEO's own observation $\mathbf{y}_{\mathrm{CEO}}$. The distortion achieved after the first l agents have reported in to the CEO is defined as d_l.

We derive a bound on the performance of this algorithm by making two assumptions. First, we assume that the quantization errors made by each agents are independent. Second, we assume that in high dimensions quantization effects are closely approximated by the test channel statistics. There are l messages at the decoder. Each message corresponds to one quantized source estimate. Since the source observations all have equal SNRs, the messages should all be given equal weight. From estimation theory we know that the minimum mean-squared estimation error of a Gaussian source of variance σ_{x}^2 in additive Gaussian noise is

$$d_l = \frac{1}{\frac{1}{\sigma_{\mathsf{x}}^2} + \frac{1}{N} + \frac{l}{N_{\mathrm{msg}}}}, \qquad (4.3)$$

where N is the known noise variance of the CEO's observation and N_{msg} is the effective noise power on the observations. To determine N_{msg} we use the rate-distortion achieving test channel for quadratic-Gaussian source. This test channel is $x = \hat{x} + v$ where $\hat{x} \sim \mathcal{N}(0, \sigma_x^2 - d)$ and $v \sim \mathcal{N}(0, d)$ is independent of $\hat{x}$. We can reverse this test channel to view $\hat{x}$ as the output: $\hat{x} = (1 - d/\sigma_x^2)x + \bar{v}$ where $\bar{v} \sim \mathcal{N}(0, (1 - d/\sigma_x^2)d)$. This doesn't quite look like and additive Gaussian noise observation because of the factor multiplying x. To get an additive Gaussian noise observation $\hat{x}$, define $\tilde{x} = \hat{x}/(1 - d/\sigma_x^2) = x + \tilde{v}$ where $\tilde{v} \sim \mathcal{N}(0, \sigma_x^2 d/(\sigma_x^2 - d))$ and let $N_{\mathrm{msg}} = \sigma_x^2 d/(\sigma_x^2 - d))$.

The value of the d parameter in the definition of N_{msg} should be set equal to the mean-squared distortion achieved when quantizing noisy sources [25, 79]. As we discussed in Section 3.1.2, noisy quantization is a special case of noisy Wyner-Ziv coding when there is no side information. Using the noisy Wyner-Ziv distortion-rate function, which can be derived from the rate-distortion function 3.5, in this special case we get $d = \sigma_{x|y}^2 + (\sigma_x^2 - \sigma_{x|y}^2)2^{-2R}$, where $\sigma_{x|y}^2$ is the same for all agents because they have equal SNRs. Substituting the value for d into the definition of N_{msg} and the result into (4.3) gives us

$$\begin{aligned} d_l &= \frac{\sigma_x^2 N[\sigma_{x|y}^2 + (\sigma_x^2 - \sigma_{x|y}^2)2^{-2R}]}{l\sigma_x^2 N + (\sigma_x^2 + (1-l)N)(\sigma_{x|y}^2 + (\sigma_x^2 - \sigma_{x|y}^2)2^{-2R})} \\ &= \frac{\sigma_x^2\,(1 + \mathrm{SNR}\,2^{-2R})}{l(1 + \mathrm{SNR}) + (1 + \mathrm{SNR} - l)\,(1 + \mathrm{SNR}\,2^{-2R})}, \end{aligned} \tag{4.4}$$

where $\mathrm{SNR} = \frac{\sigma_x^2}{N}$.

4.3.2 Serial Network

Using the same quadratic-Gaussian model as for the parallel problem, in this section we introduce estimate-and-quantize approach for serial CEO problems: 1) Start with agent $l-1$'s estimate $\hat{\mathbf{x}}_{l-1}$. 2) Vector-quantize $\hat{\mathbf{x}}_{l-1}$ to $\hat{\hat{\mathbf{x}}}_{l-1}$. 3) Transmit the corresponding quantizer index m_{l-1} to agent l. 4) Reconstruct the quantized estimate $\hat{\hat{\mathbf{x}}}_{l-1}$ at agent l. 5) Estimate the source based on the quantized estimate $\hat{\hat{\mathbf{x}}}_{l-1}$ and agent l's observation $\mathbf{y}_l$, producing the source estimate $\hat{\mathbf{x}}_l$. The distortion in $\hat{\mathbf{x}}_l$ is defined to be d_l. 6) At the end of the chain the CEO receives message m_L from agent L and fuses it together with his observation $\mathbf{y}_{\mathrm{CEO}}$ to produce d_{L+1} the final source estimate.

The derivation of the distortion this strategy yields resembles the derivation for the parallel problem. It is somewhat simpler, however, because at each stage we have only two pieces of information: the quantized estimate from the last agent and the current agents own observation. Assuming these are independent additive white Gaussian source observations would give us

$$d_l = \frac{1}{\frac{1}{N_{\mathrm{msg}}} + \frac{1}{N} + \frac{1}{\sigma_x^2}}, \tag{4.5}$$

where N_{msg} is the effective noise power on the (single) quantized source estimate $\hat{\mathsf{x}}_{l-1}$. The derivation is similar to above with $N_{\text{msg}} = \sigma_{\mathsf{x}}^2 d/(\sigma_{\mathsf{x}}^2 - d))$ except that now $d = d_{l-1} + (\sigma_{\mathsf{x}}^2 - d_{l-1})2^{-2R}$ because agent $l-1$ has an estimate of quality d_{l-1} which is better than the basic estimate of quality $\sigma_{\mathsf{x}|\mathsf{y}}^2$. This gives us an iterative expression in l, namely

$$d_{l+1} = \frac{Nd}{N+d} = \frac{N\left[d_{l-1} + (\sigma_{\mathsf{x}}^2 - d_{l-1})2^{-2R}\right]}{N + \left[d_{l-1} + (\sigma_{\mathsf{x}}^2 - d_{l-1})2^{-2R}\right]} = \frac{N\left[\frac{d_{l-1}}{N} + \left(\text{SNR} - \frac{d_{l-1}}{N}\right)2^{-2R}\right]}{1 + \left[\frac{d_{l-1}}{N} + \left(\text{SNR} - \frac{d_{l-1}}{N}\right)2^{-2R}\right]}, \tag{4.6}$$

where $\text{SNR} = \frac{\sigma_{\mathsf{x}}^2}{N}$.

■ 4.3.3 Comparison

In Fig. 4.4 and 4.5 we plot the performance of the estimate-and-quantize approaches for the parallel and serial problems, respectively. A bound on the achievable distortion-rate curves for these strategies is plotted with the dashed curve in each figure and is given by the estimation error in x when the observations $\mathsf{y}_1, \dots, \mathsf{y}_l$ are all available to the CEO losslessly. This bound is $\sigma^2_{\mathsf{x}|y_{\text{CEO}}, y_1, \dots y_l}$ for the parallel problem and $\sigma^2_{\mathsf{x}|y_1, \dots y_l}$ for the serial problem. The difference between the bounds arises from the differing definitions of d_l. In the parallel problem d_l is the distortion in the CEO's estimate of x after the first l agents have reported in. In calculating this estimate the CEO can also use his own observation y_{CEO}. In the serial problem d_l is defined as the distortion at agent l, and so the CEO's observation y_{CEO} is not used until he received the final message from agent L. At that point the two bounds are equal. The difference between the bounds is further reflected in the limits of the horizontal axes in Fig. 4.4 and 4.5. In the parallel problem, Fig. 4.4, the axis starts at 0 since the CEO has an estimate based on its own observation even before receiving any messages. In the serial problem, Fig. 4.5, the axis starts with the first agent, $l = 1$.

The performance of the estimate-and-quantize strategies as given by (4.4) and (4.4) are plotted by the dash-dot curves for each network configuration. In the figures the dashed bounds and the dash-dotted estimate-and-quantize performance curves sandwich solid curves labeled 'successive codes'. These curves indicate the performance achieved by the inter-layer coding strategies based on noisy Wyner-Ziv source coding that we introduce in Section 4.4. To help motivate these coding strategies we discuss the estimate-and-quantize strategies in terms of the Kalman filter.

■ 4.3.4 Kalman Filtering with Rate Constraints

One way to interpret both estimate-and-quantize strategies is through analogy with the Kalman filter. The discrete-time Kalman filter consists of two steps: prediction and update. Between time samples the state is driven by an unknown process noise, the effect of which the Kalman filter attempts to predict. Then, at the next time step, the predicted state estimate is updated using the new observation. Let us first consider

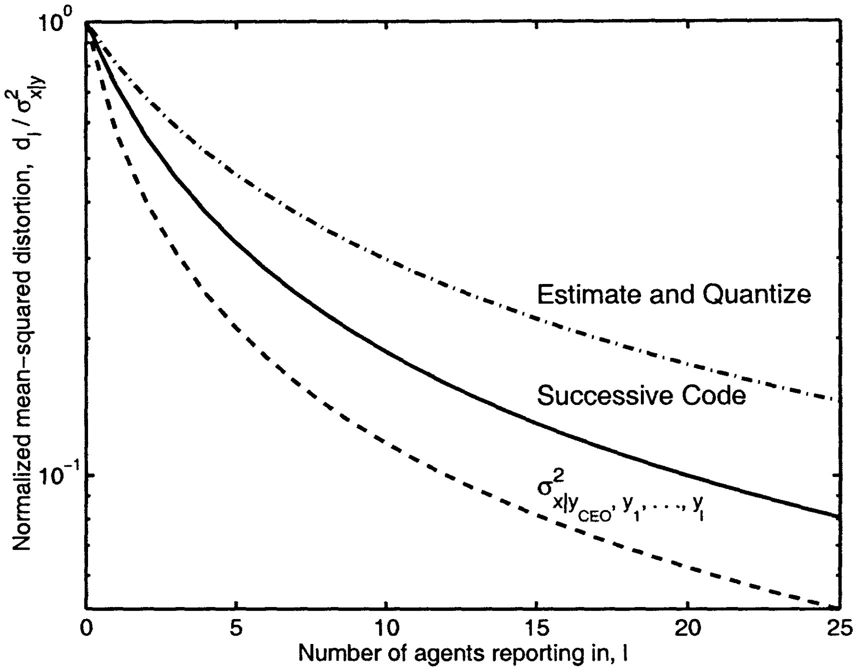


Figure 4.4. The parallel CEO problem, quadratic-Gaussian case. Estimate-and-quantize performance is plotted with the dash-dotted curve. A (generally loose) lower bound given by $\sigma^2_{x|y_{\mathrm{CEO}}, y_1, \ldots, y_l}$ is plotted with the dashed curve. The performance of the successively structured codes proposed in Section 4.4 is plotted with the solid curve.

the serial estimate-and-quantize strategy in this light. Let us think of each agent in the chain as representing a single time step. Then, the quantization error introduced at each communication step in the serial CEO algorithm is akin to the prediction error caused by the process noise in the Kalman filter. The distortion reduction in the data fusion step of the serial CEO algorithm is akin to estimation error reduction in the Kalman filter update step.

Turning to the parallel CEO algorithm, the interpretation has a slightly different flavor. For the parallel algorithm, because the source estimates are never requantized (they are all collocated at the CEO), the process noise is zero. The observations are quantized independently, introducing quantization error on top of the estimation error which means that the quality of data for the data fusion (update) step is reduced. Since there is nothing akin to process noise, the parallel CEO problem is like using a Kalman filter to estimate a constant state value, rather than a time-dependent process.

While useful, the analogy between the Kalman filter and the sensor network problems that we consider is not quite right. The place where the analogy runs into trouble is in drawing an equivalence between quantization error and process noise (in the parallel case) or estimation error (in the serial case). Unlike process or estimation error, quantizer error is a function of the encoder design that we decide to use. In the next section

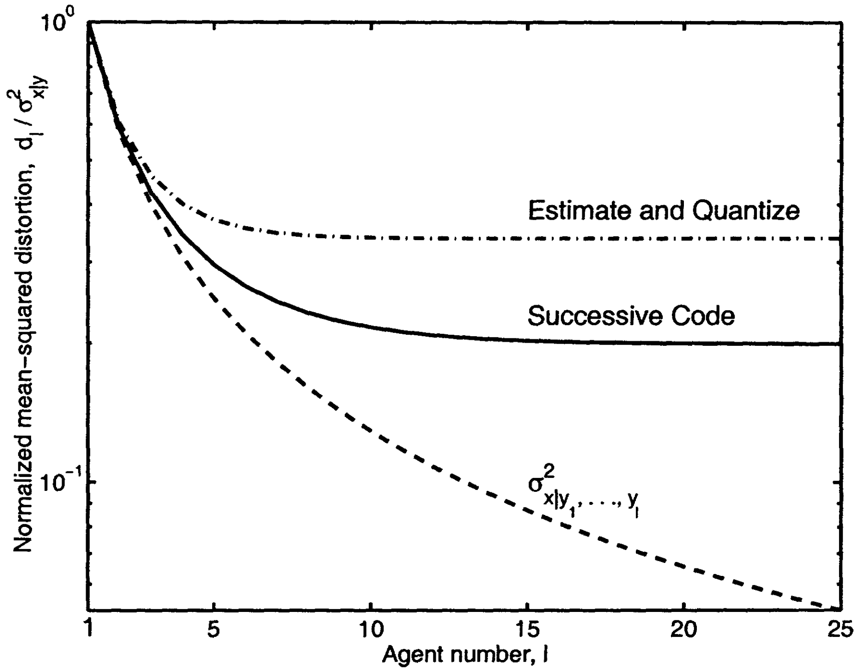


Figure 4.5. The serial CEO problem, quadratic-Gaussian case. Estimate-and-quantize performance is plotted with the dash-dotted curve. A (generally loose) lower bound given by $\sigma^2_{x|y_1,\ldots,y_l}$ is plotted with the dashed curve. The performance of the successively structured codes proposed in Section 4.4 is plotted with the solid curve.

we discuss how to use our control over encoder design to shape the quantization error in such a way that we can better exploit decoder side information to increase system performance over that achieved by the estimate-and-quantize approaches. In contrast to the estimate-and-quantize coding schemes that obey network layering principals, the approaches of the next section work across layers through coupled source and channel coding.

■ 4.4 Inter-Layer Approaches that use Decoder Side Information

When considering the inter-layer approaches, it is somewhat more natural to begin with the serial problem. Each agent in the chain must combine the data sent by the previous agent with his own observation. He must also decide what information to send on to the next agent. To most help agent $l+1$, agent l should send the message that most reduces what he thinks that agent's estimation error is. This is akin to the noisy Wyner-Ziv problem: agent l (the encoder) must bases its transmission on its source estimate $\hat{\mathbf{x}}_l$ an imperfect representation of the source, while agent $l+1$ (the decoder) has side information given by its observation $\mathbf{y}_{l+1}$. This visualization of the problem is diagrammed in the block diagram of Fig. 4.6.

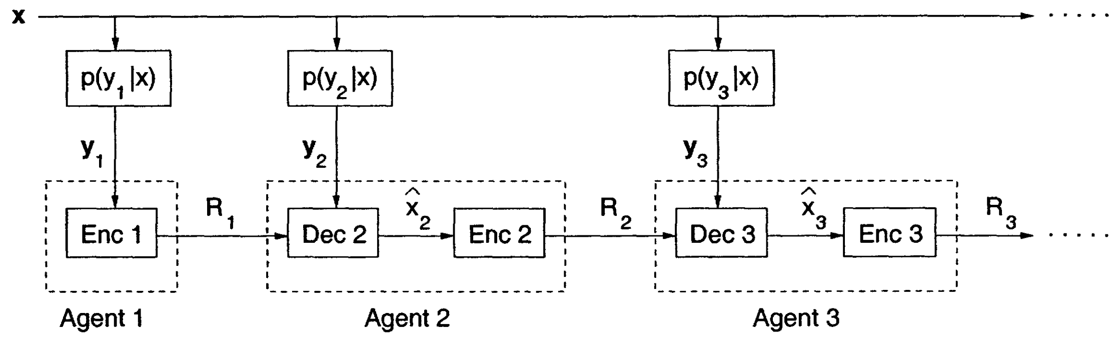


Figure 4.6. The serial CEO problem can visualized as a succession of L noisy Wyner-Ziv stages. At stage l, $\hat{x}_{l-1}$ is the encoder observation, and y_l is the side information.

This approach to the problem allows us to increase each agent's quantization rate, while keeping his communication rate fixed. The effective increase in quantization rate that is enabled by the use of decoder side information is equal to the second, negative, term in the noisy Wyner-Ziv rate-distortion function $R_{\mathrm{I}}^{\mathrm{WZ}}(d) = \min[I(y_1; u) - I(y_0; u)]$ presented in (3.1). In terms of the Kalman filter interpretation presented in Section 4.3, the new algorithms reduce the increase in uncertainty during each prediction step because by using the side information we can use higher-rate quantizers.

Now, consider the parallel CEO problem. Say the CEO decodes each agent's message sequentially. After each decoding step the CEO can use the new message to improve his source estimate. Given the CEO has decoded messages from agents $1, \ldots, l$, agent $l+1$ should take into account the quality of this estimate when deciding what information to send. This is akin to the Wyner-Ziv problem: agent $l+1$ (the encoder) must base its transmission on its noisy source observation $\mathbf{y}_{l+1}$, while the CEO (the decoder) has side information given by its estimate $\hat{\mathbf{x}}_l$. This visualization of the problem is diagrammed in the block diagram of Fig. 4.7.

While in the serial CEO problem the estimate $\hat{\mathbf{x}}_l$ plays the role of encoder measurement, in the parallel CEO problem it plays the role of decoder side information. Conversely, in the serial CEO problem the observation $\mathbf{y}_l$ plays the role of decoder side information, while it plays the role of encoder information in the parallel CEO problem. In this sense the approaches are dual.

■ 4.5 Serial CEO Problem: Quadratic-Gaussian Case

In this section we present results for the serial CEO problem in the quadratic-Gaussian case. We present iterative distortion-rate expressions as well as discussing the design implications of these results. The formal derivations appear in Appendix D.1. We concentrate on the Gaussian case with a mean-squared distortion measure because this is a case of common interest and the results have a particularly simple expression. Before proceeding with the discussion we pause to momentarily consider how the binary-

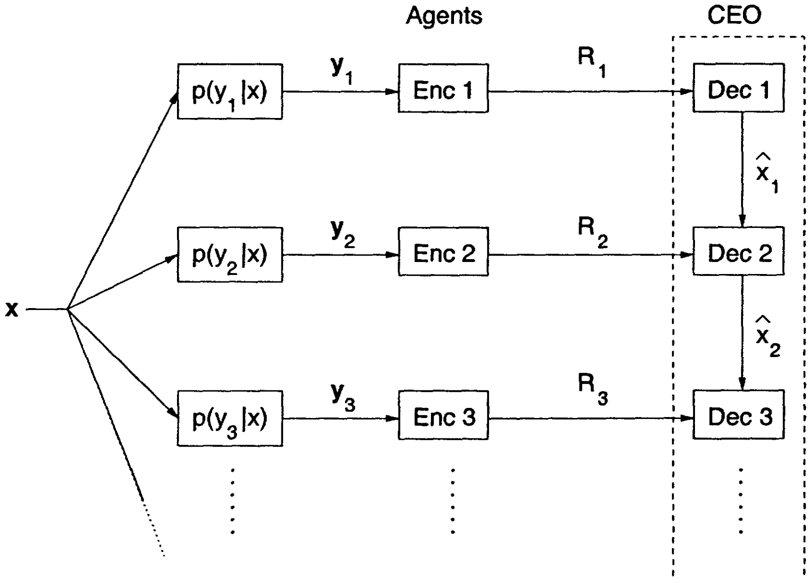


Figure 4.7. The encoding and successive decoding algorithms for the parallel CEO problem can be visualized as a succession of L noisy Wyner-Ziv stages. At stage l, y_l is the encoder observation, and $\hat{x}_{l-1}$ is the side information. In this diagram we have not indicated the CEO's direct observation $\mathbf{y}_{\mathrm{CEO}}$.

Hamming case would differ.

Example: Binary-Hamming case. In Section 3.1.1 we discussed the noisy Wyner-Ziv rate-distortion function for the binary-Hamming case. A prominent point of that discussion was that the data fusion function $f : \mathcal{Y}_0 \times \mathcal{U} \to \hat{\mathcal{X}}$ is particularly simple in the binary-Hamming case. In particular it selects either the quantized encoder observation $\mathbf{u}(s)$ or the side information $\mathbf{y}_0$ as the source estimate. If we extend this thinking to a chain of noisy Wyner-Ziv steps, as we propose for the serial CEO problem, this means that at each step each agent picks either a noisy version of the last agent's estimate or his own side information as the best source estimate. This means that no data fusion occurs at each step. In effect, the binary-Hamming case reduces to a voting problem and, since there are only two, we trust the more reliable voter. Hence, if successive codes are used, for the binary-Hamming case the problem is not particularly interesting. In the parallel problem discussed in Section 4.6 because we have more than two source of information, the problem becomes more interesting. □

■ 4.5.1 Iterative Distortion-Rate Expression

In this section we apply noisy Wyner-Ziv ideas to the quadratic-Gaussian serial CEO problem. In this problem the L agents and the CEO are linked together via rate-constrained channels of rates $R_1, R_2, \ldots, R_L$. Agent l receives message m_{l-1} from agent $l-1$ at rate R_{l-1}, and observes $\mathbf{y}_l = \mathbf{x} + \mathbf{v}_l$ where $\mathbf{x} \sim \mathcal{N}(0, \sigma_x^2 \mathbf{I})$ and $\mathbf{v}_l \sim \mathcal{N}(0, N_l \mathbf{I})$ are independent. The CEO receives message m_L from agent L at rate R_L and observes

$\mathbf{y}_{\mathrm{CEO}} = \mathbf{x} + \mathbf{v}_{\mathrm{CEO}}$ where $\mathbf{v}_{\mathrm{CEO}} \sim \mathcal{N}(0, N_{\mathrm{CEO}}\mathbf{I})$ is independent of $\mathbf{x}$ and the other noise sources. Each agent acts in turn, sending on the information that will most help the next agent in his estimation of $\mathbf{x}$. An achievable rate-distortion region for this problem can be derived using the results on noisy Wyner-Ziv coding from Chap. 3. The derivation is given in Appendix D.1 and yields a distortion expression that is iterative in l,

$$d_l = \frac{N_l\, d_{l-1}}{N_l + d_{l-1}} + \sigma^2_{\mathsf{x}|\mathsf{y}_l} \frac{\left(1 - \frac{d_{l-1}}{\sigma_{\mathsf{x}}^2}\right)}{\left(1 + \frac{d_{l-1}}{N_l}\right)} 2^{-2R_{l-1}}. \tag{4.7}$$

The distortion achieved at stage l is a function of four factors: 1) The source variance σ_{x}^2. 2) The distortion d_{l-1} achieved at stage $l-1$. 3) The quality of the current observation in terms of the observation noise variance N_l. And 4) the rate of communication R_{l-1} between agent $l-1$ and agent l.

To help understand (4.7) consider the limiting case of very large rate $R_{l-1} \to \infty$. In this case the second term approaches zero. The first term can be used to generate the lower bound $\sigma^2_{\mathsf{x}|\mathsf{y}_1,\ldots,\mathsf{y}_l}$ on d_l in an iterative manner. This can be understood as follows. If the agents are given infinite rate, they can simply forward their observations to the CEO at full resolution. Each agent in the chain could then use all the observations up to that point to make the minimum mean-squared estimate $E[\mathbf{x}|\mathbf{y}_1, \ldots, \mathbf{y}_l]$, resulting in estimation error $\sigma^2_{\mathsf{x}|\mathsf{y}_1,\ldots,\mathsf{y}_l}$. Generally, however, $R < \infty$, resulting in a non-zero second term which acts as a drag on the decay profile of d_l, slowing the decrease of d_l with l.

Example: Constant rate links with equal-variance observation noises. To better illustrate and analyze the effect of the pipeline of agents, consider the following scenario: $R = R_1 = R_2 = \ldots = R_L$, and $\mathbf{v}_1 \sim \mathbf{v}_2 \sim \mathbf{v}_3 \sim \ldots \sim \mathbf{v}_{\mathrm{CEO}} \sim \mathcal{N}(0, N\mathbf{I})$. In this case the data pipeline never decreases in capacity, but it does saturate. Under these conditions (4.7) simplifies to

$$d_l = \frac{N\, d_{l-1}}{N + d_{l-1}} + \sigma^2_{\mathsf{x}|\mathsf{y}} \frac{\left(1 - \frac{d_{l-1}}{\sigma_{\mathsf{x}}^2}\right)}{\left(1 + \frac{d_{l-1}}{N}\right)} 2^{-2R}. \tag{4.8}$$

The distortion described by (4.8) decreases monotonically with l. If the number L of agents is unbounded we can find $d_\infty = \lim_{l\to\infty} d_l$ by setting $d_l = d_{l-1} = d_\infty$ in (4.8). This yields $d_\infty \sim \Theta(2^{-R})$; specifically,

$$d_\infty = \frac{N}{2\left(1 + \frac{\sigma_{\mathsf{x}}^2}{N}\right)} \left[\sqrt{4\left(1 + \frac{\sigma_{\mathsf{x}}^2}{N}\right)\frac{\sigma_{\mathsf{x}}^2}{N} + 2^{-2R}} - 2^{-R}\right] 2^{-R}. \tag{4.9}$$

The distortion described by (4.8) decreases monotonically with l. In Fig. (4.8) we plot the decrease in mean-squared estimation error versus agent number for $\sigma_{\mathsf{x}}^2 = 4$, $N = 1$. Agent 1 has only its own observation, so its error is $\sigma^2_{\mathsf{x}|\mathsf{y}}$. Agent 2 has its own observation plus whatever information it gets at rate R from agent 1, etc. We plot the results for $R = 1, 2, 3, \infty$. The data points that correspond to each agent's estimation

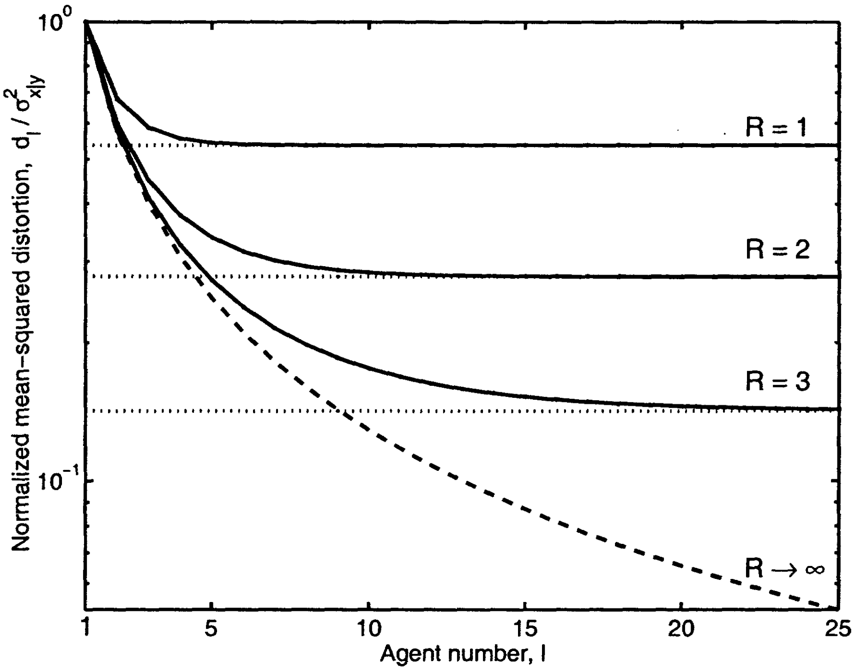


Figure 4.8. Serial CEO estimation error d_l improves with agent number. In this example $\sigma_x^2 = 4$, $N = 1$, and $R = 1, 2, 3, \infty$. Solid curves plot the evolution of the estimation error d_l for the different finite rates. Dotted lines plot the lower bound on performance for each finite rate, approached asymptotically as the number of agents l tends toward infinity. The dashed curve is $\sigma^2_{x|y_1,\ldots,y_l}$ the minimum mean-squared error in x given $y_1, \ldots, y_l$ which can be approached as the per-agent rate R grows toward infinity.

error are connected by solid lines for convenience. The limit for each R, given by (4.9), is plotted as a dashed line. □

■ 4.5.2 Design Implications

Consider what happens if the first agent in the chain has a noise-free observation ($\mathbf{y}_1 = \mathbf{x}$) while all the other agents measure the source in noise. Assuming equal rates per agent, the best strategy is for each agent to forward the first agents message, as no agent in the chain has a better estimate than the first. In the end only the CEO will need to fuse the message with his own observation. If other agents further up the chain go through the data fusion steps proposed in the inter-layer approaches described in Section 4.4, they will introduce additional quantization error, degrading system performance. In general, when deciding what to do agent l must take into consideration a number of factors that encapsulate the measurement quality and resource constraints elsewhere in the system: d_{l-1}, R_{l-1}, N_l, R_l, and N_{l+1}. Fortunately, this is all rather local information, enabled with only a modicum of two-way communication between agents.

The design implication of this discussion is that agent order matters. In particular, it is better to have low SNR, low rate agents earlier in the chain so that, at a minimum, an agent in the middle of the chain can forward on the message he receives unmodified. In addition, having the higher SNR agents come later means that they have better side information to use in message decoding. If at some point in the chain the communication rate drops, information may need to be erased. The successive coding approach encapsulated by the iterative distortion expression (4.7) easily deals with this possibility through the compound transcoding operation consisting of: decoding, data fusion, and noisy Wyner-Ziv coding.

In Section 4.5 we presented a constant rate link, equal-variance measurement noise example. Given this scenario, system designers can use d_∞ the asymptotically achieved distortion (4.9) in two ways. First, it provides a means for determining the bit pipe size required to achieve a target steady state distortion. Second, for a given constant bit pipe size, comparing d_∞ from (4.9) with the iterative expression (4.8) tells us when fusing in new data results in only a marginal decrease in distortion. For instance, in Fig. 4.8 for R = 2, after about 10 or 11 agents, the distortion decrease at each step becomes negligible. Stopping the data fusion process at that stage and simply forwarding on the message thereafter could save a lot of processing power.

■ 4.5.3 Growing the Bit Pipe Logarithmically

If the designer has some flexibility in allocating rate resources between agents, a constant rate bit pipe design is not the best choice. We would like to determine some design rules to determine how we should increase the bit pipe size to accommodate this increasing river of information. In this section we determine the rate of bit pipe growth needed to stay within a constant multiple $\kappa \geq 1$ of the lower-bound $\sigma^2_{\mathsf{x}|y_1,\ldots,y_l} = \frac{\sigma_x^2 N}{l\sigma_x^2+N} = \frac{\text{SNR}}{1+l\,\text{SNR}}$ where $\text{SNR} = \frac{\sigma_x^2}{N}$ is an individual agent's signal-to-noise ratio.

An upper bound on the transmission rate needed at each stage (i.e., $R_1 < R_2 < \cdots$) can be found by setting $d_l = \kappa\sigma^2_{\mathsf{x}|y_1,\ldots,y_l}$ and using the iterative distortion-rate expression (4.7) to solve for the rate R_l such that $d_{l+1} = \kappa\sigma^2_{\mathsf{x}|y_1,\ldots,y_{l+1}}$. This gives us

$$\begin{aligned} R_l &\leq \frac{1}{2}\log\left[\frac{[(l+1)\text{SNR}+1][l\text{SNR}-\kappa+1]}{\text{SNR}(1+\text{SNR})\kappa(\kappa-1)}\right] && (4.10)\\ &\leq \frac{1}{2}\log\left[\frac{\text{SNR}}{1+\text{SNR}}\right]+\log\left[\frac{l}{\kappa-1}\right]+O(1). && (4.11)\end{aligned}$$

We can repeat this analysis for the estimate-and-quantize approach of Section 4.3.2, giving

$$R_{\text{EQ},l} \leq \frac{1}{2}\log\left[\frac{[(l+1)\text{SNR}+1-\kappa\text{SNR}][l\text{SNR}-\kappa+1]}{\text{SNR}\kappa(\kappa-1)}\right]. \tag{4.12}$$

Subtracting (4.10) from (4.12) tells us how much rate the inter-layer approach saves:

$$R_{\text{EQ},l} - R_l = \frac{1}{2}\log\left[\left(1-\frac{\kappa\,\text{SNR}}{(l+1)\text{SNR}+1}\right)(1+\text{SNR})\right]. \tag{4.13}$$

The rate savings (4.13) decreases with increasing κ. This is intuitively correct since a larger κ means that the target distortion is larger, so more easily met, and using a more efficient scheme such as the successive approaches, is less important. Independent of κ, however, as the estimate progresses down the chain of agents (l increases), the rate savings converges to a constant $\frac{1}{2}\log[1 + \text{SNR}]$ which is intriguingly familiar.

4.6 Parallel CEO Problem: Quadratic-Gaussian Case

In this section we present results for the parallel CEO problem in the quadratic-Gaussian case. As for the serial problem, we present iterative distortion-rate expressions and discuss the design implications of these results. In addition, we connect our results to earlier results on CEO problems. The formal derivations appear in Appendix D.2. Before proceeding with the discussion we build on the discussion of the binary-Hamming case for the serial problem to understand how the problem differs in the parallel case.

Example: Binary-Hamming case. In Section 4.5 we discussed how the binary-Hamming case is not particularly interesting for the serial problem because we must make hard decision at each step. The same discussion would hold for the parallel problem if we used unmodified the refined approach of Section 4.4 that forces a data fusion step to be performed after each message is decoded. Because in the parallel problem once a message is decoded we don't have to re encode it to send on the relevant information to the CEO – all information is decoded at the CEO – we can use a slightly different strategy. The selection from the transmitted bin of the correct codeword is done as before, using the noisy Wyner-Ziv results. We slightly modify the data fusion step. In Wyner-Ziv problem , as discussed in Section 3.1.1, all the decoder can do is to pick the most reliable piece of information (side information or vector-quantized source observation). In the parallel CEO problem after l agents have reported in there are $l + 1$ pieces of information (one from each agent and one from the CEO's observation). Since the CEO generally has more than three pieces of information, a more complicated voting process can be carried out to improve the estimate as increasing number of agents report in. □

4.6.1 Iterative Distortion-Rate Expression

The derivation of the achievable rate-distortion region for the parallel CEO problem using successive codes is similar to that for the serial CEO problem, see Appendix D.2. In the quadratic-Gaussian problem the source $\mathbf{x}$ is a n-length i.i.d. zero-mean Gaussian sequence of variance σ_x^2, the noises $\mathbf{v}_l \sim \mathcal{N}(0, N_l\mathbf{I})$ are independent of each other and of the source, and agent l communicates to the CEO at rate R_l. We define d_l to be the CEO's observation after the first l agents have reported in. Let $d_0 = \sigma^2_{x|y_{\text{CEO}}}$. Then, in general,

$$d_l = \frac{N_l\, d_{l-1}}{N_l + d_{l-1}} + \frac{d_{l-1}^2}{N_l + d_{l-1}} 2^{-2R_l}. \tag{4.14}$$

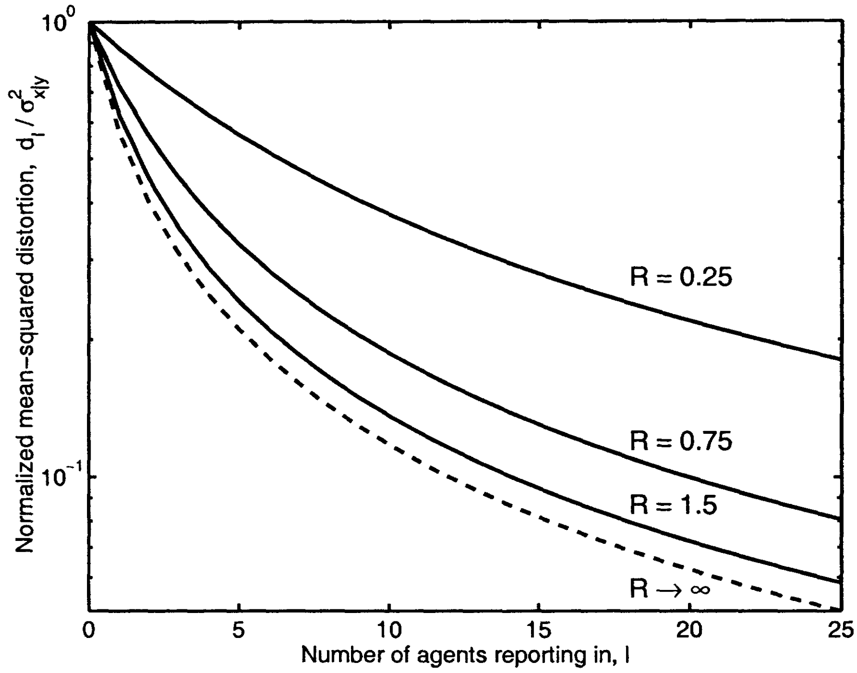


Figure 4.9. Parallel CEO estimation performance, d_l, improves with agent number. In this example $\sigma_x^2 = 4$, $N = 1$, and $R = 0.25, 0.75, 1.5, \infty$. Solid curves plot the evolution of the estimation error d_l for the different finite rates. The dashed curve plots $\sigma^2_{x|y_{CEO},y_1,\ldots,y_l}$, approached as R grows to infinity.

The distortion d_l achieved at stage l is a function of three variables: d_{l-1}, N_l and R_l. Comparing (4.14) to the iterative expression for the serial problem (4.7), we see that the first terms are identical, so the same discussion about optimality as R get very large holds, as well as thinking of the second (positive) term as a drag on distortion decay.

Example: Constant rate links with equal-variance observation noises. In order to compare our results to the serial results of Section 4.5 consider again the case where the agents have equal SNR $= \sigma_x^2/N$, i.e., $\mathbf{v}_1 \sim \mathbf{v}_2 \sim \ldots \sim \mathbf{v}_L \sim \mathbf{v}_{CEO} \sim \mathcal{N}(0, N\mathbf{I})$. In Fig. 4.9 we plot the decrease in mean-squared estimation error versus agent number for $\sigma_x^2 = 4$, $N = 1$. We plot the results for $R = 0.25, 0.75, 1.5, \infty$. The data points that correspond to each agent's estimation error are connected by solid lines for convenience. The lower bound, $\sigma^2_{x|y_{CEO},y_1,\ldots,y_l}$, is plotted as a dashed curve. In contrast to the same example developed for the serial problem in Fig. 4.8, now there is no saturation effect. Saturation does not occur because, unlike in the serial problem, all agents communicate directly to the CEO at a fixed rate. Therefore, no agent acts as a communication bottleneck and CEO accumulates infinite data as the number of agents reporting in grows to infinity. □

Example: Increasing R_l so that $d_l \leq \kappa \sigma^2_{x|y_{CEO},y_1,\ldots,y_l}$. Just as in the serial problem, we can again ask at what rate must each agent transmit so that the CEO's distortion d_l

stays within a factor κ of the lower bound $\sigma^2_{x|y_{\mathrm{CEO}},y_1,\ldots,y_l}$. By setting $d_l = \kappa\sigma^2_{x|y_{\mathrm{CEO}},y_1,\ldots,y_l}$ and $d_{l+1} = \kappa\sigma^2_{x|y_{\mathrm{CEO}},y_1,\ldots,y_{l+1}}$ we can use the iterative distortion-rate expression (4.14) to solve for R_l in the equal-SNR situation where $N = N_{\mathrm{CEO}} = N_1 = \ldots = N_{l+1}$. We find that an upper bound on R_l which is achieved using the successive coding techniques of this chapter is

$$R_l < \frac{1}{2}\log\left[\frac{(l+1)\mathrm{SNR}+1}{l\mathrm{SNR}+1}\,\frac{\kappa}{\kappa-1}\right]. \tag{4.15}$$

Similar to the serial result (4.10), as κ approaches zero R_l approaches infinity. There are two major differences between (4.15) and (4.10), however. First, R_l *decreases* in l. Since the message from each agent does not have to encapsulate all previous agents' data it should certainly be far smaller than that for the serial problem. Furthermore, as the CEO accumulates increasing amounts of data, he has better side information to use in the decoding process. Therefore, to contribute equally to the CEO's estimate, later agents can send at lower rates. Second, the rate at which later agents send converges to $\lim_{l\to\infty} R_l = \frac{1}{2}\log\left[\frac{\kappa}{\kappa-1}\right]$. □

■ 4.6.2 Design Implications

Rewriting (4.14) as (4.16) helps us to understand how d_l evolves in l.

$$\frac{d_l}{N_l} = \frac{d_{l-1}}{N_l}2^{-2R_l} + \left(1 - 2^{-2R_l}\right) - \frac{\left(1-2^{-2R_l}\right)}{1+\frac{d_{l-1}}{N_l}}. \tag{4.16}$$

In particular, the normalized mapping from $\frac{d_{l-1}}{N_l}$ to $\frac{d_l}{N_l}$ is hyperbolic, increases monotonically from the origin where it has unit-slope, and stays below the 45-degree line $d_l = d_{l-1}$ for all R_l. This mapping is plotted in Fig. 4.10 for various R_l. Moreover, for either $\frac{d_{l-1}}{N_l}$ large or R_l small, the dynamics are effectively linear, since the third term in (4.16) is small. From the hyperbolic dynamics (4.16) we can verify that if the noise variance N_l is bounded for all l, then for any fixed per-agent rate $R_l = R$, in the limit as the number of agents reporting in grows to infinity, d_l converges to zero. However, d_l approaches zero asymptotically slowly because the mapping from d_{l-1} to d_l approaches an equality as d_{l-1} nears zero.

Say we are given a set of agents with noise levels $N_1,\ldots,N_L$ and communication rates $R_1,\ldots,R_L$. One problem we are interested in is how to find the best ordering of agent transmissions. To determine the best ordering rewrite (4.16) as (4.17).

$$d_l = \left(\frac{1+\frac{d_{l-1}}{N_l}2^{-2R_l}}{1+\frac{d_{l-1}}{N_l}}\right)d_{l-1} \simeq f(d_{l-1},N_l,R_l)\,d_{l-1}, \tag{4.17}$$

Given two agents with different noise levels – N_a and N_b – and two different communication rates – R_a and R_b – we can calculate the distortion both orderings achieve

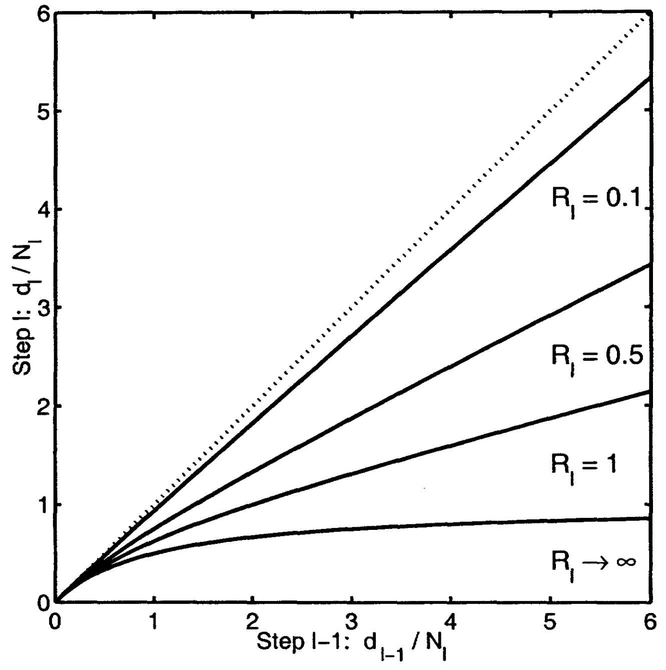


Figure 4.10. Normalized distortion dynamics as a function of R_l.

starting from some a priori distortion d:

$$d_{ab} = f[f(d, N_a, R_a)\, d, N_b, R_b]\, f(d, N_a, R_a)\, d, \tag{4.18}$$

$$d_{ba} = f[f(d, N_b, R_b)\, d, N_a, R_a]\, f(d, N_b, R_b)\, d. \tag{4.19}$$

If $d_{ab} < d_{ba}$ then it is best for agent a to transmit first and b second. This logic extends to higher numbers of agents because the best ordering is independent of starting distortion d. Therefore, we can use a type of `sort` algorithm where each pair-wise sorting decision is made based on (4.18) and (4.19). The `sort` algorithm is order $L \log(L)$. This comparison can be used to show that if $R_a = R_b$ it is best for the low SNR agent to transmit first. As will be shown in the next section, if $N_a = N_b$ it is best for the high-rate agent to transmit first.

■ 4.6.3 Special Cases and Connections to Earlier CEO Results

In this section we discuss some special case of the parallel CEO problem. We focus on the quadratic-Gaussian case with equal agent SNRs, no observation at the CEO, and a sum-rate constraint. This is the case discussed most extensively in [75, 45] and allows us to connect some of our results to the results of these earlier papers.

Two Agents, Fixed Sum-Rate

We first consider the situation where there are only two agents $L = 2$, each has equal-SNR observations, and the CEO has no observation $\mathbf{y}_{\text{CEO}} = 0$. We optimize the parallel CEO iterative distortion expression (4.14) over a sum-rate constraint $\bar{R}$ to minimize the achievable distortion using our successive coding strategy. We then use Oohama's converse [45] to show that this approach achieves the distortion-rate bound for this scenario.

Given the total communication rate $\bar{R} = R_1 + R_2$ we optimize over the fraction of total rate allocated to each agent: $R_1 = \lambda\bar{R}$ and $R_2 = (1-\lambda)\bar{R} = \tilde{\lambda}\bar{R}$. Somewhat surprisingly we will find that in general $R_1 \neq R_2$. To simplify the calculus, in this section all rates are expressed in nats.

$$
\begin{aligned}
d_1 &= \sigma_x^2 \left(\frac{1 + \frac{\sigma_x^2}{N} e^{-2\lambda\bar{R}}}{1 + \frac{\sigma_x^2}{N}} \right) && (4.20)\\
d_2 &= d_1 \left(\frac{1 + \frac{d_1}{N} e^{-2\lambda\bar{R}}}{1 + \frac{d_1}{N}} \right)\\
&= \frac{\sigma_x^2 \left(1 + \frac{\sigma_x^2}{N} e^{-2\lambda\bar{R}}\right)}{\left(1 + \frac{\sigma_x^2}{N}\right)} \left[\frac{\left(1 + \frac{\sigma_x^2}{N}\right) + \frac{\sigma_x^2}{N}\left(e^{-2\tilde{\lambda}\bar{R}} + \frac{\sigma_x^2}{N} e^{-2\bar{R}}\right)}{\left(1 + \frac{\sigma_x^2}{N}\right) + \frac{\sigma_x^2}{N}\left(1 + \sigma_x^2 e^{-2\lambda\bar{R}}\right)} \right]. && (4.21)
\end{aligned}
$$

Equation (4.20) follows from the iterative distortion expression (4.14) where $d_0 = \sigma_x^2$ since we set $\mathbf{y}_{\text{CE}} = 0$ and hence $\sigma^2_{x|y_{\text{CEM}}} = \sigma_x^2$. Defining $\text{SNR} = \frac{\sigma_x^2}{N}$, taking the derivative of (4.21) with respect to λ, and setting the result equal to 0 gives

$$
0 = \frac{d}{d\lambda} d_2 = \left[e^{-2\bar{R}}(1 + 2\,\text{SNR})\right] e^{4\lambda\bar{R}} + \left[2\,\text{SNR}^2 e^{-2\bar{R}}\right] e^{2\lambda\bar{R}} - \left[\text{SNR}^2 e^{-2\bar{R}} + (1 + \text{SNR})^2\right]. \tag{4.22}
$$

Using the quadratic equation, we can solve for $e^{2\lambda\bar{R}}$, and invert the exponent to find λ.

$$
\lambda = \frac{1}{2\bar{R}} \log_e \left[\frac{-\text{SNR}^2 + (1 + \text{SNR})\sqrt{\text{SNR}^2 + (1 + 2\,\text{SNR})e^{2\bar{R}}}}{(1 + 2\,\text{SNR})} \right]. \tag{4.23}
$$

Finally, substituting (4.23) into (4.21) to solve for the distortion results in

$$
d_2 = \sigma_x^2 \left\{ \frac{(\gamma + \text{SNR})\left[1 + 2\,\text{SNR} + \text{SNR}\,(\gamma + \text{SNR})\, e^{-2\bar{R}}\right]}{(1 + 2\,\text{SNR})^2 \gamma} \right\}, \tag{4.24}
$$

where $\gamma = \sqrt{\text{SNR}^2 + (1 + 2\,\text{SNR})e^{2\bar{R}}}$. To simplify (4.24) note that $\gamma^2 - \text{SNR}^2 = (1 +$

$2\mathrm{SNR})e^{2\bar{R}}$. To use this identity multiply (4.24) by $(\gamma - \mathrm{SNR})^2/(\gamma - \mathrm{SNR})^2$ to get

$$\begin{aligned}
d_2 &= \sigma_x^2 \left\{ \frac{(\gamma^2 - \mathrm{SNR}^2)\left[(1 + 2\,\mathrm{SNR})(\gamma - \mathrm{SNR}) + \mathrm{SNR}(\gamma^2 - \mathrm{SNR}^2)2^{-2\bar{R}}\right]}{(\gamma - \mathrm{SNR})^2(1 + 2\,\mathrm{SNR})^2\gamma} \right\} \\
&= \sigma_x^2 \left\{ \frac{(1 + 2\mathrm{SNR})e^{2\bar{R}}\left[(1 + 2\mathrm{SNR})(\gamma - \mathrm{SNR}) + \mathrm{SNR}(1 + 2\mathrm{SNR})e^{2\bar{R}}e^{-2\bar{R}}\right]}{(\gamma - \mathrm{SNR})^2(1 + 2\,\mathrm{SNR})^2\gamma} \right\} \\
&= \sigma_x^2 \left\{ \frac{e^{2\bar{R}}(\gamma - \mathrm{SNR} + \mathrm{SNR})}{(\gamma - \mathrm{SNR})^2\gamma} \right\} \\
&= \frac{\sigma_x^2 e^{2\bar{R}}}{(\gamma - \mathrm{SNR})^2}. \qquad (4.25)
\end{aligned}$$

We now show that the distortion achieved (4.25) meets a lower bound on the rate-distortion region for this problem derived by Oohama in [45]. This will demonstrate that the (4.25) is the distortion-rate function for this problem. In order to state the results of [45] define

$$D^*(s, L) \equiv \frac{\sigma_x^2}{L\,\mathrm{SNR}\left(1 - e^{-2s/L}\right) + 1}, \qquad (4.26)$$

where L is the number of agents, and $s \geq 0$ is a helper parameter. The achievable rate-distortion region for equal-SNR agents, a sum-rate constraint, and no CEO observation is shown to be a subset of the following region R^*:

$$\begin{aligned}
R^* = \{(\bar{R}, d) : \bar{R} &\geq s + \frac{1}{2}\log\left[\frac{\sigma_x^2}{d^*(s, L)}\right], \qquad (4.27) \\
d &\geq D^*(s, L)\}. \qquad (4.28)
\end{aligned}$$

For the case $L = 2$ we substitute (4.26) into (4.27) with $L = 2$ to get

$$\begin{aligned}
\bar{R} &\geq s + \frac{1}{2}\log\left[1 + 2\mathrm{SNR} - 2\mathrm{SNR}e^{-s}\right] \\
2\left(\bar{R} - s\right) &\geq \log\left[1 + 2\mathrm{SNR} - 2\mathrm{SNR}e^{-s}\right] \\
0 &\leq \left(e^{2\bar{R}}\right)e^{-2s} + (2\mathrm{SNR})e^{-s} - (1 + 2\mathrm{SNR}), \qquad (4.29)
\end{aligned}$$

where (4.29) follows from exponentiating both sides and rearranging terms. Using the quadratic equation we can solve for e^{-s},

$$e^{-s} = \frac{\gamma - \mathrm{SNR}}{e^{2\bar{R}}}, \qquad (4.30)$$

where as before $\gamma \equiv \sqrt{\mathrm{SNR}^2 + (1 + 2\,\mathrm{SNR})e^{2\bar{R}}}$. Substituting (4.30) into (4.26) with $L = 2$ and using the result in (4.28) yields

$$d \geq \frac{\sigma_x^2 e^{2\bar{R}}}{(\gamma - \mathrm{SNR})^2}, \qquad (4.31)$$

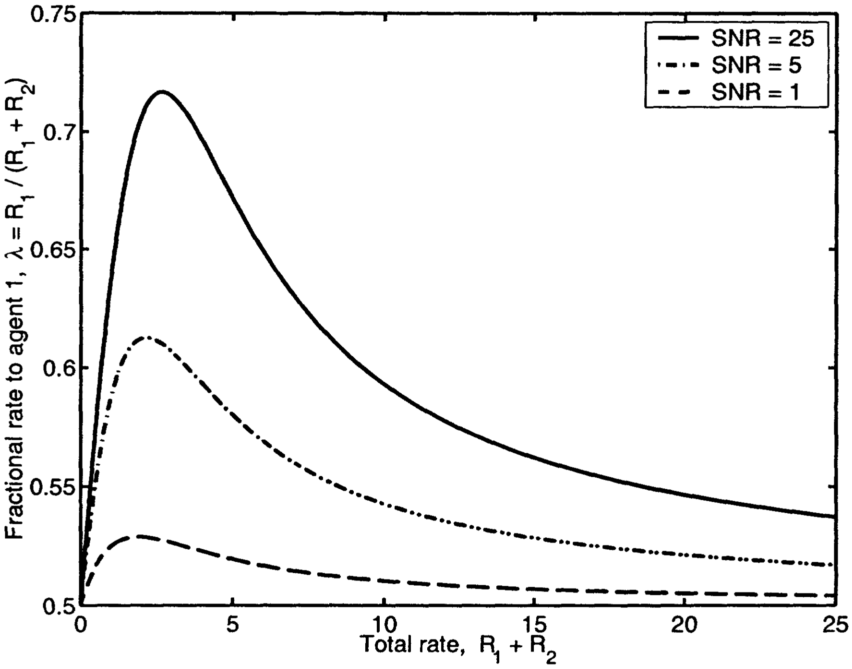


Figure 4.11. The fraction of the total rate $R_{\text{tot}} = \bar{R}$ assigned to each agent is a function of both R_{tot} and SNR.

which equals d_2, the achievable distortion given by (4.25). Hence, for the $L = 2$ case successively structured codes can achieve the rate-distortion bound for the problem. In Fig. 4.11 we plot λ as a function of $\bar{R}$, parameterized by SNR.

The asymmetric rate allocations indicated by Fig. 4.11 mean that the rate-distortion region has a flat frontier in at least one section. This is because in the two-agent case we minimize the distortion subject to a sum rate constraint. If $(R_1, R_2) = (\lambda\bar{R}, [1-\lambda]\bar{R})$ is on the boundary, then so is $(R_1, R_2) = ([1-\lambda]\bar{R}, \lambda\bar{R})$, by symmetry since the agents have identical SNRs. The line that connects these two points can be achieved by time-sharing, and must also lie on the rate-distortion bound since it has the same sum-rate as its two end points.

Another point to notice about the fractional rate allocation of (4.23), which is easier to see in Fig. 4.11, is that λ is roughly 1/2 if the sum-rate $\bar{R} = R_1 + R_2$ is either very large or very small. The latter suggests that perhaps even if $L > 2$, as the average rate per agent gets small (or if L is very large for fixed sum-rate $\bar{R}$), constraining agents to equal rates may not incur a large distortion penalty as compared to the minimum achievable distortion. We discuss this idea further in the next two section. Finally, note that we can generalize the optimization problem of this section to agents with differing SNRs, but the resultant expressions are far more complex, and in certain cases one of the agents will receive the full rate allocation, i.e., $R_1 = \bar{R}$ or $R_2 = \bar{R}$.

Intermediate Numbers of Agents, Fixed Sum-Rate

In the last section we saw that in the two agent case successively structured codes can reach the rate-distortion bound. When there are more than two agents it is more difficult to optimize the agent rate allocations subject to a sum-rate constraint. In this subsection we instead divide the total rate equally between agents and discuss how much large the resultant distortion is than is the lower bound on the rate-distortion function presented in (4.27) and (4.28) that were derived by Oohama [45]. In more recent work [46], Oohama has shown that this bound is in fact the rate-distortion function. In Fig. 4.12 we plot the percentage of extra distortion incurred by using the successive coding approach to the parallel CEO problem under the equal-rate allocations $R = R_1 = R_2 = \ldots = \bar{R}/L$. If we term the distortion achieved by the successive coding method d_{succ} and the distortion bound from (4.27) and (4.28) d_{bnd}, the the percentage penalty is calculated as

$$\text{Percent Penalty} = 100\,\frac{d_{\text{succ}} - d_{\text{bnd}}}{d_{\text{bnd}}}.$$

For the examples shown $\bar{R} = 10$. We consider three cases, SNR $= 2.5, 5, 10$. In each case all agents had the same observation SNR. For all SNRs the distortion penalty incurred is only a few percent and decreases with decreasing SNR. In all cases the penalty approaches zero as the number of agents L grows to infinity. We next discuss this asymptotic regime, which is the one considered in [75, 45].

Large Numbers of Agents, Fixed Sum-Rate

In [45, 75] the authors investigated CEO estimation performance given a total rate constraint $\bar{R}$ as the number of agents L grows to infinity. In other words, as the average per-agent rate $R = \bar{R}/L$ approaches 0. In Fig. 4.11 we observed that in the $L = 2$ case, as $\bar{R}/L$ approaches zero, equal rate allocation per agent yields the rate-distortion optimal solution. In Fig. 4.12 we observe the same type of behavior in the intermediate L case as $\bar{R}/L$ approaches zero.

We now consider the asymptotic version of these scenarios as L grows to infinity. In this situation the average rate per agent is also going to zero. Motivated by the observations of the last paragraph we analyze the per-agent equal rate solution, and show that as the number of agents grows toward infinity under a sum-rate constraint $\bar{R}$, a sequence of successively structured coding design with agent communication rates constrained to be identical $(R_1 = R_2 = \ldots = R_L = \bar{R}/L)$ converges to a rate-distortion optimal solution. The derivation we present here gives the basic ideas of the proof with a more formal derivation given in Appendix D.3. For this discussion all rates will be in nats.

Under the equal rate simplification $R_l = \frac{\bar{R}}{L} \equiv R$, and assuming equal agent SNRs,

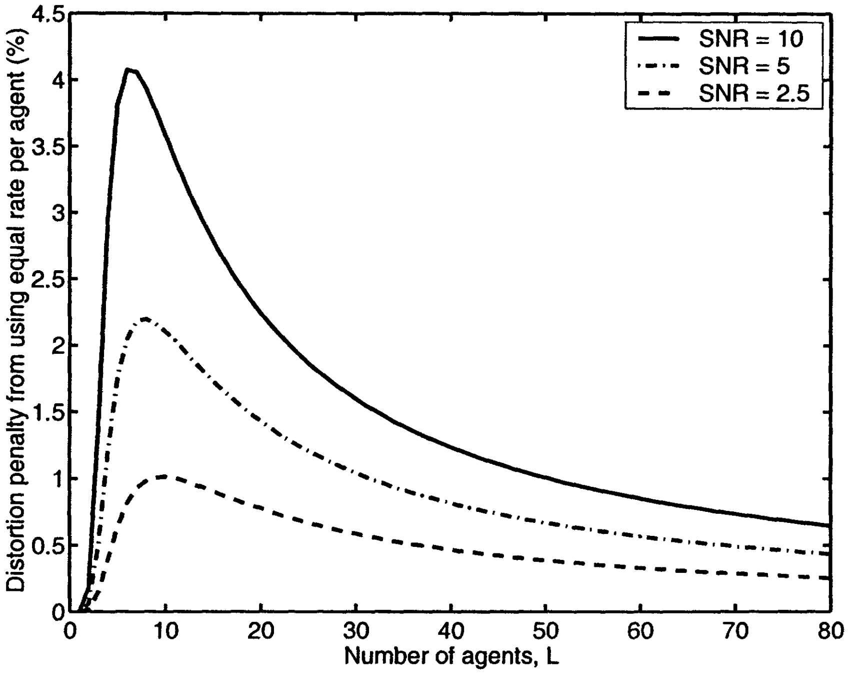


Figure 4.12. The percentage of extra distortion incurred by constraining the agents to equal rates. For the three examples presented $\bar{R} = 10$, and the SNRs of the L agents were identical within each example.

we rewrite the iterative distortion expression for the parallel CEO problem (4.14) as

$$\frac{d_l}{N} = \frac{d_{l-1}}{N}\left(\frac{1+\frac{d_{l-1}}{N}2^{-2R}}{1+\frac{d_{l-1}}{N}}\right), \tag{4.33}$$

Defining $x_l = d_l/N$ and subtracting x_l from both sides of (4.33) gives

$$\begin{aligned} x_l - x_{l-1} &= x_{l-1}\left(\frac{1+x_{l-1}e^{-2R}}{1+x_{l-1}}\right) - x_l \\ x_l - x_{l-1} &= -\frac{x_{l-1}^2(1-e^{-2R})}{1+x_{l-1}} \\ \frac{x_l - x_{l-1}}{1-e^{-2\bar{R}/L}} &= -\frac{x_{l-1}^2}{1+x_{l-1}} \end{aligned} \tag{4.34}$$

Since L is large, we substitute the first two terms of the Taylor series expansion of $1 - e^{-2\bar{R}/L} \simeq 2\bar{R}/L$ into (4.34) to get

$$\frac{x_l - x_{l-1}}{\bar{R}/L} \simeq \frac{-2x_{l-1}^2}{1+x_{l-1}}. \tag{4.35}$$

Since $\lim_{L\to\infty} \bar{R}/L = 0$, (4.35) is a discrete approximation to a differential equation. For the moment we treat this approximation as exact in the limit $L \to \infty$. In Appendix D.3 we justify this approximation.

$$\begin{aligned} \frac{dx}{dR} &= \frac{-2x^2}{1+x} \\ \int_0^{\bar{R}} dR &= \int_{\frac{\sigma_X^2}{N}}^{\frac{d}{N}} \left(-\frac{1}{2x^2} - \frac{1}{2x}\right) dx && (4.36) \\ \bar{R} &= \frac{N}{2\sigma_X^2}\left[\frac{\sigma_X^2}{d} - 1\right] + \frac{1}{2}\log\frac{\sigma_X^2}{d}. && (4.37) \end{aligned}$$

In (4.36) we separate variables and integrate. The distortion d is the (unknown) final distortion achieved after the (known) total rate $\bar{R}$ is received by the CEO. Finally, (4.37) is the rate-distortion bound for this problem as derived in [45] where $d \le \sigma_X^2$.

■ 4.7 Application to the Relay Channel

The general relay channel [21] can be split into two halves: a broadcast side and a multiple-access side. Following work in [13] we point out that viewing the agents as relays, and the CEO as a decoder, means we can use our work on the CEO problem to derive an achievable rate region for the Gaussian relay channel [21] with L relays and a particular form of multiple-access channel. The multiple-access side consists of L fixed-rate non-interfering channels so, e.g., ideas of distributed coordination [60] cannot be implemented. For $L = 1$ we simply apply noisy Wyner-Ziv source coding with side information, and the scheme reduces to one discussed in [21]. Assume the transmitter and relay have powers P_{trans} and P_{relay}, respectively, and that the additive white Gaussian noises are of powers N_{relay} and N_{dec} at the relay and decoder, respectively. Then, a combination of stripping and noisy source coding with side information allows the following rate to be achieved:

$$R_{\text{relay}} = \frac{1}{2}\log\left[\frac{P_{\text{trans}} + P_{\text{relay}} + N_{\text{dec}}}{N_{\text{dec}}\left(1 + \frac{P_{\text{relay}}N_{\text{relay}}}{P_{\text{trans}}N_{\text{relay}} + P_{\text{trans}}N_{\text{dec}} + N_{\text{relay}}N_{\text{dec}}}\right)}\right]. \tag{4.38}$$

As the relay power grows ($P_{\text{relay}} \to \infty$) this strategy achieves capacity. But, as the channel to the relay becomes perfect, ($N_{\text{relay}} \to 0$) this approach suffers because cooperation is not exploited.

For higher numbers of relays we can apply the parallel CEO solution (with or without the CEO having a direct observation of the source). In Fig. 4.13 we show schematically how the solution to the CEO problem can be applied to relay channel communications. The idea parallels one developed in [13] for achieving capacity on Gaussian broadcast channels via information embedding (instead of superposition coding). The basic idea is to think of the codeword $\mathbf{x}$ as any old i.i.d. Gaussian source sequence, use the parallel

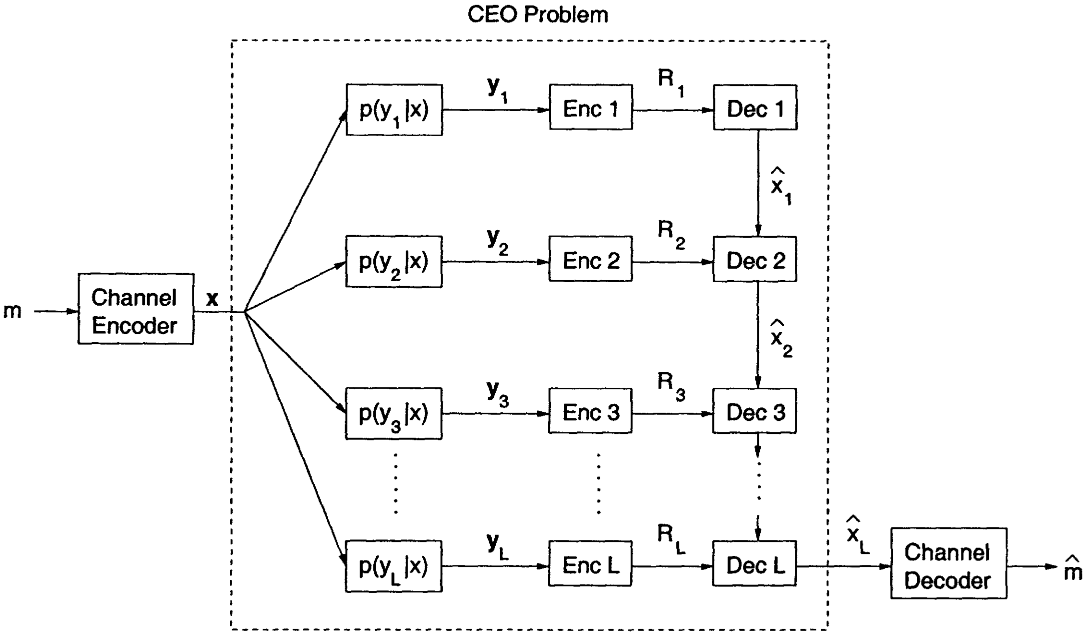


Figure 4.13. Certain classes of relay channel communication problems can be approached via the parallel CEO solution by wrapping a channel encoder and decoder around the distributed source estimator.

CEO solution to fuse the messages into a central estimate, $\hat{x}$, and then treat $\hat{x}$ as a source observation to decode. Since $\hat{x}$ and x are jointly typical, standard typicality decoding will work.

Related work in this area includes that of Schein [59, 60] and Gastpar and his colleagues [30]. In both works the assumption is made, as in [21], that the relays are fully synchronized and and are all received in the same band. If this is the case, then an optimal transmission scheme would take advantage of the correlation in the observations to make the codewords interfere constructively at the receiver, thereby getting an effective power boost. Unfortunately, figuring out how to get this constructive interference seems very difficult. Furthermore, in the frequent cases where the multiple access half of the relay channel consists of orthogonal links, the possibilities for coherent combination of channel codewords are eliminated. In such cases the CEO approach to the problem should do better in relation to capacity.

An interesting aspect of this approach to the relay channel is the central role that minimum mean-square error estimation plays; if estimation is left out of a noisy Wyner-Ziv stage, the effective channel noise will remain too high for the decoder to be able to determine the message reliably. This suggests that the CEO solution to the relay channel is particularly well suited to multilayered codes. If only a few agents are initially available to the CEO, the effective SNR at his receiver will be low. However, if more agents become available, they can be brought on-line without altering the functioning of the first set of agents. Implementationally, this Markov-like structure of the coding strategy can be quite useful.

It would be quite satisfying if we could show that some approach to the parallel CEO problem can be used as a capacity-achieving approach to a class of relay channels. This class would be relay channels where the multiple-access side consists of a set of rate-constrained, but orthogonal channels. This constraint on the multiple-access side means that constructive interference techniques cannot be used to boost the receiver power. Although the approach we present in this section gives an achievable rate region, we cannot apply the converse from the CEO problem to this relay channel problem. This is because the CEO problem converse hinges on the fact that $\mathbf{x}$ is an i.i.d. Gaussian sequences. If $\mathbf{x}$ is a codeword it need not be i.i.d. Gaussian. Of course, as we propose in this section, we can always use a randomly generated i.i.d. Gaussian codebook which looks marginally i.i.d., and the scheme discussed will work. However, to show this scheme achieves capacity, we must also show that the constraining of the codebook to be i.i.d. Gaussian does not reduce capacity. Hence, the approach proposed is an achievability result only, and converses from CEO problems are not necessarily the right tools for relay problem.

Fundamentally, the problem of channel coding over a relay channel is a distributed detection problem. The system goal is to take into account all the relay's data in order to determine the most likely codeword sent. On the other hand, in estimation problems the objective is to optimize a fidelity criterion. Generally, such fidelity criteria are very different from the probability-of-error criterion common to channel coding and detection problems. For this reason, while we can apply the coding strategies designed for the parallel CEO problem to the relay channel, it would be surprising if the optimal solution to a distributed detection problem could be separated into an optimal distributed estimation step followed by a centralized decision step.

■ 4.8 Chapter Summary

In this chapter we introduce a general model for a particular class of sensor network. These networks are characterized by: 1) the goal of getting a single node the best source estimate possible, 2) finite-rate inter-sensor communications, 3) tree-structured inter-node communications, 4) full statistical knowledge of the source and observations where the observations are conditionally independent given the source, 5) no delay constraints, and 6) no processing constraints. Using this model we can break a general sensor network into a set of prototype serial and parallel networks. We connect each of these prototype networks to the noisy Wyner-Ziv results of Chapter 3 and derive an achievable distortion-rate region for these basic networks and, by extension, any general tree network. We discuss a number of special cases and the design implication of our results. We connect our work to earlier information-theoretic work on the CEO problem and show how to use our results to generate earlier results. Finally, we show how the coding strategies proposed herein can be applied to certain classes of relay channels.

Chapter 5

Queuing with Distortion-Control

A primary task of communication networks is to move data around. In many networks data arrives at, and is transmissions from, each node in the network intermittently and unpredictable. This makes the buffers at each node susceptible to overflows, resulting in lost data, and decreased system performance. However, when the data being handled is distortion-tolerant, network protocols can be designed that exploit the distortion-tolerance of the data to reduce the probability of overflows.

In particular, we designed successively structured algorithms for content-aware networks, showing how to use data distortion-tolerance to produce robust, high-performance buffering protocols. These algorithms encode data using multiresolution source codes. The ordered information structure of this type of code is used to alleviate congestion in a controlled manner, trading off fidelity for additional memory resources to avoid uncontrolled buffer overflows. The result is an adaptive algorithm that minimizes end-to-end distortion robustly to uncertainty in arrival and departure statistics. Furthermore, the performance of the algorithm closely approximates a bound on the performance of any algorithm.

In Section 5.1 we introduce the problem of distortion-controlled queuing through a pair of illustrative applications. In Section 5.2 we discuss our approach to the problem at a high level, motivating a basic ad hoc approach that, as we show later, is nearly optimal. In Section 5.3 we describe the queuing theoretic model used for the problem and discuss multiresolution source codes. In Section 5.4 we present the design of the algorithms under consideration, and analyze their performances in Section 5.5. In Section 5.6 we contrast the performances of the algorithms, and compare the analysis with experimental results. Once we understand the performance of the algorithms we make some comments on the design implications of these ideas in Section 5.7, and conclude in Section 5.8.

■ 5.1 Introduction

As an example of an application of distortion-controlled queuing, consider a wired-to-wireless gateway manager routing multimedia content. If the wired half of the infrastructure is high capacity, any bottleneck in communications will likely occur at the gateway. The channel from gateway to receivers is often a shared medium, as in cellular systems. In some instances there may be only a single user requesting downloads of content, while at other times there may be multiple users requesting downloads. Because the number of users on the system is unpredictable, and the statistics of this process may be time-varying (e.g., typically heavier loads during the day than in the night), it is difficult to develop static protocols that deal equally well with all situations. Instead, a set of protocols that could adapt to the changing system load in real-time, without forward planning, would be ideal. To design such a set of protocols we can exploit more detailed knowledge of multimedia content characteristics. In particular, multimedia content is distortion-tolerant – it is useful at a range of levels of fidelity. This contrasts with, e.g., executables that are distortion-intolerant and must be communicated losslessly. By exploiting this distortion-tolerance at the network protocol level we can enable the gateway to adapt to unpredictable system loads in real-time.

As a second application of these ideas, consider an autonomous sensor vehicle such as a submarine or interplanetary probe. The goal of this system is to provide data for human use. As in the wired-to-wireless gateway example, the system load is unpredictable. In this case unpredictability exists both in the input process, caused by the unknown rate at which the vehicle observes phenomena of interest, and in the output process, caused by variations in channel capacity resulting from changing environmental conditions. Because the system has finite memory resources, buffer overflows are an issue. Since sensor data is similar to multimedia content in the sense that it is distortion-tolerant, a set of protocols similar to those developed for gateway manager applications can be used here.

5.2 Fundamentals

Consider a finite-memory queue buffering distortion-tolerant data. If the rate of arrivals exceeds the queue's communication rate over a span of time, the queue will overflow because of memory limitations. The ensuing uncontrolled data loss can result in large increases in overall distortion. In this chapter we design buffering protocols that use the distortion-tolerance of the queued signals to lower the fidelity at which signals are stored in a controlled manner, freeing memory resources to avoid overflows. This approach yields performance gains in terms of distortion and delay, as compared to baseline algorithms that treat all content as distortion-intolerant, and so are unable to adjust the fidelity of the stored signals dynamically.

The basic tool we use are multiresolution source codes, e.g., [26, 66]. Such codes have a very special ordered data structure, starting with a most significant layer of description to which can progressively be added refinements that increase the fidelity of the signal reconstruction. Such a layered structure allows the source to be reconstructed progressively as each refinement becomes available, rather than having to wait until the whole code is available. From the opposite perspective, if we start with all the descriptions then least-significant refinements can be deleted first in order to free memory resources, leaving the more significant descriptions unperturbed. This latter

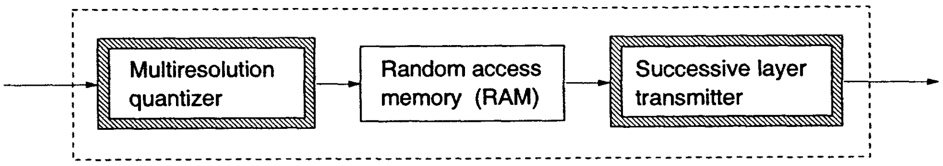

Multiresolution quantizer
Random access memory (RAM)
Successive layer transmitter

Figure 5.1. Internal structure of memory interface.

point of view is central to our work.

Once the data is encoded in a layered manner, a natural pair of storage and transmission algorithms emerge. On the one hand, if the buffer is about to overflow, description layers can be deleted to free up memory space. A natural way to do this is to start by deleting the least-significant layers, freeing up memory while incurring the least distortion. On the other hand, most-significant descriptions should be transmitted first in order to minimize distortion and delay. This algorithmic structure is shown in Fig. 5.1. We formalizes these intuitive ideas in the following sections and show that they form the basis for a near-optimal adaptive memory interface.

■ 5.3 System Model: Finite-Memory Buffers, Multiresolution Source Codes

We model the buffers as $M/M/1$ queues – Poisson arrivals and exponential service times – with finite memory, M_{tot}. The input data stream is a sequence of full-resolution signals where the ith signal to arrive is denoted $\mathbf{s}_i$. The Poisson arrival stream has rate λ which means that in an interval of τ seconds $\lambda\tau$ signals are expected. At the output of the queue, packets of M_{pac} bits are emitted according to Poisson process with rate μ_{pac}, giving an average transmission rate of $M_{\text{pac}}\mu_{\text{pac}}$ bits/sec. Our goal is to design a signal processing interface that manages the size-M_{tot} random access memory (RAM) buffer to minimize end-to-end distortion. This model is depicted in Fig. 5.2.

The output data stream is well described by two parameters. The first is the number of packet transmissions needed to empty the memory. This "time-to-empty" constant is defined as

$$\tau_{\text{emp}} \equiv \frac{M_{\text{tot}}}{M_{\text{pac}}}. \tag{5.1}$$

The second is the "packet-normalized" utilization rate which measures the ratio of input to output rates:

$$\rho_{\text{pac}} \equiv \frac{\lambda}{\mu_{\text{pac}}}. \tag{5.2}$$

To quantify the performance of our algorithm we introduce a distortion measure that we want to minimize. The distortion measure quantifies the fidelity d of a given signal approximation that the network is handling as a function of rate. In Section 5.7 we discuss extending the distortion measure to a more general performance measure that captures, for example, issues of delay and processing demands. Without loss of

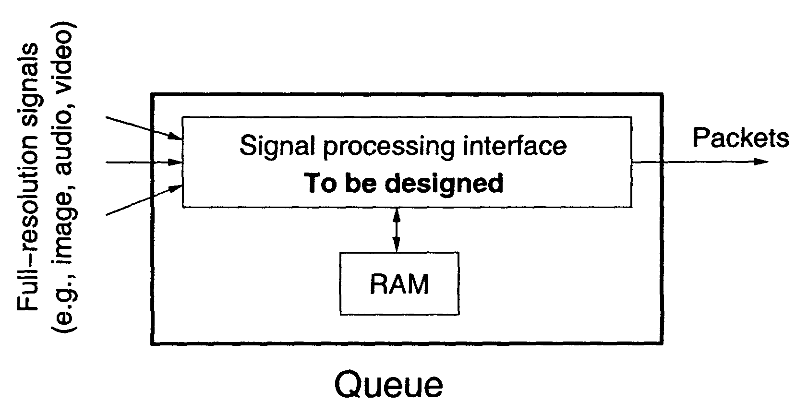


Figure 5.2. We designed the memory interface of a basic RAM of size M_{tot} to manage the buffering and distribution of distortion-tolerant data. Each signal arrival is quantized by the interface and stored in the RAM. Each departure is a packet of size M_{pac}. The arrival and departure statistics are modeled as a $M/M/1$ queuing process with arrival rate λ and departure rate μ_{pac}.

generality we assume that the source code used achieves a distortion-rate trade off according to

$$d = \exp(-f(R)), \tag{5.3}$$

where R is the rate assigned to the source, bits/signal. Traditionally, rate is measured in bits/sample, but we use this form to simplify the discussion. The change from bits/signal to bits/sample can be effected by suitable redefinition of $f(\cdot)$. Since the function $f(\cdot)$ can be arbitrary, the form of (5.3) doesn't put any restriction on the distortion-rate trade off. In our work we assume that the function $f(\cdot)$ is a monotonically increasing concave function of its argument, giving a distortion-rate trade off (5.3) that decreases monotonically in R, and is convex. The maximum distortion, incurred when no information about a received signal is communicated onwards by the buffer, is found by setting $R = 0$, i.e.

$$d_{\max} = \exp(-f(0)). \tag{5.4}$$

Multiresolution source codes are composed of K ordered subcodes, $C_1, \ldots, C_K$, of rates $R_1, \ldots, R_K$, respectively. Using subcodes $C_1, \ldots, C_k$ (where $k \leq K$), the source can be reconstructed to distortion $\exp(-f(\sum_{i=1}^{k} R_i))$. If a multiresolution source code is optimal at each step, i.e., if $R[\exp(-f(\sum_{i=1}^{l} R_i))] = \sum_{i=1}^{l} R_i$, where $R[\cdot]$ is the rate-distortion function for the source distortion pair, it is called a *successively refinable* source code [26, 57].

Example: Uniform Random Variables, Absolute Distortion, Scalar Quantization. In this case each signal s_i is a random variable distributed uniformly over $[0,1]$. The distortion measure is $d = |s_i - \hat{s}_i|$. Using nested uniform scalar quantizers of rates $R_1, \ldots, R_K$, we can achieve an average distortion-rate trade-off $E[d] = 2^{-\sum_{i=1}^{k} R_i} = \exp[-\log_e(2) \sum_{i=1}^{k} R_i]$ for all k such that $0 \leq k \leq K$. □

Example: I.i.d. Gaussian Random Vectors, Mean-Squared Distortion, Vector Quantizers. This source-distortion pairing is successively refinable [26]. Therefore given $\mathbf{s}_l \sim \mathcal{N}(0, \sigma_x^2 \mathbf{I})$ and K codes $\mathcal{C}_1, \ldots, \mathcal{C}_K$, we can achieve average distortion $E[d] = \sigma_x^2 2^{-2\sum_{i=1}^{k} R_k}$ for any k such that $0 \leq k \leq K$.□

Rather than focusing on any particular multiresolution source coding scheme, we instead develop fundamental limits on the performance of any buffering algorithm, in terms of $f(\cdot)$. This results in a benchmark against which specific algorithms can be compared. Furthermore, for simplicity, we assume all sources have the same $f(\cdot)$ function, and our objective is to minimize average distortion.

5.3.1 Lower Bound on Distortion — All Algorithms

In this section we derive a bound on the average distortion-rate performance of any buffering algorithm. This result will give us a bound with which we can compare the performances of the algorithms developed later in the chapter. We derive this bound by letting the size of the memory M_{tot} grow arbitrarily large. In this case we are no longer limited by memory since we can store losslessly for all time, all observations. This implies that we will be able to perform at least as well as any finite-memory system. This means that system performance is thus limited by the communication rate. We derive the lower bound on average distortion as follows

$$\begin{aligned} E[d] &= E[\exp[-f(R)]] \geq \exp[-f(E[R])] && (5.5) \\ &= \exp[-f(M_{\text{pac}}\mu_{\text{pac}}/\lambda)] = \exp[-f(M_{\text{tot}}/\tau_{\text{emp}}\rho_{\text{pac}})]. && (5.6) \end{aligned}$$

Eq. (5.5) follows from Jensen's inequality since the distortion is a convex function of the rate, and (5.6) follows from the average communication rate, the derivation of which we discuss next. The source and transmission models from Section 5.3 imply that in one second there are λ observations on average, and $M_{\text{pac}}\mu_{\text{pac}}$ bits are transmitted. This gives an average signal description rate $E[R] = M_{\text{pac}}\mu_{\text{pac}}/\lambda = M_{\text{pac}}/\rho_{\text{pac}} = M_{\text{tot}}/\tau_{\text{emp}}\rho_{\text{pac}}$. A good interface design for finite-sized memories tries to manage the fidelities of the signals in memory to get as close to this average as possible. To illustrate the bound (5.6) we specialize to a distortion measure that decays exponentially in rate, $d = \exp(-0.1R)$. In Fig. 5.3 we plot the performance bound for this distortion measure.

5.4 Algorithm Design

In this section we design two memory algorithms. The first, discussed in Section 5.4.1, is the baseline algorithm that treats all data as distortion-intolerant. The second, discussed in Section 5.4.2, is an adaptive algorithm that uses multiresolution source codes to take advantage of the distortion-tolerance of the signals being routed. This second design is robust to uncertainty in the queue operating conditions, as parameterized by ρ_{pac}, the packet-normalized utilization rate (5.2).

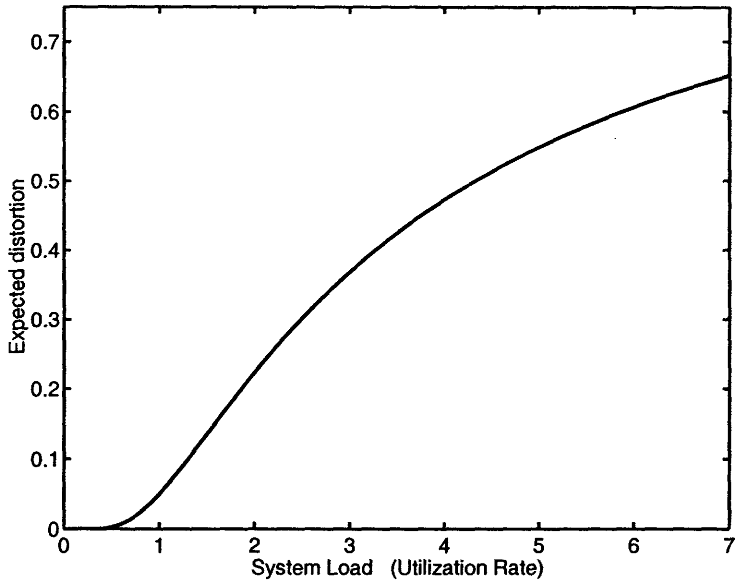


Figure 5.3. Bound on achievable average distortion for $d = \exp(-0.1R)$ as a function of system utilization.

5.4.1 Baseline Algorithm: A Network Layered Approach

The baseline algorithm does not use multiresolution source coding. When any part of a non-multiresolution source code is lost, the code becomes completely corrupted, and therefore useless. For this reason the rates at which the signals are described cannot be varied dynamically. This type of algorithm is static; the designer must decide a priori at what rate to quantize each full-resolution signal received. Since source fidelity is not adjusted dynamically based on the state of the buffer, this algorithm is appropriate for use in a network with a layered architecture where source coding is separated from network management.

Define κ_{mem} to be the maximum number of signals that can be stored in memory at any one time. Let R_i, $i = 1, \ldots, \kappa_{\text{mem}}$ be the effective rates of description (bits/signal) of each stored signal. The average distortion of the signal descriptions in memory is

$$E\left[d_{\text{mem}}; \kappa_{\text{mem}}\right] = \frac{1}{\kappa_{\text{mem}}} \sum_{i=1}^{\kappa_{\text{mem}}} \exp(-f(R_i)) \tag{5.7}$$

where d_{mem} is the distortion of the stored signals. The expected distortion is parameterized by κ_{mem}. When designing the baseline system, the designed has two choices: 1) the choice of κ_{mem}, and 2) given κ_{mem}, the choice of the R_i. Given any choice of

κ_{mem}, we now show that choosing the the R_i equal minimizes the average distortion:

$$E\left[d_{\mathrm{mem}}; \kappa_{\mathrm{mem}}\right] \geq \exp\left[-f\left(\frac{1}{\kappa_{\mathrm{mem}}}\sum_{i=1}^{\kappa_{\mathrm{mem}}} R_i\right)\right] \tag{5.8}$$

$$\geq \exp[-f(M_{\mathrm{tot}}/\kappa_{\mathrm{mem}})], \tag{5.9}$$

where (5.8) follows from applying Jensen's inequality to (5.7), and (5.9) follows from $\sum R_i \leq M_{\mathrm{tot}}$.

Equality can be achieved in (5.8) and (5.9) by letting $R_i = M_{\mathrm{tot}}/\kappa_{\mathrm{mem}}$ for all i. Thus, the optimal choice is to use equal-rate quantizers for each signal, resulting in an average quantization distortion equal to $\exp[-f(M_{\mathrm{tot}}/\kappa_{\mathrm{mem}})]$. Because before any signal can be decoded the entire non-layered source code must be received, the baseline algorithm should transmit its queued signals one at a time (e.g., FIFO).

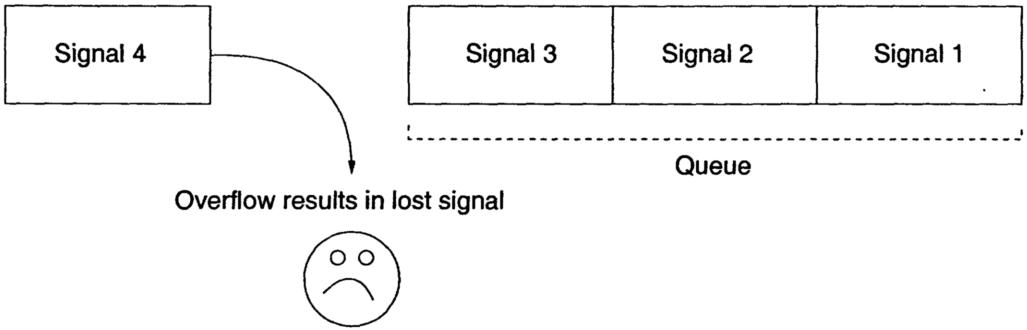
Algorithm for Baseline Algorithm

- **Initialization:**
  Divide the memory into κ_{mem} blocks of size $M_{\mathrm{tot}}/\kappa_{\mathrm{mem}}$.
- **Transmission:**
  Send bits from only a single source code until the whole code has been sent, then switch to the next signal.
- **Storage:** Of newly received signal $\mathbf{s}_m$.
  - (a) If the queue is not full then assign the signal to one of the available memory blocks. The signal is encoded at distortion $d = \exp[-f(M_{\mathrm{tot}}/\kappa_{\mathrm{mem}})]$.
  - (b) If the queue is full the new signal cannot be stored, is lost, and incurs distortion $d_{\max} = \exp[-f(0)]$.

A pictorial representation of the baseline algorithm is depicted in Fig. 5.4 with $\kappa_{\mathrm{mem}} = 3$. If a new signal arrives when the queue is full, an overflow occurs, the signal cannot be stored and is lost. This is depicted in Fig. 5.4-a. On the other hand, when a departure (packet transmission) occurs as is depicted in Fig. 5.4-b, all packet bits should be dedicated to sending a single signal's code until the whole code is transmitted. This is because decoding cannot begin until the code is completely received.

Let's now consider how this algorithm can be modified using multiresolution source codes. If a new signal arrives when the buffer is full, instead of loosing that signal completely, we can make room for it in the queue's memory by deleting the least significant information. In Fig. 5.5-a we indicate the mechanics of this "squeeze" algorithm by crossing off the least significant information to be deleted. On the other hand, when a new packet is to be transmitted, we can pick out the most significant information as shown in Fig. 5.5-b. This transmission protocol maximizes the probability that the most significant information makes it to the destination, and has the added benefit that

a. New signal arrives, but queue full



b. Departure

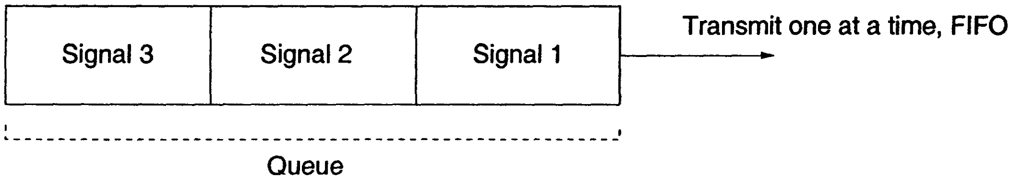


Figure 5.4. Basic idea of baseline protocol. Information is stored at constant fidelity. If new signals arrive when the queue is full they are lost. Transmissions should focus on sending the stored signals one at a time.

it also minimizes the delay on the transmission of this information. As the number of arrivals and departures grows into the tens, hundreds, and thousands, the decisions on what to keep in memory and what to transmit become more complex. In the next section we show how to turn these decisions into tractable optimization problems.

■ 5.4.2 Inter-Layer Adaptive Algorithms with Distortion-Control

The adaptive algorithm consists of two sub-algorithms with parallel structure. The first is an extraction algorithm that prioritizes descriptive layers for inclusion in the next packet. This is the half of the algorithm depicted by Fig. 5.5-b. In effect, this algorithm concatenates layers into a super-packet. The second is a storage algorithm that determines how to shuffle memory resources in order to store a newly received signal. This is the half of the algorithm depicted by Fig. 5.5-a.

Extraction and Transmission Algorithm

Suppose signals $\mathbf{s}_1, \dots, \mathbf{s}_{m-1}$ have been received and stored. Define $R_{\mathrm{dec},i}$ and $R_{\mathrm{qu},i}$, $i = 1, \dots, m-1$, respectively as the number of bits describing $\mathbf{s}_i$ already at the decoder (i.e., already transmitted by the buffer), and still retained in the queue's memory. The

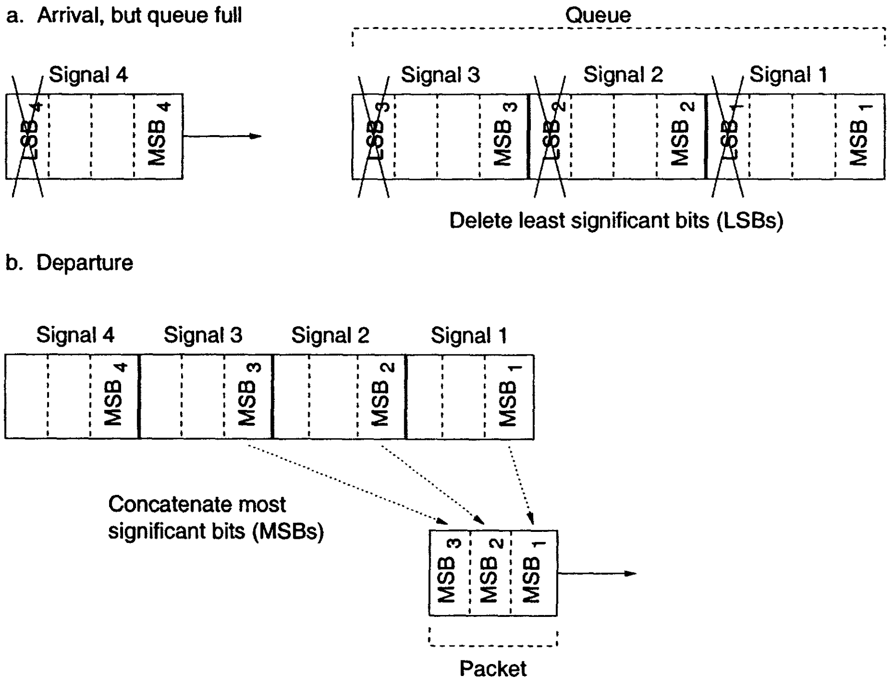

a. Arrival, but queue full
Queue
Signal 4
Signal 3
Signal 2
Signal 1
LSB 4
MSB 4
LSB 3
MSB 3
LSB 2
MSB 2
LSB 1
MSB 1
Delete least significant bits (LSBs)
b. Departure
Concatenate most significant bits (MSBs)
Packet

Figure 5.5. Basic idea of adaptive protocol. Signals are stored at variable fidelities. If a new signal arrives when the queue is full, all signals are "squeezed" to fit at lower fidelities. Transmissions should send first the most important information stored in memory, determined across all stored signals.

average distortion at the decoder is $E[d] = \frac{1}{m-1}\sum_{i=1}^{m-1}\exp(-f(R_i))$. Define $\delta_{\text{pac},i}$ to be the number of bits from the encoding of signal $\mathbf{s}_i$, to be included in the next packet.

We use Lagrange multipliers to determine the optimal choice for the $\delta_{\text{pac},i}$. The problem is constrained so that the total sum of transmitted bits does not exceed the size of the packet, i.e. $\sum_{i=1}^{m-1}\delta_{\text{pac},i} \leq M_{\text{pac}}$, and so that $0 \leq \delta_{\text{pac},i} \leq R_{\text{qu},i}$ for all i. The cost functional is

$$\mathcal{L} = \frac{1}{m-1}\sum_{i=1}^{m-1}\exp[-f(R_{\text{dec},i}+\delta_{\text{pac},i})] + \lambda\left(\sum_{i=1}^{m-1}\delta_{\text{pac},i} - M_{\text{pac}}\right). \tag{5.10}$$

If $\delta_{\text{pac},i}$ is in this range for all i we can differentiate (5.10) with respect to $\delta_{\text{pac},j}$ to get

$$\begin{aligned} \frac{d\mathcal{L}}{d\delta_{\text{pac},j}} = -f'(R_{\text{dec},j}+\delta_{\text{pac},j})\exp[-f(R_{\text{dec},j}+\delta_{\text{pac},j})] + \lambda &= 0, & (5.11)\\ f'(R_{\text{dec},j}+\delta_{\text{pac},j})\exp[-f(R_{\text{dec},j}+\delta_{\text{pac},j})] &= \lambda, & (5.12) \end{aligned}$$

where we have subsumed the $\frac{1}{m-1}$ into the definition of λ. Eq. (5.12) tells us when deciding what to transmit next, an optimal policy is to even out the description rates at the decoder.

We now specialize $f(\cdot)$ to affine function, i.e., $f(R) = \alpha R + \beta$.[1] Substitute this form of $f(\cdot)$ into (5.12) results in

$$R_{\text{dec},j} + \delta_{\text{pac},j} = \frac{1}{\alpha}\left[\log\left(\frac{\alpha}{\lambda}\right) - \beta\right]. \tag{5.13}$$

Again (5.13) tells us that the objective of each packet transmission is to even out the a posteriori description rates at the receiver. Sometimes, however, this is not possible because of, e.g., packet-size constraints or because $\delta_{\text{pac},i} \geq 0$. To find the optimal choices for the $\delta_{\text{pac},i}$ while taking into account these active constraints we must use the the Kuhn-Tucker conditions. This results in the following theorem.

Theorem 6 *Given $f(R) = \alpha R + \beta$, and the a priori rate allocations $R_{\text{qu},i}$ and $R_{\text{dec},i}$. Then, the optimal choices for the $\delta_{\text{pac},i}$, are:*

$$\delta_{\text{pac},i} = \min\left\{\max\left\{0, \frac{1}{\alpha}\left[\log\left(\frac{\alpha}{\lambda}\right) - \beta\right] - R_{\text{dec},i}\right\}, R_{\text{qu},i}\right\}, \tag{5.14}$$

where λ is chosen so that $\sum_{i=1}^{m-1} \delta_{\text{pac},i} = M_{\text{pac}}$. The a posteriori bit allocations are $R_{\text{dec},i}|_{\text{new}} = R_{\text{dec},i} + \delta_{\text{pac},i}$ and $R_{\text{qu},i}|_{\text{new}} = R_{\text{qu},i} - \delta_{\text{pac},i}$.

This method for determining which bits are most important to transmit is akin to "water-filling" for colored Gaussian channels in channel-coding theory, and is illustrated in Fig. 5.6.

Storage Algorithm

Now, suppose a new signal $\mathbf{s}_m$ is received and must be stored. The a priori buffer rate allocations, $R_{\text{qu},i}$, $i = 1, \ldots, m-1$ upper-bound the a posteriori rate allocations after $\mathbf{s}_m$ has been received. Since, at most, all memory resources can be assigned to store $\mathbf{s}_m$, its a posteriori rate allocation is upper-bounded by M_{tot}; therefore set $R_{\text{qu},m} = M_{\text{tot}}$. Define $\{\delta_{\text{qu},i}\}$, $i = 1, 2, \ldots, m$, to be the changes made in order to store $\mathbf{s}_m$.

We again use Lagrange multipliers to determine the optimal choices for the $\delta_{\text{qu},i}$. The total sum of a posteriori bit allocations is constrained not to exceed the total amount of memory resources, i.e. $\sum_{i=1}^{m}(R_{\text{qu},i} - \delta_{\text{qu},i}) \leq M_{\text{tot}}$, and $0 \leq \delta_{\text{qu},i} \leq R_{\text{qu},i}$ for all i. We want to make this constraint an equality so as to maximize the use of memory resources, thereby minimizing distortion. The cost functional is:

$$\mathcal{L} = \frac{1}{m}\sum_{i=1}^{m} \exp[-f(R_{\text{dec},i} + R_{\text{qu},i} - \delta_{\text{qu},i})] + \lambda\left(\sum_{i=1}^{m}(R_{\text{qu},i} - \delta_{\text{qu},i}) - M_{\text{tot}}\right), \tag{5.15}$$

---

[1] Affine $f(\cdot)$ cover, e.g., the case of successive refinement codes for white Gaussian sources under a mean-squared distortion measure where $\alpha = 2\log_e(2)$ and $\beta = -\log_e(\sigma_x^2)$, resulting in $d = \exp(-f(R)) = \sigma_x^2 2^{-2R}$.

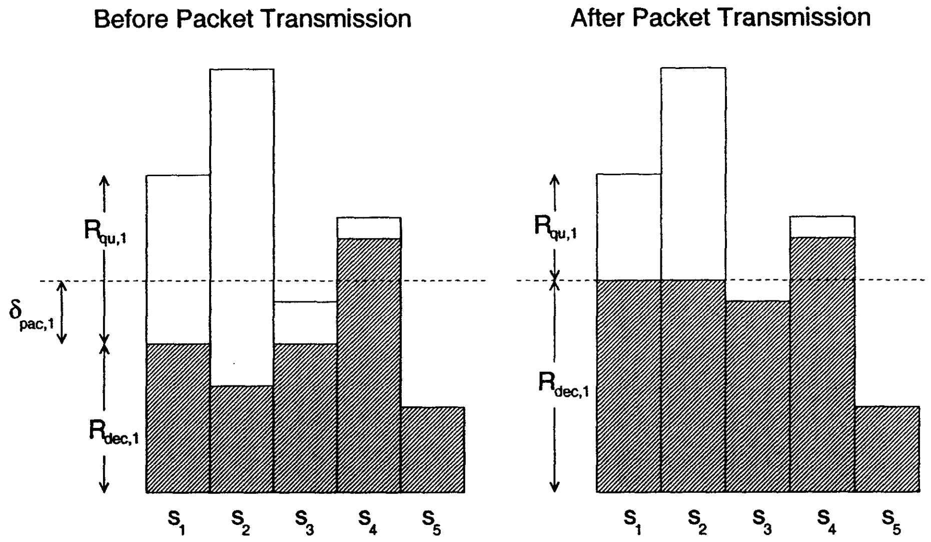


Figure 5.6. Determine packet contents by "water-filling" to the dashed line, which satisfies $\sum_{i=1}^{m-1} \delta_{\text{pac},i} = M_{\text{pac}}$. Shaded rectangles indicate $R_{\text{dec},i}$ and white rectangles $R_{\text{qu},i}$.

where $0 \leq \delta_{\text{qu},i} \leq R_{\text{qu},i}$. If $0 < \delta_{\text{qu},i} < R_{\text{qu},i}$ for all i we can differentiate (5.15) with respect to $\delta_{\text{qu},j}$ to get

$$\frac{d\mathcal{L}}{d\delta_{\text{qu},j}} = f'(R_{\text{dec},j} + R_{\text{qu},j} - \delta_{\text{qu},j}) \exp[-f(R_{\text{dec},j} + R_{\text{qu},j} - \delta_{\text{qu},j})] - \lambda = 0$$

$$\lambda = f'(R_{\text{dec},j} + R_{\text{qu},j} - \delta_{\text{qu},j}) \exp[-f(R_{\text{dec},j} + R_{\text{qu},j} - \delta_{\text{qu},j})]. \tag{5.16}$$

where we have subsumed the $\frac{1}{m}$ into the λ. Eq. (5.16) tells us that after the storage of the new signal, the optimal a posteriori overall description rates (in the queue and at the decoder) are uniform across the signals. Specializing (5.16) to $f(R) = \alpha R + \beta$ results in

$$R_{\text{dec},j} + R_{\text{qu},j} - \delta_{\text{qu},j} = -\frac{1}{\alpha} \log \left[\frac{\lambda}{\alpha}\right] - \frac{\beta}{\alpha}.$$

In general, however, some of the $\delta_{\text{qu},i}$ will equal 0 or $R_{\text{qu},i}$. To find the optimal choice for $\delta_{\text{qu},i}$ while taking into account these active constraints we must use the the Kuhn-Tucker conditions. We state the result for affine $f(R) = \alpha R + \beta$.

Theorem 7 *Given $f(R) = \alpha R + \beta$, and the a priori rate allocations, $R_{\text{dec},i}$ and $R_{\text{qu},i}$. Then, the optimal choices for the $\delta_{\text{qu},i}$, are:*

$$\delta_{\text{qu},i} = \min\left\{\max\left\{0, R_{\text{dec},i} + R_{\text{qu},i} + \frac{\beta}{\alpha} + \frac{1}{\alpha}\log\left(\frac{\lambda}{\alpha}\right)\right\}, R_{\text{qu},i}\right\} \tag{5.18}$$

where λ is chosen so that $\sum_{i=1}^{m}(R_{\text{qu},i} - \delta_{\text{qu},i}) = M_{\text{tot}}$. The a posteriori bit allocations are $R_{\text{dec},i}|_{\text{new}} = R_{\text{dec},i}$ and $R_{\text{qu},i}|_{\text{new}} = R_{\text{qu},i} - \delta_{\text{qu},i}$.

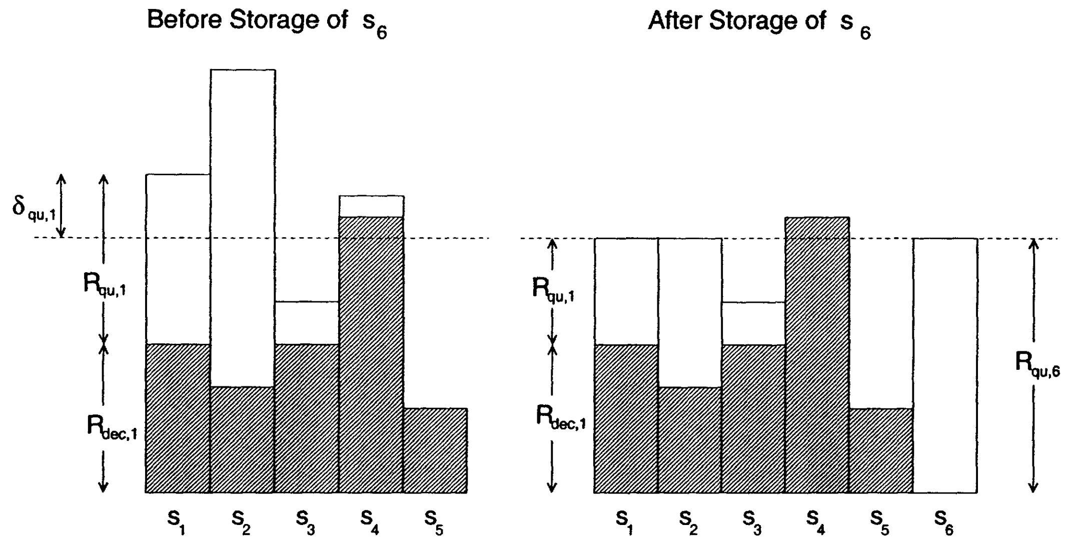


Figure 5.7. Determine memory re-allocations by "water-filling" to the dashed line, which satisfies $\sum_{i=1}^{m-1}(R_{\mathrm{qu},i} - \delta_{\mathrm{qu},i}) = M_{\mathrm{tot}}$. Shaded rectangles indicate $R_{\mathrm{dec},i}$ and white rectangles $R_{\mathrm{qu},i}$.

This method for determining bit allocations is again akin to "water-filling" for colored Gaussian sources in channel coding theory, and is illustrated in Fig. 5.7.

Algorithms for Adaptive Algorithm

- **Extraction and Transmission:**
  (a) Calculate $\{\delta_{\mathrm{pac},i}\}$ according to Thm. 6.
  (b) Concatenate the most significant $\delta_{\mathrm{pac},i}$ bits of each $\mathbf{s}_i$ into a packet.
  (c) Transmit the packet.
  (d) Increase $R_{\mathrm{dec},i}$ by $\delta_{\mathrm{pac},i}$, and decrease $R_{\mathrm{qu},i}$ by $\delta_{\mathrm{pac},i}$.
- **Storage:** Of newly received signal $\mathbf{s}_m$.
  (a) Calculate $\{R_{\mathrm{qu},i} - \delta_{\mathrm{qu},i}\}$ according to Thm. 7.
  (b) Reduce the queue memory allocated $\mathbf{s}_i$ from $R_{\mathrm{qu},i}$ to $R_{\mathrm{qu},i} - \delta_{\mathrm{qu},i}$.
  (c) Store $\mathbf{s}_m$ with a multiresolution encoding at rate $R_{\mathrm{qu},m} - \delta_{\mathrm{qu},m} = M_{\mathrm{tot}} - \delta_{\mathrm{qu},m}$.

For many applications it will be necessary to introduce granularity in $\delta_{\mathrm{pac},i}$ and $\delta_{\mathrm{qu},i}$, e.g., we can only send an integer number of bits each time. We have not taken such effects into account explicitly, but they can be accommodated through the choice of the function $f(\cdot)$.

5.5 Algorithm Analysis

In this section we derive bounds on the distortion performance of the baseline and adaptive algorithms. There are two sources of distortion to consider. The first is quantization noise incurred during source coding which is increased by fidelity reduction. The second is overflow distortion, incurred when the buffer memory is full, a new signal arrives the buffer overflows resulting in a lost signal, thereby increasing distortion. Note that the increase in quantization noise via fidelity reduction is only experienced by the adaptive algorithm, while losses due to buffer overflows are only experienced by the baseline algorithm.

Putting the two sources together we get an expression for average distortion

$$E[d] = E[d_{\text{mem}} | m \geq 1] \Pr[\text{not lost}] + d_{\max} \Pr[\text{lost}], \tag{5.19}$$

where d is the overall distortion, d_{mem} is the distortion of files in memory (we condition on the event that the memory is not empty, since if the memory is empty there are no signals to calculate the distortion of) and $d_{\max}$ is the distortion incurred when a signal is lost (5.4).

5.5.1 Steady State Performance of Baseline Algorithm

In this section we derive bounds for the baseline algorithm described in Section 5.4.1. These bounds will provide the benchmark with which we compare the performance of the adaptive algorithm. The performance of the baseline algorithm is parameterized by κ_{mem}, the total number of signal descriptions that can be stored in the buffer at any given time. Using Markov chain analysis we derive the steady-state probabilities that there are m items, $m = 0, 1, \ldots, \kappa_{\text{mem}}$, in the buffer.

Recall that the expected time to download one packet of M_{pac} bits is $1/\mu_{\text{pac}}$ seconds. There can only be κ_{mem} items in memory at any given time, and each item is assigned $M_{\text{tot}}/\kappa_{\text{mem}}$ bits. We cab convert the download rate from packets per second to signals per second as follows

$$\begin{aligned} \mu_{\text{pac}} \frac{\text{packets}}{\text{sec}} &= \mu_{\text{pac}} M_{\text{pac}} \frac{\text{bits}}{\text{sec}} \\ &= \mu_{\text{pac}} M_{\text{pac}} \frac{\kappa_{\text{mem}}}{M_{\text{tot}}} \frac{\text{signals}}{\text{sec}} \\ &= \frac{\mu_{\text{pac}} \kappa_{\text{mem}}}{\tau_{\text{emp}}} \frac{\text{signals}}{\text{sec}} \\ &\equiv \mu_{\text{sig}} \frac{\text{signals}}{\text{sec}}. \end{aligned}$$

The rate of observation λ is also expressed in terms of signals observed per second. Since nothing happens to the signals stored in memory between events, we can concentrate solely on those times when observations are made or packet transmissions occur, which is a discrete-time process.

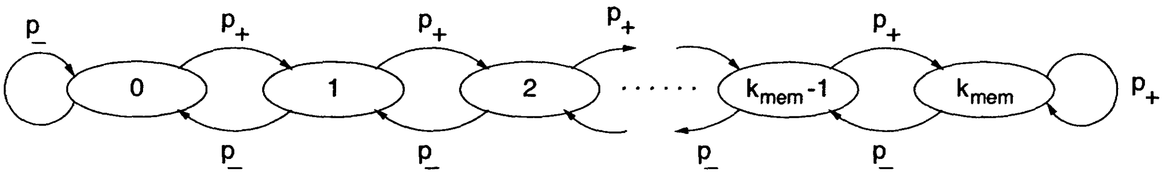


Figure 5.8. The Markov chain denoting the number of items in memory.

The number of signals in the buffer can be modeled as a Markov chain as shown in Fig. 5.8. The state at time t, m_t, indicates the number of signals in the buffer. Time is indexed on events (observations/transmissions), i.e. $t = 1, 2, \ldots$ Given m signals in memory at time t, we calculate transition probabilities from the Poisson arrival rate λ, and the exponentially distributed waiting times, parameterized by the signal-normalized inter-transmission rate $\mu_{\text{sig}} = \frac{\mu_{\text{pac}}\kappa_{\text{mem}}}{\tau_{\text{emp}}}$. Except at the ends of the chain when $m = 0$ or $m = \kappa_{\text{mem}}$,

$$p_{m_{t+1}|m_t}(m+1|m) = \frac{\lambda}{\lambda + \mu_{\text{sig}}} = \frac{\rho_{\text{sig}}}{\rho_{\text{sig}} + 1} = p_+, \tag{5.20}$$

$$p_{m_{t+1}|m_t}(m-1|m) = \frac{\mu_{\text{sig}}}{\lambda + \mu_{\text{sig}}} = \frac{1}{\rho_{\text{sig}} + 1} = p_-, \tag{5.21}$$

where ρ_{sig} is the signal-normalized utilization rate defined as

$$\rho_{\text{sig}} \equiv \frac{\lambda}{\mu_{\text{sig}}} = \lambda \frac{\tau_{\text{emp}}}{\kappa_{\text{mem}}\mu_{\text{pac}}}. \tag{5.22}$$

The reciprocal of the signal-normalized transmission rate $1/\mu_{\text{sig}}$ is the average time it takes the baseline algorithm to transmit one of its stored signals. At the ends of the chain, when $m_t = 0$, $p_- = p_{m_{t+1}|m_t}(0|0)$ and when $m_t = \kappa_{\text{mem}}$, $p_+ = p_{m_{t+1}|m_t}(\kappa_{\text{mem}}|\kappa_{\text{mem}})$.

To derive the steady-state probabilities of the Markov chain depicted in Fig. 5.8, note the following relationships,

$$\begin{aligned} p_0 &= (p_-)p_0 + (p_-)p_1 \\ p_1 &= (p_+)p_0 + (p_-)p_2 \\ p_2 &= (p_+)p_1 + (p_-)p_3 \\ &\vdots \end{aligned}$$

which can be rewritten as

$$\begin{aligned} (p_+)p_0 &= (p_-)p_1 \\ (p_+)p_1 &= (p_-)p_2 \\ (p_+)p_2 &= (p_-)p_3 \\ &\vdots \end{aligned}$$

This tells us that for $1 \leq k < \kappa_{\text{mem}}$, $p_{k+1} = (p_+/p_-)p_k$. Using this exponential relationship together with the fact that $\sum_{k=1}^{\kappa_{\text{mem}}} p_k = 1$, we can derive the steady-state probabilities that there are m items in the length-κ_{mem} queue:

$$p_m(m; \kappa_{\text{mem}}) = \begin{cases} \frac{(1-\rho_{\text{sig}})\rho_{\text{sig}}^m}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}} & 0 \leq m \leq \kappa_{\text{mem}} \\ 0 & \text{otherwise.} \end{cases} \tag{5.23}$$

A signal is "lost" when it arrives to find the queue full, so it cannot be stored. The steady-state probability that signal $\mathbf{s}_k$ is lost is

$$\begin{aligned} \Pr[\mathbf{s}_k \text{ lost}] &= \Pr[\mathbf{s}_k \text{ arrives}, m = \kappa_{\text{mem}}] \\ &= \Pr[\mathbf{s}_k \text{ arrives}] \Pr[m = \kappa_{\text{mem}}] && (5.24) \\ &= \frac{(1-\rho_{\text{sig}})\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}} && (5.25) \end{aligned}$$

where (5.24) follows from the independence of the events, and (5.25) follows because $\Pr[\mathbf{s}_k \text{ arrives}] = 1$ and by substituting in the steady state probabilities of being in each state from (5.23).[2]

To calculate the expected distortion of the baseline algorithm we recall the general expression for expected distortion from (5.19):

$$\begin{aligned} E[d; \kappa_{\text{mem}}] &= E[d_{\text{mem}} | m \geq 1] \Pr[\text{not lost}] + d_{\max} \Pr[\text{lost}] \\ &\geq \exp[-f(M_{\text{tot}}/\kappa_{\text{mem}})](1 - \Pr[\text{lost}]) + d_{\max} \Pr[\text{lost}] && (5.26) \\ &= \exp\left[-f\left(\frac{M_{\text{tot}}}{\kappa_{\text{mem}}}\right)\right]\left(\frac{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}}\right) + d_{\max}\frac{(1-\rho_{\text{sig}})\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}} && (5.27) \end{aligned}$$

where (5.26) follows from the distortion expression for the signals stored in the buffer (5.9), and (5.27) follows from the probability that any signal is lost (5.25). Equality is obtained in (5.26) by allocating equal description rates to all signals stored in the buffer.

To get a normalized distortion expression, divide (5.27) by $d_{\max} = \exp(-f(0))$ to get

$$E[d; \kappa_{\text{mem}}]_{\text{norm}} = \exp\left[-f\left(\frac{M_{\text{tot}}}{\kappa_{\text{mem}}}\right) + f(0)\right]\left(\frac{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}}\right) + \frac{(1-\rho_{\text{sig}})\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}} \tag{5.28}$$

The expected distortion given by (5.28) is a function of κ_{mem}, which is under the designer's control, but also of ρ_{sig} which is itself a function of κ_{mem}, τ_{emp}, and ρ_{pac}. The last two of these parameters are not under the designer's control and ρ_{pac} may not be known by the designer.

---

[2] Note that $\Pr[\mathbf{x}_k \text{ lost}] \neq \Pr[m = \kappa_{\text{mem}}, \text{next signal arrival}]$
$= \Pr[m = \kappa_{\text{mem}}] \Pr[\text{next signal arrival}]$
because we are calculating the probability that a particular signal, $\mathbf{s}_k$ is lost, not that a signal is lost at a particular time, which is what this calculation would yield.

If the designer knows the system operating condition as parameterized by ρ_{sig}, he can optimize the baseline algorithm to this particular utilization rate. The performance of this "optimized" baseline algorithm is found by taking the minimum across all choices of κ_{mem}:

$$\begin{aligned}\min_{\kappa_{\text{mem}}} E\left[d; \kappa_{\text{mem}}\right]_{\text{norm}} = \min_{\kappa_{\text{mem}}} \Bigg\{ & \exp\left[-f\left(\frac{M_{\text{tot}}}{\kappa_{\text{mem}}}\right) + f(0)\right]\left(\frac{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}}\right) \\ & + \frac{(1-\rho_{\text{sig}})\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}}\Bigg\}. \end{aligned} \tag{5.29}$$

Unfortunately, in many situations ρ_{sig} is unknown, and so the optimization in (5.29) cannot be performed. In these cases the adaptive algorithm becomes particularly attractive. We refer to the performance characteristic (5.29) as that of the 'optimized' baseline design, and define $\kappa_{\text{mem,opt}}$ to be the κ_{mem} that meets this bound, i.e.

$$\begin{aligned}\kappa_{\text{mem, opt}} = \arg\min_{\kappa_{\text{mem}}} \Bigg\{ & \exp\left[-f\left(\frac{M_{\text{tot}}}{\kappa_{\text{mem}}}\right) + f(0)\right]\left(\frac{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}}\right) \\ & + \frac{(1-\rho_{\text{sig}})\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}}\Bigg\}. \end{aligned} \tag{5.30}$$

■ 5.5.2 Steady State Performance of Adaptive Algorithm

We conjecture that the performance of the adaptive algorithm (Section 5.4.2) lies between the lower bound on performance given by (5.6), and the performance of the optimized baseline algorithm given by (5.29). Simulation results bear out this conjecture, examples of which are presented in the next section.

■ 5.6 Comparison of Algorithms

To compare the performances of the adaptive and baseline to each other and to the bound on all algorithms, we restrict ourselves to linear $f(R)$, $f(R) = \alpha R$. This gives a distortion-rate trade off of $d = \exp(-\alpha R)$. In Fig. 5.9 we plot typical steady state performance curves versus ρ_{pac} for $M_{\text{tot}} = 1200$, $M_{\text{pac}} = 30$, and $\alpha = 0.1$. The dotted and dash-dotted curves show the performance of the baseline algorithm as predicted by the analysis of Section 5.5.1, for $\kappa_{\text{mem}} = 200$ (low quantization rate) and $\kappa_{\text{mem}} = 80$ (high quantization rate, respectively, where quantization rate equals $M_{\text{tot}}/\kappa_{\text{mem}}$. Experimental results for these cases are shown by the +s and ×s, and closely match the analysis. The dashed curve is the performance of the optimized baseline algorithm given by (5.29). The solid curve is the performance bound on all algorithms given by (5.6). The experimental performance of the adaptive algorithm, indicated by o's, closely approximates the bound on all algorithms

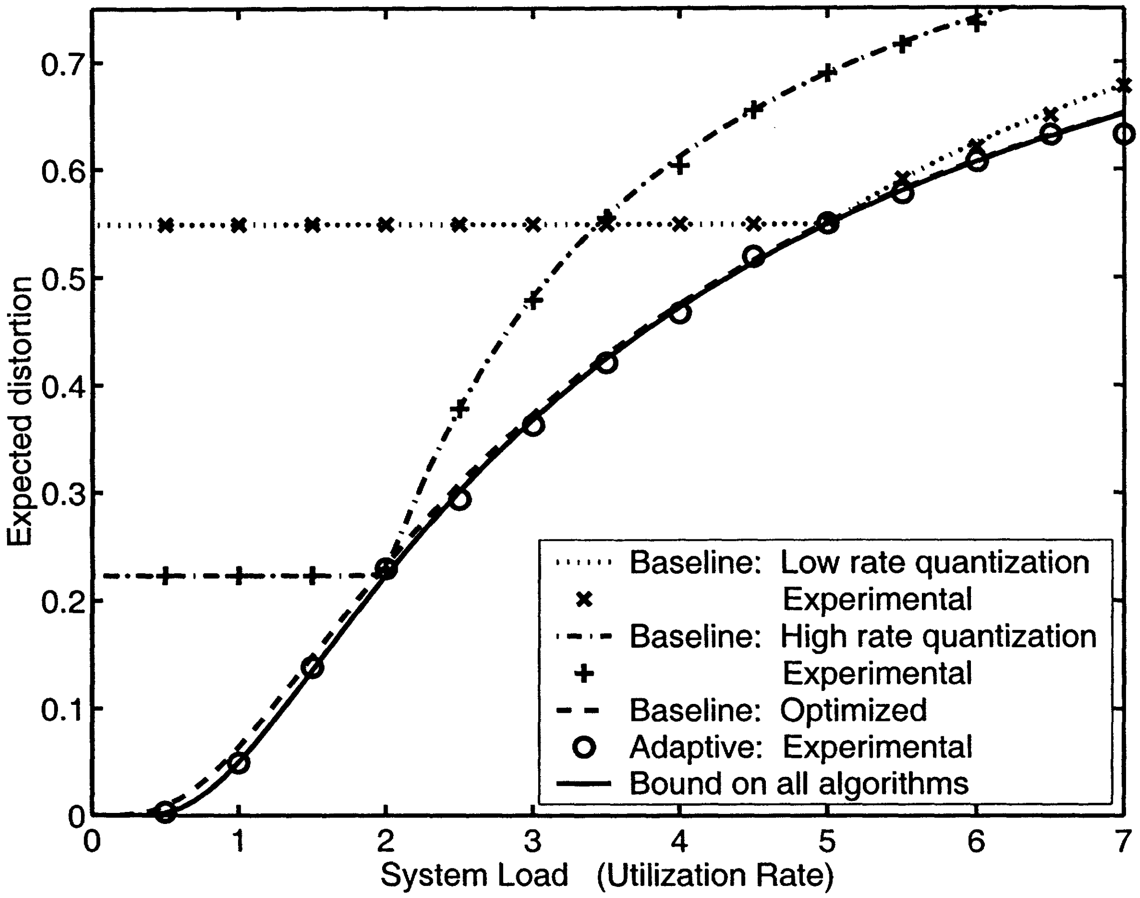


Figure 5.9. Expected distortion versus ρ_{pac}, $M_{\text{tot}} = 1200$, $M_{\text{pac}} = 30$, and $\alpha = 0.1$. The expected distortion for the low-rate ($M_{\text{tot}}/\kappa_{\text{mem}} = 1200/200 = 6$) and high-rate ($M_{\text{tot}}/\kappa_{\text{mem}} = 1200/80 = 15$) quantization cases are plotted as dotted and dash-dotted curves. Experimental confirmation of the analysis is plotted as +'s and ×'s, respectively. The choices of $\kappa_{\text{mem}} = 80, 200$ are optimal for $\rho_{\text{pac}} \simeq 5$ and $\rho_{\text{pac}} \simeq 2$, respectively. The performance of the optimized baseline and the lower bound on all algorithms are plotted as dashed and solid curves, respectively. The experimental performance of the adaptive algorithm, indicated by o's, closely approximates the lower bound.

From Fig. 5.9 we observe that the performance of the optimized baseline algorithms is quite close to the lower bound. This means that if ρ_{pac} is known, the baseline algorithm can be optimized (i.e., κ_{mem} can be chosen) to this particular ρ_{pac}, allowing us to capitalize on the computational simplicity of the baseline algorithm. The disadvantage of doing this is that the baseline performance is quite fragile and depends markedly on exact knowledge of ρ_{pac}. In situations where ρ_{pac} is uncertain or unknown, the baseline algorithm cannot be guaranteed to give good results, and the uniformly good performance of the adaptive algorithm becomes very attractive.

Figure 5.9 also illustrates the differences between the two baseline algorithms. Both algorithms have two distinct regions of operation: $\kappa_{\mathrm{mem}} < \kappa_{\mathrm{mem,opt}}$ and $\kappa_{\mathrm{mem}} > \kappa_{\mathrm{mem,opt}}$ where $\kappa_{\mathrm{mem,opt}}$ is the break point where the baseline algorithm comes closest to the bound on all algorithms. These are the *memory-constrained* and *communication-constrained* regions of operation, respectively. In the memory-constrained region the distortion is dominated by the first term in (5.19) — quantization noise — which is invariant to changes in ρ_{pac}. To meet the lower bound we must increase the rate at which each signal is quantized. The beneficial effect of increasing the quantization rate in the memory-constrained region can be seen by comparing the performances of the two algorithm when both are in the memory-constrained region ($0 \leq \rho_{\mathrm{pac}} \leq 2$ in Fig. 5.9).

In the communication-constrained region the distortion is dominated by the second term in (5.19) — memory overflow — which is an increasing function of ρ_{pac}. To meet the lower bound we must reduce the probability-of-overflow. This can be done by increasing κ_{mem} the number of signals that can be stored in memory at a given time. The beneficial effect of increasing κ_{mem} in the communication-constrained region can be seen by comparing the performances of the two algorithms when both are in the communication-constrained region ($5 \leq \rho_{\mathrm{pac}} \leq 7$ in Fig. 5.9).

To quantify the superiority of the adaptive algorithm, we approximate the extra resources (memory or communication rate) necessary for the performance of the baseline algorithm (κ_{mem} fixed) to match the performance bound on all algorithms (5.6). Matching this bound guarantees that the performance of the adaptive algorithm is also matched. Since the performance of the adaptive algorithm is quite close to the all-algorithm bound, this gives a good sense of the superiority of the adaptive algorithm.

We show that the baseline performance transitions from the memory-constrained to communication-constrained regions of operation occurs at $\rho_{\mathrm{sig}} \simeq 1$. We do the analysis in the large-κ_{mem} region because the memory densities of RAM chips today is so high that in any deployed system κ_{mem} is likely to be big. Hence, $\rho_{\mathrm{sig}} < 1$ means we are in the memory-constrained region while $\rho_{\mathrm{sig}} > 1$ implies that we are in the communication-constrained region.

■ 5.6.1 Memory-Constrained Region

We start with the steady state average distortion of the baseline algorithm from (5.28) using $d = \exp(-f(R)) = \exp(-\alpha R)$,

$$\begin{aligned}
E\left[d; \kappa_{\mathrm{mem}}\right]_{\mathrm{norm}} &= \exp[-\alpha M_{\mathrm{tot}}/\kappa_{\mathrm{mem}}] \left(\frac{1-\rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}}}{1-\rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}+1}} \right) + \frac{(1-\rho_{\mathrm{sig}})\rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}}}{1-\rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}+1}} && (5.31) \\
&\simeq \exp[-\alpha M_{\mathrm{tot}}/\kappa_{\mathrm{mem}}] && (5.32) \\
&= \exp[-\alpha M_{\mathrm{tot}} \rho_{\mathrm{sig}}/\tau_{\mathrm{emp}} \rho_{\mathrm{pac}}] && (5.33) \\
&> \exp[-\alpha M_{\mathrm{tot}}/\tau_{\mathrm{emp}} \rho_{\mathrm{pac}}], && (5.34)
\end{aligned}$$

where (5.32) follows from $\kappa_{\mathrm{mem}} \gg 1$ and $\rho_{\mathrm{sig}} < 1$, (5.33) from $\rho_{\mathrm{sig}} = \tau_{\mathrm{emp}} \rho_{\mathrm{pac}}/\kappa_{\mathrm{mem}}$, and (5.34) follows from $\rho_{\mathrm{sig}} < 1$.

Equation (5.32) confirms that the overall distortion in the memory-constrained region is dominated by the first term of (5.31), i.e. the quantization noise. In addition (5.34) is equal to the bound on all algorithms that we developed in (5.6). This tells us that the expected distortion of the baseline algorithm is strictly greater than this bound, as it should be. We now quantify how much better the adaptive system does than the baseline. The measure of improvement we use is how much memory is save by using the adaptive algorithm. In other words, how much must we increase M_{tot} by so that the inequality of (5.34) is met with equality? We call the new total memory size $\tilde{M}_{\mathrm{tot}}$. Defining $\tilde{M}_{\mathrm{tot}} = \gamma_{\mathrm{mem}} M_{\mathrm{tot}}$, then

$$\begin{aligned} \tilde{M}_{\mathrm{tot}} &= \gamma_{\mathrm{mem}} M_{\mathrm{tot}}, & (5.35)\\ \tilde{\tau}_{\mathrm{emp}} &= \gamma_{\mathrm{mem}} \tau_{\mathrm{emp}}, & (5.36)\\ \tilde{\rho}_{\mathrm{sig}} &= \frac{\rho_{\mathrm{pac}} \tilde{\tau}_{\mathrm{emp}}}{\kappa_{\mathrm{mem}}} = \frac{\gamma_{\mathrm{mem}} \rho_{\mathrm{pac}} \tau_{\mathrm{emp}}}{\kappa_{\mathrm{mem}}} = \gamma_{\mathrm{mem}} \rho_{\mathrm{sig}}. & (5.37) \end{aligned}$$

Eq. (5.36) tells us that for a fixed communication rate, as we increase the size of the memory M_{tot}, it takes longer on average to download the entire memory. Similarly (5.37) tells us that as we increase M_{tot}, the signal-normalized utilization rate increases proportionally. In other words, the system gets busier since it takes longer to transmit a signal while the arrival rate stays fixed.

For equality in (5.34) we need

$$\begin{aligned} \exp(-\alpha M_{\mathrm{tot}} / \tau_{\mathrm{emp}} \rho_{\mathrm{pac}}) &= \exp(-\alpha \tilde{M}_{\mathrm{tot}} / \kappa_{\mathrm{mem}}) & (5.38)\\ &= \exp(-\alpha \gamma_{\mathrm{mem}} M_{\mathrm{tot}} / \kappa_{\mathrm{mem}}) & (5.39) \end{aligned}$$

where (5.38) comes from setting (5.32) to (5.34). Solving (5.39) for γ_{mem} we find

$$\gamma_{\mathrm{mem}} = \frac{\kappa_{\mathrm{mem}}}{\tau_{\mathrm{emp}} \rho_{\mathrm{pac}}} = \frac{1}{\rho_{\mathrm{sig}}}. \tag{5.40}$$

But, for this choice of γ_{mem} we get

$$\tilde{\rho}_{\mathrm{sig}} = \frac{\rho_{\mathrm{pac}} \tilde{\tau}_{\mathrm{emp}}}{\kappa_{\mathrm{mem}}} = \frac{\rho_{\mathrm{pac}} \tau_{\mathrm{emp}} \gamma_{\mathrm{mem}}}{\kappa_{\mathrm{mem}}} = 1. \tag{5.41}$$

This means that we are no longer in the region where $\tilde{\rho}_{\mathrm{sig}} << 1$, so the approximate equality of (5.32) is not necessarily valid. We need to calculate (5.31) in the region as $\tilde{\rho}_{\mathrm{sig}}$ approaches unity from below, and confirm that the expected distortion is still roughly $\exp(-\tilde{M}_{\mathrm{tot}} / \kappa_{\mathrm{mem}})$.

$$\begin{aligned} \lim_{\tilde{\rho}_{\mathrm{sig}} \to 1} E\left[d; \kappa_{\mathrm{mem}}\right]_{\mathrm{norm}} = \lim_{\tilde{\rho}_{\mathrm{sig}} \to 1} \Bigg\{ & \exp\left[\frac{-\alpha \tilde{M}_{\mathrm{tot}}}{\kappa_{\mathrm{mem}}}\right] \left(\frac{-\kappa_{\mathrm{mem}} \rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}-1}}{-(\kappa_{\mathrm{mem}}+1) \rho_{\mathrm{sig}}{}^{\kappa}_{\mathrm{mem}}} \right) \\ & + \left. \frac{\kappa_{\mathrm{mem}} \rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}-1} - (\kappa_{\mathrm{mem}}+1) \rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}}}{-(\kappa_{\mathrm{mem}}+1) \rho_{\mathrm{sig}}^{\kappa_{\mathrm{mem}}}} \Bigg\} \right|_{\rho_{\mathrm{sig}}=1} \end{aligned} \tag{5.42}$$

$$
\begin{aligned}
&= \exp[-\alpha \tilde{M}_{\text{tot}}/\kappa_{\text{mem}}]\frac{\kappa_{\text{mem}}}{1+\kappa_{\text{mem}}} + \frac{1}{1+\kappa_{\text{mem}}} \\
&\simeq \exp[-\alpha \tilde{M}_{\text{tot}}/\kappa_{\text{mem}}] && (5.43) \\
&= \exp[-\alpha M_{\text{tot}}/\tau_{\text{emp}}\rho_{\text{pac}}], && (5.44)
\end{aligned}
$$

where (5.42) follows from using L'Hopital's Rule to evaluate (5.31) at $\tilde{\rho}_{\text{sig}} = 1$, (5.28) and l'Hoptial's Rule evaluated at $\tilde{\rho}_s = 1$, (5.43) follows from $\kappa_{\text{mem}} >> 1$, and (5.44) follows from $\tilde{M}_{\text{tot}} = \gamma_{\text{mem}} M_{\text{tot}}$ and (5.40). Since (5.44) is equal to the bound of (5.34), this is the correct choice for γ_{mem}. Thus, if $\rho_{\text{sig}} < 1$ we must increase our memory size by a factor $\gamma_{\text{mem}} = 1/\rho_{\text{sig}}$ in order for the baseline algorithm to do as well as the bound given by (5.34).

5.6.2 Communication-Constrained Region

Now we investigate the communication-constrained region where $\rho_{\text{sig}} > 1$ and $\kappa_{\text{mem}} \gg 1$. Starting from (5.31) we have

$$
\begin{aligned}
E\left[d; \kappa_{\text{mem}}\right]_{\text{norm}} &= \exp(-\alpha M_{\text{tot}}/\kappa_{\text{mem}}) \left(\frac{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}} \right) + \frac{(1-\rho_{\text{sig}})\rho_{\text{sig}}^{\kappa_{\text{mem}}}}{1-\rho_{\text{sig}}^{\kappa_{\text{mem}}+1}} && (5.45) \\
&\simeq \exp(-\alpha M_{\text{tot}}/\kappa_{\text{mem}}) \frac{1}{\rho_{\text{sig}}} + \left(1 - \frac{1}{\rho_{\text{sig}}}\right) && (5.46)
\end{aligned}
$$

where (5.46) follows from $\rho_{\text{sig}} > 1$ and $\kappa_{\text{mem}} \gg 1$. Equation (5.46) tells us that in the communication-constrained region, as ρ_{sig} gets increasingly larger than 1, the overall distortion is dominated by the second term of (5.45), the probability of overflow. Note that as ρ_{sig} gets significantly bigger than 1, the first and last terms of (5.46) converge to zero, leaving distortion $d_{\text{max}} = \exp(0) = 1$. If we do not know ρ_{sig}, and we guess poorly, we could end up with terrible performance.

In the memory-constrained region we were interested in how much we had to increase the total memory size M_{tot} to match the performance bound. Now, in the communication-constrained region, we want to know how much we need increase the communication rate by to meet the performance bounds. The transmission rate is parameterized by $\mu_{\text{pac}}/\tau_{\text{emp}}$, in terms of fraction of the memory downloaded per second. We increase this rate to $\gamma_{\text{com}}\mu_{\text{pac}}/\tau_{\text{emp}} = \mu_{\text{pac}}/\bar{\tau}_{\text{emp}}$ and ask, for what γ_{com} do we match (5.34), the bound on possible performance of all algorithms? If $\bar{\tau}_{\text{emp}} = \tau_{\text{emp}}/\gamma_{\text{com}}$, then

$$
\begin{aligned}
\bar{M}_{\text{tot}} &= M_{\text{tot}}, && (5.47) \\
\bar{\rho}_{\text{sig}} &= \rho_{\text{pac}}\bar{\tau}_{\text{emp}}/\kappa_{\text{mem}} = \rho_{\text{sig}}/\gamma_{\text{com}}. && (5.48)
\end{aligned}
$$

Equation (5.47) tells us the memory resources are unaffected by a change in communication resources, but (5.48) says that the signal-normalized utilization rate ρ_{sig} increases.

To solve for γ_{com} we set the lower bound on all algorithms (5.34) equal to the expected distortion (5.46):

$$\begin{aligned}
\exp(-\alpha M_{\text{tot}}/\tau_{\text{emp}}\rho_{\text{pac}}) &= \exp(-\alpha \bar{M}_{\text{tot}}/\kappa_{\text{mem}})\frac{1}{\bar{\rho}_{\text{sig}}} + \left[1 - \frac{1}{\bar{\rho}_{\text{sig}}}\right] && (5.49)\\
\exp(-\alpha M_{\text{tot}}/\tau_{\text{emp}}\rho_{\text{pac}}) - 1 &= \left(\exp(-\alpha M_{\text{tot}}/\kappa_{\text{mem}}) - 1\right)\frac{\gamma_{\text{com}}}{\rho_{\text{sig}}} && (5.50)\\
\gamma_{\text{com}} &= \rho_{\text{sig}}\left[\frac{1 - \exp(-\alpha M_{\text{tot}}/\tau_{\text{emp}}\rho_{\text{pac}})}{1 - \exp(-\alpha M_{\text{tot}}/\kappa_{\text{mem}})}\right]. && (5.51)
\end{aligned}$$

■ 5.6.3 Section Summary

In Fig. 5.10 we graphically summarize the results of this section. In the memory-constrained region, the total memory size M_{tot} must be increase to $\gamma_{\text{mem}}M_{\text{tot}}$ (solid curve) for the baseline performance to match the bound. In the communication-constrained region, the communication rate μ_{pac} must be increase to $\gamma_{\text{com}}\mu_{\text{pac}}$ (dotted and dash-dotted curves) for the baseline performance to match the bound. To recap, the factors γ_{mem} and γ_{com} are:

$$\begin{aligned}
\gamma_{\text{mem}} &= \frac{1}{\rho_{\text{sig}}}\\
\gamma_{\text{com}} &= \rho_{\text{sig}}\left[\frac{1 - \exp(-\alpha M_{\text{tot}}/\kappa_{\text{mem}}\rho_{\text{sig}})}{1 - \exp(-\alpha M_{\text{tot}}/\kappa_{\text{mem}})}\right].
\end{aligned}$$

If κ_{mem} is chosen poorly for a given ρ_{sig} (or ρ_{sig} changes or is unknown), the performance relative to the adaptive algorithm declines quickly, and significantly. This means that the baseline algorithm is quite fragile, compared with the adaptive one. In the communication-constrained region this fragility is more marked for small κ_{mem} (e.g., the high-rate quantization curve of Fig. 5.10). If κ_{mem} is small, the quantization rate is large, but since communication resources are limited, buffer overflows are more likely. In Fig. 5.10 this effect is indicated by the increased need for communication resources of the high-rate baseline algorithm versus the low rate baseline algorithm when trying to match the adaptive algorithm's performance.

One very useful characteristics of the adaptive algorithm is that it needs not be tuned to the specific utilization factor, ρ_{pac}. The algorithm is independent of ρ_{pac} and so works well for any particular ρ_{pac}, or across a range of ρ_{pac}. This is particularly useful when the conditions in which the system operates are unpredictable or time-varying. The preceding analysis quantifies the gain of this added robustness.

One intermediate algorithm between the adaptive and baseline algorithms would be one that does not use hierarchical source coding, but can vary the fidelity at which new signals are stored. Then, if the algorithm determines that it is loosing too many signals to buffer overflows, it can reduce the quantization rate, increasing κ_{mem}. On the other hand, if it has too many resources free much of the time it can increase the quantization rate, decrease κ_{mem}. The question is, what probability of overflow

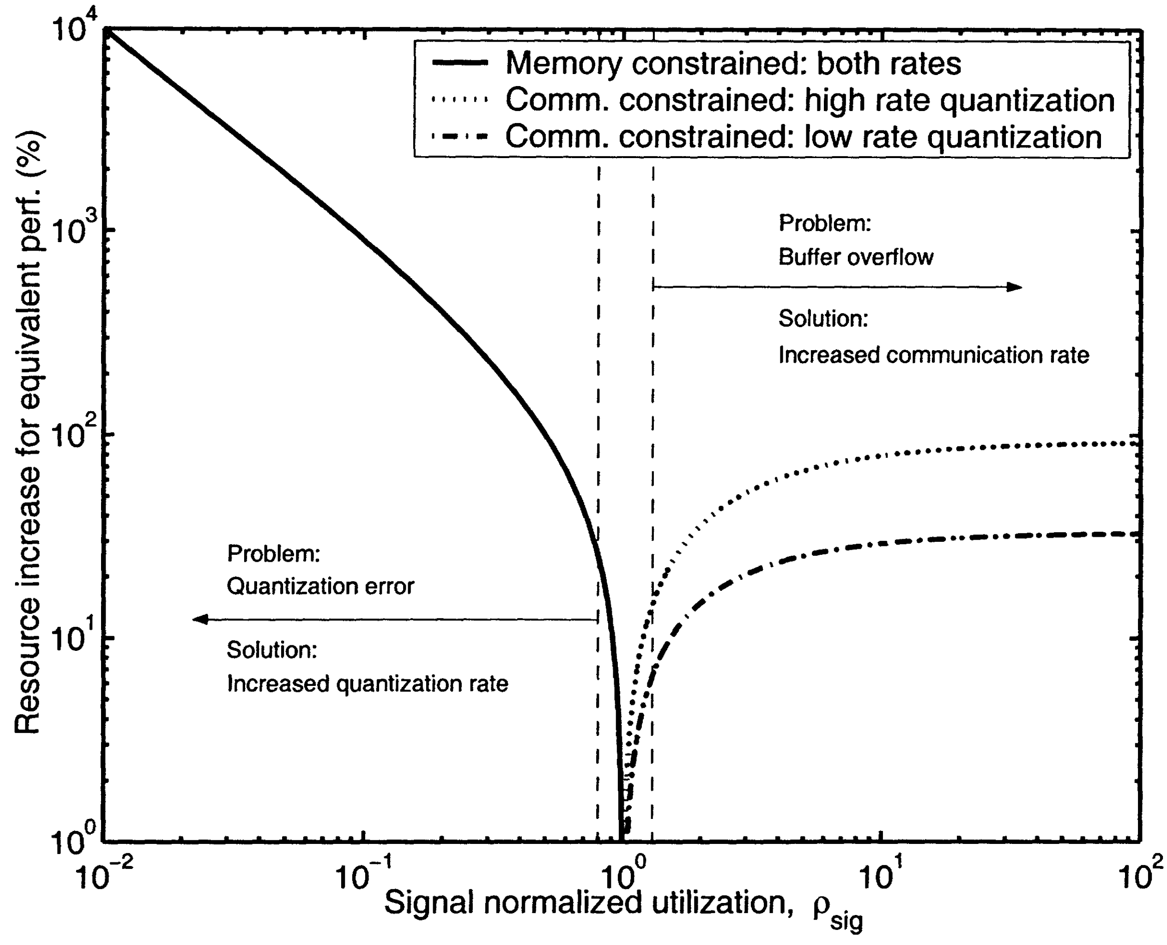


Figure 5.10. The percentage by which system resources must increase so that baseline algorithm performance is guaranteed to match adaptive algorithm performance for fixed $M_{\text{tot}} = 1200$, $\tau_{\text{emp}} = 40$, $\alpha = 0.1$, and $\kappa_{\text{mem}} = 200$ (low quantization rate) and $\kappa_{\text{mem}} = 80$ (high quantization rate). The transition from memory-constrained to communication-constrained operation occurs at $\rho_{\text{sig}} \simeq 1$.

should the algorithm try to achieve? One possible answer is given by the analysis of the probability of buffer overflow (5.25). Since the break point of the baseline algorithm is at $\rho_{\text{sig}} \simeq 1$, this would be a good region to operate in. Equation (5.25) together with l'Hopital's Rule tells us that when ρ_{sig} is approximately one, the probability of buffer overflow is approximately $1/(1 + \kappa_{\text{mem}})$. An algorithm could adjust its storage fidelity to try to match this probability of overflow would likely display performance traits between those of the adaptive and baseline algorithms.

5.7 Design Implications

In this chapter we have developed and analyzed a pair of buffering algorithms. In this section we discuss some extensions of the work to other scenarios.

Delay Constraints. In the adaptive algorithm developed in this chapter the objective was to minimize the average distortion of each signal. Because of the particular structures of the priority storage and extraction protocols (most-significant-bits through first) we minimized the delay of the most-significant-bits as a by-product. If we more directly address delay constraints we can explore some interesting implications and extensions of our results.

Consider delay-constrained applications such as voice over the Internet (VoIP). For these applications there is a maximum delay that can be associated with each resolution layer. If the resolution layer does not reach the destination before the maximum delay limit is exceeded, the network should simply drop that packet. This necessitates modification of the priority storage and extraction protocols of Section 5.4.2 to delete such now-useless information packets.

At a more subtle level, delay measures can be incorporated into the general distortion measure d, changing $d = \exp(-f(R))$ to $d = \exp(-f(R, \tau))$ where τ is now a measure of delay. Now the distortion does not solely measure reconstruction fidelity, but also the delay until that reconstruction is made.

Different Distortion Measures. Following on the idea of modifying the distortion measure to take into account reconstruction delay, we can also change the distortion measure on a per-signal basis. This would be important if we had signals of different sizes (e.g., small versus large images), but even more so for different classes of data (e.g., audio versus image or video). Furthermore, the distortion measure could also be changed to give different quality-of-services (QoS) to the different data classes. In each case, we can resolve for the optimal greedy storage and extraction protocols as in Section 5.4.2, but the complexity of the optimization will be greater because there are now multiple distortion measures.

We can also assign different distortion measures to the baseline and adaptive algorithms. We could do this if there is a distortion penalty for using multiresolution source codes. As discussed in [26], not all source codes are successively refinable. This means that the distortion-rate trade offs attainable in a single step using a block code, as the baseline algorithm does, can be better than those attainable for using a multi-step progressive code, as the adaptive algorithm does. In other words, there may a price for the extra flexibility that such a source code yields. Furthermore, the adaptive algorithm will need more header information than the baseline to route each of the layers of description. And finally, as discussed earlier, the adaptive algorithms is always going to be more computationally expensive than will the baseline. By penalizing the adaptive algorithm through a distortion measure that decreases more slowly as a function of rate we can model all these effects and better determine when the adaptive algorithm should be used.

Paralleling the discussion in Section 5.6, we define two distortion measure $d_{\mathrm{base}} = \exp(-\alpha_{\mathrm{base}}R)$ and $d_{\mathrm{adapt}} = \exp(-\alpha_{\mathrm{adapt}}R)$ and calculate γ_{mem} and γ_{com}, generalizing to two distortion measures the derivations in Section 5.6.1 and 5.6.2, respectively.

$$\gamma_{\mathrm{mem}} = \frac{\alpha_{\mathrm{adapt}}}{\alpha_{\mathrm{base}}}\frac{1}{\rho_{\mathrm{sig}}} \tag{5.52}$$

$$\gamma_{\mathrm{com}} = \rho_{\mathrm{sig}}\left[\frac{1-\exp(-\alpha_{\mathrm{adapt}}M_{\mathrm{tot}}/\kappa_{\mathrm{mem}}\rho_{\mathrm{sig}})}{1-\exp(-\alpha_{\mathrm{base}}M_{\mathrm{tot}}/\kappa_{\mathrm{mem}})}\right]. \tag{5.53}$$

For example, in Fig. 5.11, we plot the performance of a baseline algorithm with $\alpha_{\mathrm{base}} = 0.1$ as before, and $\alpha_{\mathrm{adapt}} = 0.09$. This means that the adaptive algorithm is ten percent less efficient in terms of rate. In contrast to Fig. 5.10, the baseline can now outperform the adaptive algorithm near to the normalized system load point $\rho_{\mathrm{sig}} = 1$. Outside of the dashed lines, the adaptive algorithm outperforms both baseline algorithms, but the performance gain is reduced, particularly in the communication-constrained region of operation. In general, the relations (5.52) and (5.53) can be used to give the designer a more accurate assessment of the attractiveness of the adaptive protocol design.

Memory Fragmentation. The baseline protocol is attractive in that it writes to and reads from the RAM in well defined blocks. Unless special measures are taken, use of the adaptive algorithm is likely to result in a fragmented memory where the data stored for each signal is stored in a number of different memory locations. This problem becomes increasingly acute the more fine is each layer of quantization. While the lower bound in this chapter were derived assuming that the memory is infinitely divisible, this is not a good model for real RAMs. We must choose a smallest block size of memory to work with. For instance, in the simulations of Section 5.6 we did this and the results remained close to the bounds, see Fig. 5.9. As the size of the basic block of memory is increased, the general approach remains valid (though the optimization are in fact integer programs so integer techniques or rounding must be used), but the performance curves will become less smooth because of the increasing granularity. One advantage of working with larger basic memory blocks is that memory fragmentation becomes less of a problem.

Non-Poisson Queuing Statistics. A final problem of interest is in non-$M/M/1$ queues that have more realistic traffic patterns. We conjecture that the protocols developed herein will do well for such queue because they perform robustly regardless of the true $M/M/1$ queue statistics. To show this we would have to show that on a per-sample-path basis, this algorithms performs about as well as any algorithm could. We leave this idea for future analysis.

■ 5.8 Chapter Summary

In this chapter we present a model of queues routing distortion-tolerant data. We design a pair of buffering protocols for this problem. We design the first, baseline algorithm,

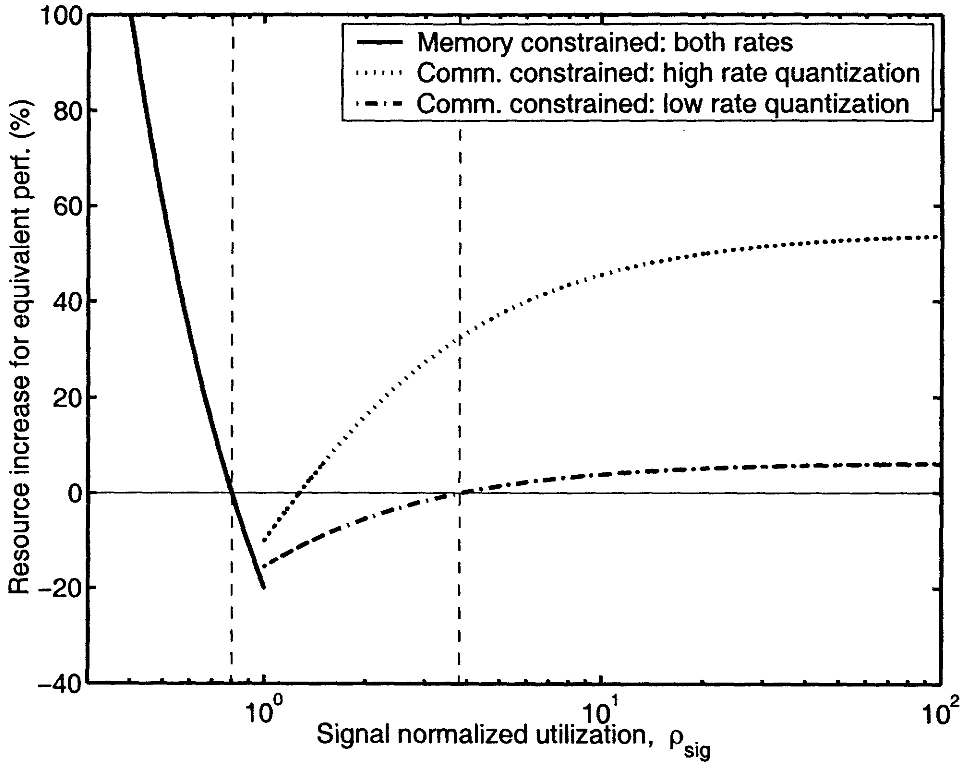


Figure 5.11. The percentage by which system resources must increase so that baseline algorithm performance is guaranteed to match adaptive algorithm performance for fixed $M_{\text{tot}} = 1200$, $\tau_{\text{emp}} = 40$, $\alpha_{\text{base}} = 0.1$, $\alpha_{\text{adapt}} = 0.09$ and $\kappa_{\text{mem}} = 200$ (low quantization rate) and $\kappa_{\text{mem}} = 80$ (high quantization rate).

to work in a network with a layered protocol stack. This algorithm does not exploit any particular knowledge of signal characteristics. We then design an adaptive algorithm that works across protocol layers and exploits the distortion-tolerant character of the signal content being handled. We quantify the performances of both algorithms and show that the adaptive algorithm is robust to uncertainty in queue statistics and closely approximates a performance bound on all algorithms. Finally, we discuss some issues of design that must be considered when applying these ideas.

Chapter 6

Conclusions

In this thesis we have developed source coding algorithms that work across traditional network layers, and make use of detailed knowledge of data characteristics. These algorithms realize substantial performance gains when compared to traditional approaches that operate within the paradigm of layered network architectures.

We found that ideas of coding with side information are particularly useful in networking contexts, because they provide insightful ways to process distributed sources of information. We built on these ideas to extend Wyner-Ziv source coding and channel coding with side information to noisy encoder observations. We developed the rate-distortion and capacity expressions for general finite-alphabet sources, and evaluated the results for the binary-Hamming and quadratic-Gaussian cases.

Using these tools and insights, we then investigated some fundamental problems of estimation under communication constraints. The coding strategies we proposed blur the boundaries between communication and estimation aspects of the problem. We investigated these problems in the context of data fusion for sensor networks. We showed how any general sensor tree can be decomposed into basic serial and parallel networks. We then took two design approaches to these prototype networks. We first developed a sense of what is possible in a layered network architecture – where the communication and estimation functions are kept separate – by designing and analyzing estimate-and-quantizer strategies. Then, designing inter-layer algorithms, we refined these basic approaches to take advantage of decoder side information. These refined designs were based on noisy Wyner-Ziv source coding. This approach led to substantial performance gains and convenient iterative distortion-rate expressions for the achievable region in quadratic-Gaussian scenarios. Using these expressions we connected our work to earlier work in the field and demonstrated that successive coding techniques can be used to achieve the rate-distortion bound in certain situations. We also discussed how the design insights provided by our results can be used when structuring communications in a sensor network, or when determining how best to allocate resources among sensor nodes.

We also showed how to design inter-layer protocols for content buffering and distribution. When combined with multiresolution source codes, knowledge of content characteristics enabled the design of memory interfaces that can adaptively trade off content fidelity for storage resources. This trade off significantly enhances end-to-end system performance, as it makes the probability that data is lost in an uncontrolled manner due to buffer overflows negligible. For purposes of comparison, we designed baseline schemes that are appropriate for use in a layered network architecture. As compared to these baseline schemes, we showed that the adaptive system performs very well in terms of distortion, delay, and system robustness – closely approximating a bound on the performance of any memory interface.

■ 6.1 Future Work

Kalman Filtering with Rate Constraints. The data fusion techniques presented in this thesis jointly address estimation and communication. The sensor network problems we considered are special cases of more general problems of sequential estimation under communication constraints. In Chapter 4 we used a Kalman Filtering analogy to discuss the structuring of coding strategies for the serial and parallel CEO problems. This discussion points to the general research topic of Kalman Filtering under rate constraints. If, in the serial CEO problem, we consider the index of each sensor node as a time index, then each vector-measurement is a time-indexed source observation. In the serial CEO problem the source $\mathbf{x}$ is constant from time sample to time sample. A more general problem would allow the source to evolve. Kalman Filtering with rate constraints is related to an emerging area of research in the control community studying feedback over rate constrained channels [68, 58]. The side information perspective might offer new insight into these problems and, as with the scalar example presented in Section 2.1.1, some ideas for implementations as well. Starting from these connections we hope to explore further this general research area.

Converse for the Serial CEO Problem. A complete theory of sensor networks on communication trees needs a performance bound: a converse. Oohama constructed a converse in [45] for the parallel network configuration in quadratic-Gaussian scenarios. Much work remains to be done in developing a converse for the serial network configuration. In the serial problem, no message from any agent except the last agent arrives at the CEO without being further degraded (because of transcoding and saturation effects). A converse for the serial problem is thus complicated by the fact that the degradation each message undergoes is a function of the coding scheme chosen.

Communication Graphs with Cycles. Understanding data fusion algorithms in sensor networks with tree-structured communication graphs is a precursor to communication graphs with loops. An example would be when two agents are able to converse (over finite rate channels) before sending jointly determined messages to the CEO. A first step in this direction would be to understand how the strategies presented herein must be modified for use in such situations. Such results might parallel those developed for iterative estimation on graphs. Many iterative algorithms, such as Belief Propagation, are exact on trees, but can often be usefully applied to graphs with cycles. The understanding of the behavior of these algorithms in such settings is a current area of research. A parallel development of iterated achievability results in information theory might be able to build on some of the emerging results in this area.

Implementations using Iterative Techniques. Iterative estimation ideas might also have a role to play in implementing the coding ideas presented herein. In the discussion of data fusion for sensor networks, all information flow was unidirectional from the agents to the CEO. In some situations, such as for the two-agents and infinite-agent parallel CEO problems, we were able to show that unidirectional information flow is enough – smoothing is not needed. It may be that when finite block-lengths are imposed, and perfectly reliable decoding can no longer be assumed, iterative decoding techniques would be more robust. For example, such techniques might be able to detect and correct for mistakes made in the middle of the decoding process.

Queuing with Distortion-Control. Exciting opportunities lie in applying ideas of queuing with distortion-control to real-world networks. The design rules we determined can be used immediately to determine the optimal packet size for baseline algorithms used in networks with steady system loads. Microchips or circuit-switched networks would be good examples. To integrate these ideas into existing or newly-defined standards, we must determine the overhead associated with protocol definition, i.e., how much extra header information is required. Following that, we would like to extend these ideas to sources with memory (e.g., video), and to test the algorithms on sequences of queues. We expect that the performances robustness of the adaptive algorithms will continue to hold in these more general situations.

Appendix A

Derivations: Noisy Wyner-Ziv Coding

A.1 Finite-Alphabet Rate-Distortion Function

In this appendix we prove a single-letter expression for the rate-distortion function $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ of source coding with side information and noisy source observations at the encoder. The rate-distortion function $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ is a tight lower bound on the rate needed to guarantee that $\frac{1}{n}E\left[\sum_{i=1}^{n} D(\mathsf{x}_i, \hat{\mathsf{x}}_i)\right]$ can be made arbitrarily close to d for a sufficiently long block length n. Formally, repeating the statement of Thm. 4 from Chapter 3, we show the following.

Theorem 8 *Let a triple of random source and observation vectors,* $(\mathbf{x}, \mathbf{y}_0, \mathbf{y}_1)$ *and a distortion measure* $D(\cdot,\cdot)$ *be given such that:*

(a) $p_{\mathbf{x},\mathbf{y}_0,\mathbf{y}_1}(\mathbf{x}, \tilde{\mathbf{y}}, \bar{\mathbf{y}}) = \prod_{i=1}^{n} p_{\mathsf{x}}(x_i) p_{\mathsf{y}_0|\mathsf{x}}(\tilde{y}_i|x_i) p_{\mathsf{y}_1|\mathsf{x}}(\bar{y}_i|x_i)$

(b) $D(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{n}\sum_{i=1}^{n} D(x_i, \hat{x}_i)$.

Then a sequence of length-n block encoder-decoder pairs can be designed such that if $\mathbf{y}_1$ *is encoded at rate R,* $\mathbf{x}$ *can be recovered to within average distortion d with arbitrarily small probability of failure as n grows to infinity if and only if*

$$R \geq R_{\mathrm{I}}^{\mathrm{WZ}}(d) = \min_{p_{\mathsf{u}|\mathsf{y}_1}(u|y_1) \in \mathcal{U}} [I(\mathsf{y}_1; \mathsf{u}) - I(\mathsf{y}_0; \mathsf{u})], \tag{A.1}$$

where the set $\mathcal{U}$ *consists of all posteriors relating the random variable* u *to the encoder observation* $\mathbf{y}_1$ *that satisfy the following conditions:*

(i) $x \leftrightarrow \mathsf{y}_1 \leftrightarrow \mathsf{u}$,

(ii) $\mathsf{y}_0 \leftrightarrow \mathsf{x} \leftrightarrow \mathsf{u}$,

(iii) $E\left[D(\mathsf{x}, f(\mathsf{y}_0, \mathsf{u}))\right] \leq d$ *for some memoryless function* $f : \mathcal{Y}_0 \times \mathcal{U} \to \hat{\mathcal{X}}$.

In the rest of this appendix we prove this theorem and evaluate the rate-distortion function for two special cases. In Section A.1.1 we show convexity of $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$. In

Section A.1.2 we show a converse, i.e., a lower-bound on $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$, for finite-alphabet sources and arbitrary distortion measures. In Section A.1.3 we demonstrate an achievable region for $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ that matches the converse of Section A.1.2 and so defines the rate-distortion function for this problem. In Section A.2 we evaluate $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ for the binary-Hamming case and in Section A.3 for the quadratic-Gaussian case.

■ A.1.1 Convexity

The noisy Wyner-Ziv rate-distortion function (A.1) is a non-increasing convex function of d. Monotonicity follows because as d increases the domain of minimization increases as well. In the rest of this section we show that $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ is convex in d.

let d_a and d_b be two distortion values, and let u_a, $f_a(\cdot,\cdot)$ and u_b, $f_b(\cdot,\cdot)$ be the corresponding auxiliary random variables and data fusion functions that achieve $R_{\mathrm{I}}^{\mathrm{WZ}}(d_a)$ and $R_{\mathrm{I}}^{\mathrm{WZ}}(d_b)$, respectively. Let q be an independent time-sharing random variable such that $\Pr(q = a) = \lambda$ and $\Pr(q = b) = 1 - \lambda$. Define $u = (q, u_q)$ and let $f(u, y_0) = f_q(u_q, y_0)$. Then the distortion becomes

$$d = E\left[D(x,\hat{x})\right] = \lambda E\left[D(x, f_a(u_a, y_0))\right] + (1-\lambda)E\left[D(x, f_b(u_b, y_0))\right] = \lambda d_a + (1-\lambda)d_b,$$

and (A.1) becomes

$$\begin{aligned} &I(y_1; u) - I(y_0; u) \\ &\quad = H(y_1) - H(y_1|u_q, q) - H(y_0) + H(y_0|u_q, q) \\ &\quad = H(y_1) - \lambda H(y_1|u_a) - (1-\lambda)H(y_1|u_b) - H(y_0) + \lambda H(y_0|u_a) + (1-\lambda)H(y_0|u_b) \\ &\quad = \lambda[I(y_1; u_a) - I(y_0; u_a)] + (1-\lambda)[I(y_1; u_b) - I(y_0; u_b)]. \qquad (A.3) \end{aligned}$$

If we define w to be the auxiliary random variable that achieves the rate-distortion bound for distortion d we have

$$\begin{aligned} R_{\mathrm{I}}^{\mathrm{WZ}}(d) &= I(y_1; w) - I(y_0; w) \\ &\le I(y_1; u) - I(y_0; u) && (A.4) \\ &= \lambda[I(y_1; u_a) - I(y_0; u_a)] + (1-\lambda)[I(y_1; u_b) - I(y_0; u_b)] && (A.5) \\ &= \lambda R_{\mathrm{I}}^{\mathrm{WZ}}(d_a) + (1-\lambda)R_{\mathrm{I}}^{\mathrm{WZ}}(d_b). && (A.6) \end{aligned}$$

where (A.4) follows because u achieves the correct distortion but does not necessarily minimize the rate, (A.5) from substituting in from (A.3), and (A.6) since u_a andu_b were defined to be rate-distortion achieving auxiliary random variables for distortions d_a and d_b, respectively.

A.1.2 Converse

In this section we show that $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ is a lower bound on the achievable rate-distortion region for the noisy Wyner-Ziv problem.

$$nR \geq H(m) \geq H(m|\mathbf{y}_0) \geq I(m;\mathbf{y}_1|\mathbf{y}_0) = \sum_{i=1}^{n} I(m; y_{1,i}|\mathbf{y}_0, \mathbf{y}_1^{i-1}) \tag{A.7}$$

$$= \sum_{i=1}^{n} \left[H(y_{1,i}|\mathbf{y}_0, \mathbf{y}_1^{i-1}) - H(y_{1,i}|\mathbf{y}_0, \mathbf{y}_{i-1}, m)\right]$$

$$= \sum_{i=1}^{n} \left[H(y_{1,i}|y_{0,i}) - H(y_{1,i}|\mathbf{y}_0, \mathbf{y}_{i-1}, m)\right] \tag{A.8}$$

$$\geq \sum_{i=1}^{n} \left[H(y_{1,i}|y_{0,i}) - H(y_{1,i}|\mathbf{y}_0, m)\right] = \sum_{i=1}^{n} \left[H(y_{1,i}|y_{0,i}) - H(y_{1,i}|y_{0,i}, u_i)\right] \tag{A.9}$$

$$= \sum_{i=1}^{n} I(y_{1,i}; u_i|y_{0,i}) = \sum_{i=1}^{n} \left[H(u_i|y_{0,i}) - H(u_i|y_{0,i}, y_{1,i})\right]$$

$$\geq \sum_{i=1}^{n} \left[H(u_i|y_{0,i}) - H(u_i|y_{1,i})\right] = \sum_{i=1}^{n} \left[I(u_i; y_{1,i}) - I(u_i; y_{0,i})\right] \tag{A.10}$$

$$\geq \sum_{i=1}^{n} R_{\mathrm{I}}^{\mathrm{WZ}}\left(E\left[D(x_i, f_{ni}(y_{0,i}, u_i))\right]\right) \tag{A.11}$$

$$\geq nR_{\mathrm{I}}^{\mathrm{WZ}}\left(E\left[\frac{1}{n}\sum_{i=1}^{n} D(x_i, f_{ni}(y_{0,i}, u_i))\right]\right) \tag{A.12}$$

$$\geq nR_{\mathrm{I}}^{\mathrm{WZ}}(d).$$

Line	Justification
(A.7)	Range of m; conditioning reduces entropy; entropy positive; chain rule.
(A.8)	The observations $\mathbf{y}_0$ and $\mathbf{y}_1$ are pairwise i.i.d.
(A.9)–(A.10)	Conditioning reduces entropy and $u_i \equiv (m, y_{0,1}, \ldots, y_{0,i-1}, y_{0,i+1}, y_{0,n})$.
(A.11)	Definition of (information) noisy Wyner-Ziv rate-distortion function where f_{ni} is the data fusion function for ith sample of the estimate in the length-n case.
(A.12)	Jensen's inequality.

A.1.3 Achievability

We now show that we can find a source code that can achieve $E\left[D(\mathbf{x}, f(\mathbf{y}_0, \mathbf{u}))\right] \leq d$ while operating at a rate R arbitrarily close to $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$. We use $T_{\mathrm{x}}^{n}(\epsilon)$ to denote the set of ϵ-strongly typical sequence of length n according to the distribution $p_{\mathrm{x}}(x)$.

Definition 1 *The set of length-n vectors $\mathbf{x}$ that are ϵ-strongly typical according to a* *finite-alphabet probability measure* $p_x(x)$ *is defined as* $T_x^n(\epsilon)$ *where*

$$T_x^n(\epsilon) = \{\mathbf{x}\} : \begin{cases} |N(x_0; \mathbf{x}) - np_x(x_0)| < n\epsilon|\mathcal{X}|^{-1} & \textit{for all } x_0 \in \mathcal{X} \;\; \textit{s.t.} \;\; p_x(x_0) > 0 \\ N(x_0; \mathbf{x}) = 0 & \textit{if } p_x(x_0) = 0, \end{cases}$$

where $N(x_0; \mathbf{x})$ *indicates the (integer) number of samples in the vector* $\mathbf{x}$ *equal to* x_0*.*

Consider a fixed $p_{u|y_1}(u|y_1)$ and function $f(u, y_0)$. The marginal for u is $p_u(u)$. We construct a rate-distortion achieving code as follows.

- **Codebook Generation:** Let $R_1 = I(y_1; u) + \epsilon$. Generate a random codebook $\mathcal{C}$ consisting of 2^{nR_1} codewords $\mathbf{u}(s)$ where $s \in S_1 = \{1, \ldots, 2^{nR_1}\}$. Generate each codeword in an i.i.d. manner according to $p_{\mathbf{u}}(\mathbf{u}(s)) = \prod_{i=1}^n p_u(u_i(s))$.

  Let $R_2 = I(y_1; u) - I(y_0; u) + 3\epsilon$. Subdivide $\mathcal{C}$ into 2^{nR_2} subsets or "bins" Accomplish this subdivision of $\mathcal{C}$ by drawing a uniform random variable in $\{1, \ldots, 2^{nR_2}\}$ for each codeword $\mathbf{u}(s) \in \mathcal{C}$. This is the bin to which we assign $\mathbf{u}(s)$. Let $B(m)$ denote the codewords assigned to bin m. There are approximately $2^{n(I(y_0;u)-2\epsilon)}$ codewords in each bin.

- **Encoding:** Given the encoder observation $\mathbf{y}_1$ find the codeword $\mathbf{u}(s) \in \mathcal{C}$ that satisfies $(\mathbf{u}(s), \mathbf{y}_1) \in T_{u,y_1}^n(\epsilon)$. If no such codeword exists an error has occurred. If more than one such codeword exists the encoder selects one at random. Given the selected codeword, $\mathbf{u}(s)$, the encoder transmits the index m of the bin such that $\mathbf{u}(s) \in B(m)$.

- **Decoding:** The decoder looks for a $\mathbf{u}(s) \in B(m)$ such that $(\mathbf{u}(s), \mathbf{y}_0) \in T_{u,y_0}^n(\epsilon)$. If there is a unique such $\mathbf{u}(s)$ the decoder calculates $\hat{\mathbf{x}}$ where $\hat{x}_i = f(u_i(s), y_{0,i})$. If there is no such $\mathbf{u}(s)$, or more than one, than an error has occurred.

- **Probability of error:** Without loss of generality, in calculating the probability of error, we assume the message $m = 1$ is being communicated. We consider four possible errors, and show that each contributes negligibly to the probability of error:

  1. The sequences $(\mathbf{x}, \mathbf{y}_0, \mathbf{y}_1) \notin T_{x,y_0,y_1}^n(\epsilon)$. Since $\mathbf{y}_0$ and $\mathbf{y}_1$ are the outputs of a pair of independent discrete memoryless channels with $\mathbf{x}$ as the inputs, this event has negligible probability of error by the weak law of large numbers.
  2. The observations $\mathbf{y}_1$ typical, but there is no $\mathbf{u}(s) \in \mathcal{C}$ such that $(\mathbf{y}_1, \mathbf{u}(s)) \in T_{y_1,u}^n(\epsilon)$. The probability of this is negligible if $R_1 > I(y_0; u)$, which is true by construction.
  3. The sequences $\mathbf{u}(s)$ and $\mathbf{y}_0$ are not jointly typical. The probability of this event goes to zero as n grows to infinity by the Markov Lemma [24] since $u \leftrightarrow y_1 \leftrightarrow y_0$.

4. There exists another $\tilde{s} \in B(m)$ such that $(\mathbf{u}(\tilde{s}), \mathbf{y}_0) \in T^n_{u,y_0}(\epsilon)$, but $(\mathbf{u}(\tilde{s}), \mathbf{x}) \notin T^n_{u,x}(\epsilon)$. This probability is upper bounded by the size of the bin $|B(m)|$ times the probability that $\mathbf{u}(\tilde{s})$, an independent sequence generated according to $p_{\mathbf{u}}(\mathbf{u}(\tilde{s})) = \prod_{i=1}^n p_u(u_i(\tilde{s}))$ is jointly typical with $\mathbf{y}_0$:
$$\Pr \leq 2^{n(I(y_0;u)-2\epsilon)} 2^{-n(I(y_0;u)-\epsilon)} = 2^{-n\epsilon},$$
which can be made smaller than any target probability of error if n is chosen large enough.
5. Given that $\mathbf{u}(s)$ is recovered correctly, by the Markov Lemma $(\mathbf{x}, \mathbf{u}(s), \mathbf{y}_0) \in T^n_{x,u,y_0}(\epsilon)$ and therefore the empirical joint distribution can be made as close as we want to the chosen distribution $p_{x,u,y_0}(x, u, y_0)$. This implies that $E[D(\mathbf{x}, \hat{\mathbf{x}})] = E[D(\mathbf{x}, f(\mathbf{y}_0, \mathbf{u}(s)))] \leq d$.

A.2 Binary-Hamming Case

In this section we derive $R^{\text{WZ}}_{\text{I}}(d)$ for a discrete symmetric binary source subject to the Hamming distortion measure. The approach is based on that of [84], but modified to take into account the noisy encoder observations. In Section 3.1.1 we gave an informal discussion of these results and plotted the rate-distortion function in Fig. 3.2. It will help to keep both in mind during the following formal exposition. Finally, in that earlier discussion we found that the rate-distortion function $R^{\text{WZ}}_{\text{I}}(d)$ was a convex combination of a function $g(d)$ and the point $(0, p_0)$. In the ensuing discussion we define the function $\bar{g}(d)$ which is basically puts $g(\cdot)$ and the point $(0, p_0)$ together into one function. We will term the lower convex envelope of this new function $\bar{g}(\cdot)$ the function $g^*(d)$ which we will show is equal to the rate-distortion function $R^{\text{WZ}}_{\text{I}}(d)$.

In the binary-Hamming case $\mathbf{x}$ is a sequence of i.i.d. Bernoulli random variables: $\Pr(x_i = 1) = p$ and $\Pr(x_i = 0) = 1 - p$. The variables $\mathbf{y}_0$ and $\mathbf{y}_1$ are observations of $\mathbf{x}$ through independent binary-symmetric channels with cross-over probabilities p_0 and p_1, respectively. This results in posterior distributions $p(y_{0,i} \neq x_i) = p_0$ and $p(y_{1,i} \neq x_i) = p_1$ where we have used $y_{0,i}$ and $y_{1,i}$ to denote the ith samples of the observations $\mathbf{y}_0$ and $\mathbf{y}_1$, respectively. To present the derivation we slightly abuse notation and use $H(p)$ to denote the entropy rate of a Bernoulli random variable x where $\Pr(x = 1) = p$. Using this notation $H(x) = H(p) = -p\log(p) - (1-p)\log(1-p)$. We also use use $*$ to denote binary convolution, i.e., $p * q = p(1-q) + q(1-p)$.

Define the function
$$\bar{g}(\beta) \equiv \begin{cases} H(p_0 * p_1 * \beta) - H(\beta), & 0 \leq \beta < \frac{p_0 - p_1}{1 - 2p_1} \\ 0. & \beta = \frac{p_0 - p_1}{1 - 2p_1} \end{cases} \tag{A.15}$$
The significance of the quantity $(p_0 - p_1)/(1 - 2p_1)$ is that $p_1 * \frac{p_0 - p_1}{1 - 2p_1} = p_0$, which is a Hamming distortion achievable with zero rate by using the side information y_0 as the source estimate. Also define
$$g^*(d) = \inf_{\theta, \beta_a, \beta_b} [\theta \bar{g}(\beta_a) + (1 - \theta)\bar{g}(\beta_b)], \tag{A.16}$$

where the infimum is taken with respect to all $\theta \in [0,1]$ and $\beta_a, \beta_b \in [0,(p_0-p_1)/(1-2p_1)]$ such that $d = \theta[p_1 * \beta_a] + (1-\theta)[p_1 * \beta_b]$. The function $\bar{g}(\beta)$ is shown to be convex for $0 \leq \beta < \frac{p_0-p_1}{1-2p_0}$ in Lemma A of [84]. Because of the convexity of $\bar{g}(\cdot)$ in this range, by Jensen's inequality we know that the infimum in (A.16) cannot be attained by a convex combination of $\bar{g}(\beta_a)$ and $\bar{g}(\beta_b)$ for $0 \leq \beta_a, \beta_b < (p_0-p_1)/(1-2p_1)$. If, however, we set $\beta_b = (p_0-p_1)/(1-2p_1)$, then $\bar{g}(\beta_b) = 0$, is outside the convex region of $\bar{g}(\cdot)$, and so may help achieve the infimum of (A.16). Substitute $\beta_b = (p_0-p_1)/(1-2p_1)$, $\beta = \beta_a$, and (A.15) into (A.16) to simplify the minimization problem,

$$g^*(d) = \inf_{\theta,\beta}\{\theta[H(p_0 * p_1 * \beta) - H(\beta)]\}. \tag{A.17}$$

The infimum is taken with respect to all $\theta \in [0,1]$ and $\beta \in [0,(p_0-p_1)/(1-2p_1)]$ such that $d = \theta p_1 * \beta_a + (1-\theta)p_0$. We now show that $R_{\mathrm{I}}^{\mathrm{WZ}}(d) = g^*(d)$ by showing that $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ is upper and lower bounded by $g^*(d)$.

- **Upper Bound:** $R_{\mathrm{I}}^{\mathrm{WZ}}(d) \leq g^*(d)$

  First, let u be the output of a binary symmetric channel with cross-over probability β, $(0 \leq \beta \leq 0.5)$ when the input is y_1. u and x are related by a cascade of 2 binary symmetric channels with cross-over probabilities β and p_1, which is probabilistically equivalent to a single binary symmetric channel of cross-over probability $p_1 * \beta$. Similarly, y_0 and u are related via a cascade of three binary symmetric channels with cross-over probabilities p_0, p_1, β, yielding an effective cross-over probability $p_0 * p_1 * \beta$. Finally, if we let $\hat{x} = f(y_0, u) = u$, we get $E[D(\hat{x}, x)] = p_1 * \beta$. These observations give us

  $$I(y_0; u) - I(y_1; u) = [1 - H(p_0 * p_1 * \beta)] - [1 - H(\beta)] = H(\beta) - H(p_0 * p_1 * \beta). \tag{A.18}$$

  and so

  $$R_{\mathrm{I}}^{\mathrm{WZ}}(p_1 * \beta) \leq H(\beta) - H(p_0 * p_1 * \beta). \tag{A.19}$$

  Second, let u be degenerate, e.g. $u = 1$ and $H(u) = 0$. Then set $\hat{x} = y_0$ which achieves $E[D(\hat{x}, x)] = p_0$ at zero rate.

  Finally, consider a convex combination of these two scenarios. Let d, $0 \leq d \leq p_0$, be given and $d = \theta[p_1 * \beta] + (1-\theta)p_0$. Then, since $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ is convex,

  $$\begin{aligned} R_{\mathrm{I}}^{\mathrm{WZ}}(d) &= R_{\mathrm{I}}^{\mathrm{WZ}}(d)(\theta[p_1 * \beta] + (1-\theta)p_0) \\ &\leq \theta R_{\mathrm{I}}^{\mathrm{WZ}}(p_1 * \beta) + (1-\theta)R_{\mathrm{I}}^{\mathrm{WZ}}(p_0) \\ &\leq \theta[H(\beta) - H(p_0 * p_1 * \beta)], \end{aligned} \tag{A.20}$$

  where (A.20) holds by substituting in (A.19) and $R_{\mathrm{I}}^{\mathrm{WZ}}(p_0) = 0$. Since these inequalities hold for any $\theta \in [0,1]$ and $\beta \in [0,(p_0-p_1)/(1-2p_1)]$, we have shown $R_{\mathrm{I}}^{\mathrm{WZ}}(d) \leq g^*(d)$.

- **Lower Bound:** $R_{\mathrm{I}}^{\mathrm{WZ}}(d) \geq g^*(d)$

The minimizing distribution of (3.1) must satisfy the following conditions: $y_0 \leftrightarrow x \leftrightarrow y_1 \leftrightarrow u$ and $E[D(\hat{x},x)] < d$. We will show that $R_{\mathrm{I}}^{\mathrm{WZ}}(d) \geq g^*(d)$ by showing that $I(y_1;u) - I(y_0;u) \geq g^*(d)$ for any satisfactory distribution.

Define the sets

$$\mathcal{A} = \{u : f(0,u) = f(1,u)\}, \qquad \mathcal{A}^c = \{u : f(0,u) \neq f(1,u)\}. \tag{A.21}$$

Then we have

$$d \geq E[D(\hat{x},x)] = \Pr(u \in \mathcal{A}) E[D(\hat{x},x)|u \in \mathcal{A}] + \Pr(u \in \mathcal{A}^c) E[D(\hat{x},x)|u \in \mathcal{A}^c]. \tag{A.22}$$

(a) We first show that

$$E[D(\hat{x},x)|u \in \mathcal{A}^c] \geq p_0. \tag{A.23}$$

Rewrite the left-hand side of (A.23) as

$$E[D|u \in \mathcal{A}^c] = \sum_{u \in \mathcal{A}^c} \frac{\Pr(u = u)}{\Pr(u \in \mathcal{A}^c)} E[D|u = u, u \in \mathcal{A}^c]. \tag{A.24}$$

We now lower-bound the last factor of (A.24). In $\mathcal{A}^c$, if $f(0,u) = 0$ then $f(1,u) = 1$. Therefore,

$$\begin{aligned} E[D|u = u, u \in \mathcal{A}^c] &= \Pr(x = 1, y_0 = 0|u = u) + \Pr(x = 0, y_0 = 1|u = u) \\ &= \Pr(y_0 = 0|x = 1)\Pr(x = 1|u = u) \\ &\quad + \Pr(y_0 = 1|x = 0)\Pr(x = 0|u = u) \qquad \text{(A.25)} \\ &= p_0[\Pr(x = 1|u = u) + \Pr(x = 0|u = u)] = p_0 \qquad \text{(A.26)} \end{aligned}$$

where (A.25) follows from the Markov relationship $y_0 \leftrightarrow x \leftrightarrow u$, and we have dropped the explicit conditioning on $u \in \mathcal{A}^c$ in the right-hand expressions. Substituting (A.26) into (A.24) and summing shows that (A.23) holds. If, on the other hand, we had chosen $f(0,u) = 1$ for $u \in \mathcal{A}^c$ then we would have derived $E[D|u = u, u \in \mathcal{A}^c] = 1 - p_0 \geq p_0$, and so (A.23) would again hold.

(b) Now, we focus on $E[D|u \in \mathcal{A}]$. First write

$$E[D|u \in \mathcal{A}] = \sum_{u \in \mathcal{A}} \frac{\Pr(u = u)}{\Pr(u \in \mathcal{A})} E[D|u = u, u \in \mathcal{A}]. \tag{A.27}$$

Substituting (A.27) and (A.23) into (A.22) gives us

$$\begin{aligned} d &\geq \Pr(u \in \mathcal{A}) \sum_{u \in \mathcal{A}} \frac{\Pr(u = u)}{\Pr(u \in \mathcal{A})} E[D|u = u, u \in \mathcal{A}] + \Pr(u \in \mathcal{A}^c) p_0 \qquad \text{(A.28)} \\ &= \theta \sum_{u \in \mathcal{A}} \lambda_u d_u + (1 - \theta) p_0 \equiv d', \qquad \text{(A.29)} \end{aligned}$$

where $\theta = \Pr(u \in \mathcal{A})$, $\lambda_u = \Pr(u = u)/\Pr(u \in \mathcal{A})$, and $d_u = E[D|u = u, u \in \mathcal{A}]$. Now,

$$\begin{aligned} I(y_1; u) - I(y_0; u) &= H(y_0|u) - H(y_1|u) \\ &\geq \sum_{u\in\mathcal{A}} [H(y_0|u = u) - H(y_1|u = u)] \Pr(u = u) \\ &= \theta \sum_{u\in\mathcal{A}} \lambda_u [H(y_0|u = u) - H(y_1|u = u)]. \qquad \text{(A.30)} \end{aligned}$$

Since, for $u \in \mathcal{A}$, $f(0,u) = f(1,u)$, call this $f(u)$. Then

$$d_u = E[D|u = u, u \in \mathcal{A}] = \Pr(x \neq f(u)|u = u, u \in \mathcal{A}). \qquad \text{(A.31)}$$

Since $f(u)$ is a deterministic function of u, this tells us that

$$H(x|u = u) = H(x \neq f(u)|u = u) = H(d_u), \qquad \text{(A.32)}$$

and, since x and y_0 are related via a binary symmetric channel with cross-over probability p_0, we have

$$H(y_0|u = u) = H(p_0 * d_u). \qquad \text{(A.33)}$$

We now use (A.31) to derive an expression for $H(y_1|u = u)$.

$$\begin{aligned} d_u &= E[d|u = u, u \in \mathcal{A}] = \Pr(x \neq f(u)|u = u) \\ &= \Pr(x \neq y_1, y_1 = f(u)|u = u) + \Pr(x = y_1, y_1 \neq f(u)|u = u) \\ &= \Pr(x \neq y_1|y_1 = f(u), u = u) \Pr(y_1 = f(u)|u = u) \\ &\quad + \Pr(x = y_1|y_1 \neq f(u), u = u) \Pr(y_1 \neq f(u)|u = u) \\ &= \Pr(x \neq y_1)(1 - \Pr(y_1 \neq f(u)|u = u)) + (1 - \Pr(x \neq y_1)) \Pr(y_1 \neq f(u)|u = u) \\ &= p_0 * \Pr(y_1 \neq f(u)|u = u), \\ &= p_0 * \beta_u, \qquad \text{(A.34)} \end{aligned}$$

where $\beta_u \equiv \Pr(y_1 \neq f(u)|u = u)$. Using the definition of $*$ we can solve for β_u,

$$\beta_u = \Pr(y_1 \neq f(u)|u = u) = \frac{d_u - p_0}{1 - 2p_0}.$$

And, from similar arguments as led to (A.33) we get

$$H(y_1|u = u) = H(\beta_u). \qquad \text{(A.36)}$$

Substituting (A.33), (A.34) and (A.36) into (A.30) gives us

$$\begin{aligned} I(y_1; u) - I(y_0; u) &\geq \theta \sum_{u\in\mathcal{A}} \lambda_u [H(p_o * p_1 * \beta_u) - H(\beta_u)] \\ &\geq \theta[H(\sum_{u\in\mathcal{A}} \lambda_u [p_0 * p_1 * \beta_u]) - H(\sum_{u\in\mathcal{A}} \lambda_u \beta_u)] \qquad \text{(A.37)} \\ &= \theta[H(p_0 * p_1 * \beta) - H(\beta)], \qquad \text{(A.38)} \end{aligned}$$

where (A.37) follows from the convexity of $\bar{g}(\cdot)$, and (A.38) from $\beta = \sum_{u\in\mathcal{A}} \lambda_u \beta_u$ and the linearity of convolution.

So, we have shown that for any distribution on $x, y_0, y_1, u, \hat{x} = f(y_0, u)$ satisfying $E\left[D(\hat{x}, x)\right] < d$ and $y_0 \leftrightarrow x \leftrightarrow y_1 \leftrightarrow u$, there exists $\theta \in [0,1]$ and $0 \leq \beta \leq (p_0 - p_1)/(1-2p_1)$ such that

$$1.\quad \theta[p_0 * \beta] + (1-\theta)p_0 = d' \leq d. \tag{A.39}$$

$$2.\quad I(y_1; u) - I(y_0; u) \geq g^*(d') \geq g^*(d), \tag{A.40}$$

where the final inequality holds from (A.29) and because $g^*(d)$ is non increasing in d. This means that the minimization problem in (A.1) is lower-bounded by $g^*(d)$, and so $R_{\mathrm{I}}^{\mathrm{WZ}}(d) \geq g^*(d)$.

A.3 Quadratic-Gaussian Case

To extend the results of Theorem 4 to continuous alphabets, we must partition $\mathcal{Y}_1^n \times \mathcal{X}^n \times \mathcal{Y}_0^2$ so as to preserve the Markov relationship $y_1 \leftrightarrow x \leftrightarrow y_0$ for each block length n. See [83, 45] for more details. Given this extension of Theorem 4 to continuous alphabets, we now derive the test channel that achieves the rate-distortion function for the quadratic-Gaussian case. In (3.5) we stated that the rate-distortion function for this source-distortion pair is

$$R_{\mathrm{I}}^{\mathrm{WZ}}(d) = \frac{1}{2}\log\left[\frac{\sigma^2_{x|y_0} - \sigma^2_{x|y_0,y_1}}{d - \sigma^2_{x|y_0,y_1}}\right].$$

- **Upper Bound:** $R_{\mathrm{I}}^{\mathrm{WZ}}(d) \leq R^*(d)$

  The sequences $\mathbf{y}_0$ and $\mathbf{y}_1$ are observations of $\mathbf{x}$ through i.i.d. additive Gaussian noise channels: $y_{0,i} = x_i + v_{0,i}$, $y_{1,i} = x_i + v_{1,i}$, where $v_{j,i} \sim N(0, N_j)$ and the two noise sources are independent of each other and of the source. Because the Markov condition $p(x, y_0, y_1, u) = p(x)p(y_0|x)p(y_1|x)p(u|y_1)$ implies $p(y_0, y_1, u) = p(y_0|y_1)p(y_1)p(u|y_1)$, the rate-distortion function (A.1) can be rewritten as

  $$R_{\mathrm{I}}^{\mathrm{WZ}}(d) = \min_{p(u|y_1)} \min_{f} I(y_1; u|y_0). \tag{A.42}$$

  We first find an upper-bound $R^*(d)$ on $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$ and later show that $R^*(d)$ is also a lower bound and therefore equal to $R_{\mathrm{I}}^{\mathrm{WZ}}(d)$.

  Define the auxiliary random variable $u = \alpha y_1 + e$ where $e \sim N(0, \alpha d^*)$ is independent of y_1, y_0. For this choice of u, define $R^*(d)$ as

  $$R^*(d) = I(y_1; u|y_0) = \frac{1}{2}\log\left[1 + \frac{\alpha}{d^*}(\sigma^2_{x|y_0} + N_1)\right], \tag{A.43}$$

where $\sigma^2_{x|y_0}$ is the minimum mean-squared estimation error of x given y_0. The minimum mean-squared estimation error for x given y_0 and u is

$$\sigma^2_{x|y_0,u} = \frac{\sigma^2_{x|y_0}}{1 + \frac{\alpha\sigma^2_{x|y_0}}{\alpha N_1 + d^*}}. \tag{A.44}$$

Set (A.44) equal to d, the target distortion and solve for α/d^*:

$$\frac{\alpha}{d^*} = \frac{\sigma^2_{x|y_0} - d}{d(\sigma^2_{x|y_0} + N_1) - \sigma^2_{x|y_0}N_1} = \frac{\sigma^2_{x|y_0} - d}{(d - \sigma^2_{x|y_0,y_1})(\sigma^2_{x|y_0} + N_1)}, \tag{A.45}$$

where we have used the relation $\sigma^2_{x|y_0,y_1} = \sigma^2_{x|y_0}N_1/(\sigma^2_{x|y_0} + N_1)$. Substitute (A.45) into (A.43) to get

$$R^*(d) = \frac{1}{2}\log\left[\frac{\sigma^2_{x|y_0} - \sigma^2_{x|y_1,y_0}}{d - \sigma^2_{x|y_1,y_0}}\right], \tag{A.46}$$

where $\sigma^2_{x|y_1,y_0}$ is the minimum mean-squared estimation error of x given y_1 and y_0, and $\sigma^2_{x|y_1,y_0} \leq d \leq \sigma^2_{x|y_0}$. The span of d is lower bounded by the estimation error in x given both observations (y_1 and y_0), and is upper bounded by the estimation error in x given y_0, and ignoring y_1 altogether.

If we set $d^* = d - \sigma^2_{x|y_1,y_0}$ and $\alpha = (\sigma^2_{x|y_0} - d)/(\sigma^2_{x|y_0} + N_1)$, we can show that the minimum mean-squared estimation estimator of x given y_0 and u is

$$\hat{x} = f(y_0, u) = E\left[x|y_0, u\right] = \frac{d}{N_0}y_0 + \left[1 + \frac{N_1}{\sigma^2_{x|y_0}}\right]u. \tag{A.47}$$

From (A.44) we know that $E\left[(x - f(y_0, u))^2\right] = d$, so this function satisfies the target distortion.

If we explore the statistical relationship between x and $\hat{x}$, it looks like a standard rate-distortion achieving test channel for a Gaussian source. In other words, $\hat{x} \sim N(0, \sigma^2_x - d)$ and $x = \hat{x} + \tilde{e}$ where $\tilde{e} \sim \mathcal{N}(0, d)$ is independent of $\hat{x}$.

- **Lower Bound:** $R^{\mathrm{WZ}}_{\mathrm{I}}(d) \geq R^*(d)$

  We now show that $R^*(d) \leq R^{\mathrm{WZ}}_{\mathrm{I}}(d)$. We do this by deriving the rate-distortion function for the less general case when y_0 is observed both at the encoder and at the decoder. Call the rate-distortion function for this problem $\bar{R}(d)$. Clearly $\bar{R}(d) \leq R^{\mathrm{WZ}}_{\mathrm{I}}(d)$ because of the extra information at the encoder. We show that $R^*(d) = \bar{R}(d)$.

  Define $\tilde{x} = x - E\left[x|y_0\right]$ and $\tilde{y}_1 = y_1 - E\left[x|y_0\right] = \tilde{x} + v_1$. Because $\tilde{x}$ is the minimum mean-squared estimation error of x from y_0 then by the orthogonality principal $\tilde{x}$ and y_0 are independent. Furthermore, v_1 is independent of y_0 by definition. By these independence relationships, and because x, y_1 and y_0 are jointly Gaussian, $p(\tilde{x}, \tilde{y}_1 | y_0) = p(\tilde{x}, \tilde{y}_1)$. Therefore $p(\tilde{x}|\tilde{y}_1, y_0) = p(\tilde{x}|\tilde{y}_1)$ and we can ignore y_0 when estimating $\tilde{x}$. The problem is thereby reduced to source coding a source in additive white Gaussian noise for which the rate-distortion region is known from [79] to be

$$\bar{R}(d) = \frac{1}{2}\log\left[\frac{\sigma^2_{\tilde{x}} - \sigma^2_{\tilde{x}|\tilde{y}_1}}{d - \sigma^2_{\tilde{x}|\tilde{y}_1}}\right] = \frac{1}{2}\log\left[\frac{\sigma^2_{x|y_0} - \sigma^2_{x|y_0,y_1}}{d - \sigma^2_{x|y_0,y_1}}\right].$$

Thus, we have $R^*(d) = \bar{R}(d) \leq R^{\mathrm{WZ}}_{\mathrm{I}}(d) \leq R^*(d)$ which implies that $R^*(d) = R^{\mathrm{WZ}}_{\mathrm{I}}(d)$.

Appendix B

Derivations: Noisy Information Embedding

B.1 Finite-Alphabet Capacity Expression

In this appendix we prove a single-letter expression for the information embedding capacity function $C_{\mathrm{I}}^{\mathrm{IE}}(d)$ when the encoder has noisy source observations. The rate-distortion function $C_{\mathrm{I}}^{\mathrm{IE}}(d)$ is a tight upper bound on the rate that we can reliable communicate at while guaranteeing that $\frac{1}{n}E\left[\sum_{i=1}^{n} D(x_i, w_i)\right]$ can be made arbitrarily close to d for a sufficiently long block length n. Formally, repeating the statement of Thm. 5 from Chapter 3, we show the following.

Theorem 9 *Let a random pair of sources* $(\mathbf{x}, \mathbf{y})$, *a distortion measure* $D(\cdot,\cdot)$, *and a memoryless channel law* $p(z|w)$ *be given such that*

(a) $p_{\mathbf{x},\mathbf{y}}(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{n} p_{x,y}(x_i, y_i)$,

(b) $D(\mathbf{x}, \mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} D(x_i, w_i)$,

where $\mathbf{w} = \mathbf{x} + \mathbf{e}$ *is the channel input and* $\mathbf{e}$ *is the embedding signal. Then, a sequence of length-n encoder-decoder pairs can be designed such that a message of rate R can be communicated to the decoder with an arbitrarily small probability of decoding error as n grows to infinity while satisfying the average distortion constraint d if and only if*

$$R < C_{\mathrm{I}}^{\mathrm{IE}}(d) = \max_{p_{u|y}(u|y)\in\mathcal{U}} [I(z; u) - I(u; y)] \tag{B.1}$$

where the set $\mathcal{U}$ *consists of all posteriors* $p(u|y)$ *relating the auxiliary random variable* u *to the host information* y *that satisfy the two conditions:*

(i) $p(u|x, y, e, w, z) = p(u|y)$.

(ii) $E[D(x, w)] \leq d$ *where* $w = x + e$ *and* $e = f(u, y)$ *for some memoryless function* $f : \mathcal{U} \times \mathcal{Y} \to \mathcal{E}$.

Just as in [31], because $C_{\mathrm{I}}^{\mathrm{IE}}$ is convex in the distribution $p_{e|u,x}$, then the distribution is deterministic, simplifying (B.1). This is the reason why the maximum can be taken over all distributions $p_{u|y}(u|y)$ and functions $f : \mathcal{U} \times \mathcal{Y} \to \mathcal{E}$, where $e = f(u, y)$.

B.1.1 Converse

In this section we show that $C_{\mathrm{I}}^{\mathrm{IE}}(d)$ is an upper bound on the achievable communication rate for the noisy information embedding problem.

$$nR = H(m) = I(m; z^n) + H(m|z^n) \tag{B.2}$$

$$= I(m; z^n) - I(m; y^n) + H(m|z^n) \tag{B.3}$$

$$\leq \sum_{i=1}^{n} [I(u(i); z_i) - I(u(i); y_i)] + H(m|z^n) \tag{B.4}$$

$$\leq n \max_i [I(u(i); z_i) - I(u(i); y_i)] + H(m|z^n) \tag{B.5}$$

$$= n[I(u; z) - I(u; y)] + H(m|z^n) \tag{B.6}$$

$$\leq nC_{\mathrm{I}}^{\mathrm{IE}}(d) + P_e^{(n)} nR + 1. \tag{B.7}$$

Line	Justification
(B.2)	m distributed uniformly in $\{1, \ldots, 2^{nR}\}$.
(B.3)	$I(m; y^n) = 0$ by independence of m and y^n.
(B.4)	By [31] Lemma 4, where $u_i \triangleq (m, z^{i-1}, y_{i+1}^n)$.
(B.6)	$u = u(i)$ such that i corresponds to the maximum term of (B.5).
(B.7)	Fano inequality and definition of $C_{\mathrm{I}}^{\mathrm{IE}}(d)$.

Rearranging terms in (B.7) we have

$$P_e^{(n)} \geq 1 - \frac{C_{\mathrm{I}}^{\mathrm{IE}}(d)}{R} - \frac{1}{nR} \tag{B.8}$$

which shows for $R > C_{\mathrm{I}}^{\mathrm{IE}}(d)$, the probability of error is bounded away from 0.

B.1.2 Achievability

We now show that we can find a channel code that can achieve $E[D(\mathbf{x}, f(\mathbf{u}(s), \mathbf{y}))] \leq d$ while operating at a rate C arbitrarily close to $C_{\mathrm{I}}^{\mathrm{IE}}(d)$. We construct a capacity achieving code as follows:

- **Codebook Generation:** Let $R_1 = I(u; z) - 2\epsilon$. Generate a random codebook $\mathcal{C}$ consisting of 2^{nR_1} $\mathbf{u}(s)$ where $s \in S_1 = \{1, \ldots, 2^{nR_1}\}$. Generate each codeword in an i.i.d. manner according to $p_{\mathbf{u}}(\mathbf{u}(s)) = \prod_{i=1}^{n} p_u(u_i(s))$.

  Let $R_2 = I(u; y) + 3\epsilon$. Subdivide $\mathcal{C}$ into 2^{nR_2} subcodes $\mathcal{C}_j$, indexed by $j \in \{1, \ldots, 2^{nR_2}\}$. Accomplish this subdivision of $\mathcal{C}$ by drawing a uniform random variable in $\{1, \ldots, 2^{nR_2}\}$ for each codeword $\mathbf{u}(s) \in \mathcal{C}$, where $s \in S_1$. This is the subcode to which we assign $\mathbf{u}(s)$. Each subcode has approximately $2^{n(R_1 - R_2)} = 2^{n(I(u;z) - I(u;y) - \epsilon)} = 2^{n(C_{\mathrm{I}}^{\mathrm{IE}}(d) - \epsilon)}$ codewords in its codebook.

- **Encoding:** The message $m = m$ specifies the subcode $\mathcal{C}_m$ that we use. For a given observation sequence y^n, the encoder looks for a codeword $\mathbf{u}(s) \in \mathcal{C}_m$ that satisfies $(\mathbf{u}(s), \mathbf{y}) \in T^n_{u,y}(\epsilon)$. If no such codeword exists an error has occurred and the encoder chooses a codeword from $\mathcal{C}_m$ at random. If more than one such codeword exists, the encoder can pick any one of them. Given the selected codeword, the embedding signal $\mathbf{e}$ is calculated in a sample-by-sample manner according to $e_i = f(u(s)_i, y_i)$.

- **Decoding:** The decoder looks for a $\mathbf{u}(s) \in \mathcal{C}$ such that $(\mathbf{u}(s), \mathbf{z}) \in T^n_{u,z}(\epsilon)$. If there is a unique $\mathbf{u}(s)$ then the estimated information sequence is $\hat{m} = j$ where $j \in S_2$ is the index of the codebook that contains $\mathbf{u}(s)$. If there is no $\mathbf{u}(s)$ that satisfies joint typicality, or there is more than one, an error has occurred and the decoder assigns the index $\hat{m} = 0$.

- **Probability of error:** Without loss of generality, in calculating the probability of error, we assume the message $m = 1$ is being communicated. We consider four possible errors, and show that each contributes negligible to the probability of error:

  1. The pair $(\mathbf{x}, \mathbf{y}) \notin T^n_{x,y}(\epsilon)$. Since $\mathbf{x}$ and $\mathbf{y}$ are respectively the inputs and outputs of a discrete memoryless channel this event has negligible probability of error by the weak law of large numbers.
  2. The observation $\mathbf{y}$ typical, but there is no $\mathbf{u}(s) \in \mathcal{C}_1$ such that $(\mathbf{y}, \mathbf{u}(s)) \in T^n_{y,u}(\epsilon)$. The probability of this is negligible if $R_2 > I(u; y)$, which is true by construction, $R_2 = I(u; y) + 2\epsilon$.
  3. The pairs $(\mathbf{x}, \mathbf{y})$ and $(\mathbf{u}(s), \mathbf{y})$ are jointly typical, but $(\mathbf{u}(s), \mathbf{z})$ is not. First, we observe that $(\mathbf{x}, \mathbf{y}, \mathbf{u}(s)) \in T^n_{x,y,u}(\epsilon)$ by the Markov Lemma [24] since $x_i \leftrightarrow y_i \leftrightarrow u_i$. Second, since $e_i = f(u_i(s), y_i)$, and $(\mathbf{x}, \mathbf{y}, \mathbf{u}(s)) \in T^n_{x,y,u}(\epsilon)$ then $(\mathbf{x}, \mathbf{y}, \mathbf{u}(s), \mathbf{e}) \in T^n_{x,y,u,e}(\epsilon)$. Third, since $\mathbf{z}$ is generated i.i.d. according to $p_{z|x,e}(z_i|x_i, e_i)$, $(\mathbf{x}, \mathbf{y}, \mathbf{u}(s), \mathbf{w}, \mathbf{z}) \in T^n_{x,y,u,w,z}(\epsilon)$. This implies that $(\mathbf{u}(s), \mathbf{z}) \in T^n_{u,z}(\epsilon)$ by the definition of joint typicality.
  4. There exists a $\mathbf{u}(\tilde{s} \in \mathcal{C}$, $\mathbf{u}(\tilde{s}) \neq \mathbf{u}(s)$ such that $(\mathbf{u}(\tilde{s}), \mathbf{z}) \in T^n_{u,z}(\epsilon)$. This probability is upper bounded by the size of the codebook $|\mathcal{C}|$ times the probability that $\mathbf{u}(\tilde{s})$, an independent codeword generated according to $p_{\mathbf{u}}(\mathbf{u}(\tilde{s})) = \prod_{i=1}^n p_u(u_i(\tilde{s}))$, is jointly typical with $\mathbf{z}$:

$$\Pr \leq 2^{n(I(u;z)-2\epsilon)} 2^{-n(I(u;z)-\epsilon)} = 2^{-n\epsilon}.$$

B.2 Quadratic-Gaussian Case

In this section we first develop the test channel that achieves the rate of (3.8) and then show that we can do no better.

- **Lower Bound:** $C_{\mathrm{I}}^{\mathrm{IE}}(d) \geq C^*(d)$

  The host $\mathbf{x}$ is an i.i.d. zero-mean Gaussian source with variance σ_x^2. The observation $\mathbf{y}$ is the host corrupted by i.i.d. additive white Gaussian noise $y_i = x_i + v_{0,i}$ where $v_0 \sim N(0, N_0)$. The communication channel is an additive white Gaussian noise channel $z_i = x_i + e_i + v_{1,i}$ where $v_1 \sim N(0, N_1)$. We define $u = e + \alpha y$ where $E\left[e^2\right] = d$, and $y = x + v_0$. The two terms in the capacity expression simplify as follows

$$
\begin{aligned}
I(u; z) &= h(z) - h(z|u) \\
&= h(e + x + v_1) + h(e + \alpha(x + v_0)) - h(e + x + v_1, e + \alpha(x + v_0)) \\
&= 0.5\log[(2\pi\epsilon)d + \sigma_x^2 + N_1] + 0.5\log[(2\pi\epsilon)d + \alpha^2(\sigma_x^2 + N_0)] \\
&\qquad -0.5\log[(2\pi\epsilon)^2(d + \sigma_x^2 + N_1)(d + \alpha^2(\sigma_x^2 + N_0)) - (d + \alpha\sigma_x^2)^2] \\
&= \frac{1}{2}\log\left[\frac{(d + \sigma_x^2 + N_1)(d + \alpha^2(\sigma_x^2 + N_0))}{(d + \sigma_x^2 + N_1)(d + \alpha^2(\sigma_x^2 + N_0)) - (d + \alpha\sigma_x^2)^2}\right]. \qquad \text{(B.10)} \\
I(u; y) &= h(u) - h(u|y) \\
&= h(e + \alpha(x + v_1)) - h(e + \alpha(x + v_0)|x + v_0) \\
&= \frac{1}{2}\log\left[\frac{d + \alpha^2(\sigma_x^2 + N_0)}{d}\right]. \qquad \text{(B.11)}
\end{aligned}
$$

Subtract (B.11) from (B.10) to get

$$
I(u; z) - I(u; y) = \frac{1}{2}\log\left[\frac{d(d + \sigma_x^2 + N_1)}{(d + \sigma_x^2 + N_1)(d + \alpha^2(\sigma_x^2 + N_0)) - (d + \alpha\sigma_x^2)^2}\right]. \qquad \text{(B.12)}
$$

Differentiate (B.12) with respect to α, and set equal to zero to get

$$
\alpha = \frac{d}{(d + N_0 + N_1) + \frac{N_0}{\sigma_x^2}(d + N_1)}. \qquad \text{(B.13)}
$$

Substitute (B.13) into (B.12) to get

$$
C^*(d) = I(u; z) - I(u; y) = \frac{1}{2}\log\left[1 + \frac{d}{\sigma_{x|y}^2 + N_1}\right]. \qquad \text{(B.14)}
$$

The above derivation, leading to (B.14) is analogous to the approach taken in [20]. To see that the resultant achievable rate is the channel capacity we could use Theorem 2b of [42] which tells us the channel capacity when $\mathbf{y}$ is known at both encoder and decoder. An alternate method allows us to leverage the work in [20] where a converse is already proven.

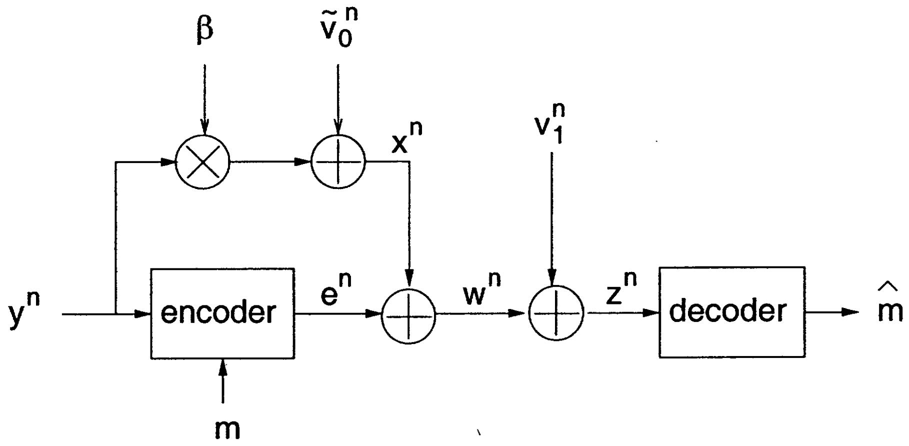


Figure B.1. Channel coding with noisy side information, Gaussian case. The signal y is viewed as the perfectly known channel state. The random quantities m, $\mathbf{e}$, $\mathbf{w}$, and $\mathbf{z}$ are respectively the message, embedding signal, composite signal, and channel output. The two unknown sources of noise are $\tilde{\mathbf{v}}_0$ and $\mathbf{v}_1$ which are the estimation error and channel noise, respectively.

- **Upper Bound:** $C_{\mathrm{I}}^{\mathrm{IE}}(d) \leq C^*(d)$

  So far in this section we have viewed $\mathbf{x}$ as the host and $\mathbf{y} = \mathbf{x} + \mathbf{v}_0$ as the observations. Now we reverse that point of view and consider $\mathbf{y}$ as the host and $\mathbf{x}$ as generated from $\mathbf{y}$ according to

  $$x_i = \beta y_i + \tilde{v}_{0,i}. \tag{B.15}$$

  If we let $\beta = \frac{\sigma_x^2}{\sigma_x^2 + N_0}$ in (B.15) and let $\tilde{v}_{0,i} \sim N\left(0, \frac{\sigma_x^2 N_0}{\sigma_x^2 + N_0}\right)$, the joint distribution $p_{x,y}(x,y)$ is the same as before, but our point of view is reversed, with x_i described as a function of y_i rather than the other way round. Other than the known multiplier β, our problem is now identical to the one considered in [20]. We can use those results to determine the channel capacity.

  With these changes the channel output is

  $$\begin{aligned} z_i &= w_i + \beta y_i + \tilde{v}_{0,i} + v_{1,i}, & \text{(B.16)} \\ &= w_i + \left(\frac{\sigma_x^2}{\sigma_x^2 + N_0}\right) y_i + \tilde{v}_{0,i} + v_{1,i}. & \text{(B.17)} \end{aligned}$$

  This scenario is illustrated in Fig. B.1.

  Since $\mathbf{y}$ is known perfectly at the encoder and the triple of random variables $(y_i, \tilde{v}_{0,i}, v_{1,i})$ are independent for all i, we use results of [20] to state the channel capacity as

  $$C^*(d) = \frac{1}{2}\log\left[1 + \frac{d}{\sigma_{\tilde{v}_0}^2 + \sigma_{v_1}^2}\right] = \frac{1}{2}\log\left[1 + \frac{d}{\sigma_{x|y}^2 + N_1}\right], \tag{B.18}$$

which is identical to (B.14).

Appendix C

The Serial Markov Lemma

In this appendix we discuss the Markov Lemma and extensions. This lemma, introduced by Berger [8], is instrumental in a number of network information theory problems. We have already used when extending Wyner-Ziv coding and information embedding to noisy encoder observations, as discussed in appendices A and B. The basic idea of the Markov Lemma is that under certain conditions joint typicality is transitive. That is, if $(\mathbf{x}, \mathbf{y})$ are jointly typical and $(\mathbf{y}, \mathbf{z})$ are jointly typical then, under the special conditions we discuss in Section C.1, $(\mathbf{x}, \mathbf{z})$ will also be jointly typical. In this appendix we generalize Berger's result to a serial form of the lemma that we need to show achievability in the serial CEO problem as discussed in Appendix D. We will also use a form of the Markov Lemma in proving the parallel CEO results, but this requires a different extension of the lemma that has already been accomplished in [40, 45].

In Section C.1 we introduce Berger's Markov lemma and discuss our extension. In Section C.2 we introduce a form of strong typicality that we will later need which we term $\epsilon(n)$-strong typicality. Then in Sections C.3 and C.4 we prove the Serial Markov Lemma in two different ways.

■ C.1 Introduction

Generally joint typicality is non-transitive. This means that the joint typicality of $(\mathbf{x}, \mathbf{y})$ and of $(\mathbf{y}, \mathbf{z})$ does not imply that $(\mathbf{x}, \mathbf{z})$ are jointly typical. This is the case even if joint typicality is defined by the Markov relationship $p(x, y, z) = p(x)p(y|x)p(z|y)$. We demonstrate this through Berger's canonical example [8].

Example: Joint Typicality is Non-Transitive. Consider the joint distribution $p(x, y, z) = p(x)p(y)p(z)$ where x, y, and z are i.i.d. equi-probable binary random variables. Let $\underline{0}$ and $\underline{1}$ denote m-length strings of all zeros and all ones, respectively. Define $\mathbf{x} = \underline{0}\ \underline{0}\ \underline{1}\ \underline{1}$, $\mathbf{y} = \underline{0}\ \underline{1}\ \underline{0}\ \underline{1}$, and $\mathbf{z} = \mathbf{x}$. Then, $(\mathbf{x}, \mathbf{y}) \in T^n_{p_{\mathsf{x},\mathsf{y}}}(\epsilon) = T^n_{p_\mathsf{x} p_\mathsf{y}}(\epsilon)$ and $(\mathbf{y}, \mathbf{z}) \in T^n_{p_{\mathsf{y},\mathsf{z}}}(\epsilon) = T^n_{p_\mathsf{y} p_\mathsf{z}}(\epsilon)$, but $(\mathbf{x}, \mathbf{z}) \notin T^n_{p_{\mathsf{x},\mathsf{z}}}(\epsilon) = T^n_{p_\mathsf{x} p_\mathsf{z}}(\epsilon)$ since they fail to disagree in roughly half the places which would be necessary for them to be jointly strongly typical. □

The Markov Lemma [8] tells us that if $(\mathbf{x}, \mathbf{y})$ are jointly typical, and if $\mathbf{z}$ is generated from $\mathbf{y}$ in a conditionally pairwise i.i.d. manner, $p(\mathbf{z}|\mathbf{y}) = \prod_{i=1}^{n} p(z_i|y_i)$, then $(\mathbf{x}, \mathbf{z})$ will be jointly typical. The difference with the example of the preceding paragraph is that for the Markov Lemma $(\mathbf{y}, \mathbf{z})$ are not only jointly typical, but they are generated in a *pairwise i.i.d.* manner. This guarantees a particularly simple (memoryless) relationship between $\mathbf{y}$ and $\mathbf{z}$. The Markov Lemma can be used to prove a number of network information theory results such as, e.g., Wyner-Ziv source coding with side information. For that problem, in the current notation, $\mathbf{x}$ would be the codeword, $\mathbf{y}$ the source, and $\mathbf{z}$ the side information which is generated in a pairwise i.i.d. manner with the source.

In this appendix we show that if $(\mathbf{x}, \mathbf{y}) \in T^n_{\mathsf{x},\mathsf{y}}(\epsilon)$ and if $\mathbf{y}$ looks marginally i.i.d. then if we use a suitably designed code to transcode $\mathbf{y}$ into $\mathbf{z}$, we can guarantee that the probability that $(\mathbf{x}, \mathbf{z}) \notin T^n_{\mathsf{x},\mathsf{z}}(\epsilon)$ goes to zero as n grows to infinity. This is an extension of the Markov Lemma to a sucession of coding steps: if $(\mathbf{x}, \mathbf{y})$ are jointly typical and $(\mathbf{y}, \mathbf{z})$ are jointly typical, *and* $\mathbf{z}$ is an encoding of $\mathbf{y}$ using a particular class of codes, then $\mathbf{x}$ and $\mathbf{z}$ are jointly typical. We term this extension the Serial Markov Lemma.

We consider two classes of codes. The first, proposed in [18], use a dithered encoding rule that results in $(\mathbf{y}, \mathbf{z})$ that cannot be differentiated from a pair $(\mathbf{y}, \tilde{\mathbf{z}})$ generated in a pairwise i.i.d. manner. The second class of codes were introduced by Viswanathan and Berger [75], and have useful stationary properties. This class of codes is less powerful than those produced by the dithered encoding rule, but seems to give us the minimal needed structure to show the Serial Markov Lemma.

C.2 $\epsilon(n)$-Strong Typicality

In Section A.1.3 we defined ϵ-strongly typical sequence. In this section we slightly broaden this set to $\epsilon(n)$-strong typicality. The set of length-n vectors $\mathbf{x}$ that are $\epsilon(n)$-strongly typical according to a finite-alphabet probability measure $p_{\mathsf{x}}(x)$ is defined as $T^n_{\mathsf{x}}(\epsilon(n))$ where

$$T^n_{\mathsf{x}}(\epsilon(n)) = \{\mathbf{x}\} : \begin{cases} |N(x_0;\mathbf{x}) - np_{\mathsf{x}}(x_0)| < n\epsilon(n)|\mathcal{X}|^{-1} & \text{for all } x_0 \in \mathcal{X} \text{ s.t. } p_{\mathsf{x}}(x_0) > 0 \\ N(x_0;\mathbf{x}) = 0 & \text{if } p_{\mathsf{x}}(x_0) = 0. \end{cases}$$

In our earlier definition of typicality $\epsilon(n)$ equalled the constant ϵ. In this section we show that we can make the $\epsilon(n)$ a monotonically decreasing function of n such that, as long as the decrease is not too fast, the resultant set will retain all the classic properties of the strongly typical set where $\epsilon(n)$ is a constant. We will need this variation on the regular definition to show the Serial Markov Lemma.

Lemma 1 *For all* $\epsilon(n)$ *such that* $1/\epsilon(n)^2 \in o(n)$[1]*, if* $\mathbf{x}$ *is an i.i.d. random vector such that* $p(\mathbf{x}) = \prod_{i=1}^n p(x_i)$*, then* $\Pr[T^n_{\mathsf{x}}(\epsilon(n))] \to 1$ *as* $n \to \infty$.

Proof:
We express the random variable $N(x_0;\mathbf{x})$ as the sum of the i.i.d. binary indicator random

---

[1] The order notation "little" $o(n)$ means that if $f(n) \in o(n)$ then $\lim_{n\to\infty} \frac{f(n)}{n} = 0$.

variables I_k, $1 \leq k \leq n$ where

$$I_k = \begin{cases} 0 & \text{if } x_k \neq x_0 \\ 1 & \text{if } x_k = x_0 \end{cases}$$

Convergence in the mean is shown as follows:

$$E\left[N(x_0;\mathbf{x})\right] = E\left[\sum_{k=1}^{n} I_k\right] = \sum_{k=1}^{n} E\left[I_k\right] = \sum_{k=1}^{n} p(x_0) = np(x_0).$$

The variance grows as:

$$\begin{aligned}\operatorname{var}(N(x_0;\mathbf{x})) =& E\left[\left(\sum_{k=1}^{n} I_k - np(x_0)\right)^2\right] = E\left[\left(\sum_{k=1}^{n} I_k\right)^2\right] - n^2p(x_0) \\ =& \sum_{k=1}^{n} E\left[I_k^2\right] + \sum_{k\neq l} E\left[I_kI_l\right] - n^2p(x_0) = \sum_{k=1}^{n} E\left[I_k\right] + \sum_{k\neq l} E\left[I_k\right]E\left[I_l\right] - n^2p(x_0) \\ =& np(x_0) + n(n-1)p(x_0)^2 - n^2p(x_0) = np(x_0)[1-p(x_0)] \leq n.\end{aligned}$$

Putting these together with Chebychev's inequality we get

$$\begin{aligned}\lim_{n\to\infty} \Pr\left[|N(x_0;\mathbf{x}) - np(x_0)| \geq n\epsilon(n)|\mathcal{X}|^{-1}\right] &\leq \lim_{n\to\infty} \frac{\operatorname{var}(N(x_0;\mathbf{x}))}{(n\epsilon(n)|\mathcal{X}|^{-1})^2} \\ &\leq \lim_{n\to\infty} \frac{n}{(n\epsilon(n)|\mathcal{X}|^{-1})^2} = \lim_{n\to\infty} \frac{1}{n\epsilon(n)^2|\mathcal{X}|^{-2}}. \qquad \text{(C.4)}\end{aligned}$$

The limit in (C.4) is zero as long as $\lim_{n\to\infty} \frac{1}{n\epsilon(n)^2} = 0$. This condition is satisfied as long as $\frac{1}{\epsilon(n)^2} \in o(n)$, e.g., $\epsilon(n) = \frac{1}{\sqrt{n}^{1-\delta}}$ where $0 < \delta \leq 1$.

Since there are only finitely many values of $x_0 \in \mathcal{X}$, it follows that

$$\begin{aligned}&\lim_{n\to\infty} \Pr\left[\bigcup_{x_0\in\mathcal{X}} \{\mathbf{x} : |N(x_0;\mathbf{x}) - np(x_0)| > n\epsilon(n)|\mathcal{X}|^{-1}\}\right] \\ &\qquad \leq \lim \sum_{x_0} \Pr\left[\{\mathbf{x} : |N(x_0;\mathbf{x}) - np(x_0)| > n\epsilon(n)|\mathcal{X}|^{-1}\}\right] \qquad \text{(C.5)} \\ &\qquad \leq \lim \sum_{x_0} \frac{1}{n\epsilon(n)^2|\mathcal{X}|^{-1}} = \lim \frac{1}{n\epsilon(n)^2} = 0, \qquad \text{(C.6)}\end{aligned}$$

where the final equality holds as long as $1/\epsilon(n)^2 \in o(n)$. □

To understand the properties of the set $T_X^n(\epsilon(n))$ lets also define the a standard strongly typical set $T_X^n(\epsilon_0)$ where $\epsilon_0 = \epsilon(0)$. First, $T_X^n(\epsilon(n)) \subset T_X^n(\epsilon_0)$ since the definition of the former is more restrictive. However, $T_X^n(\epsilon(n))$ can only be slightly smaller than

$T_x^n(\epsilon)$ since $\lim_{n\to\infty}\Pr[T_x^n(\epsilon(n))] = \lim_{n\to\infty}\Pr[T_x^n(\epsilon_0)] = 1$, i.e., both sets contain most of the probability mass. Because both sets contain the same set of sequences that make up most of the probability mass, all the properties of $T_x^n(\epsilon_0)$ can be re-derived for $T_x^n(\epsilon(n))$. Therefore, in the following we treat $T_x^n(\epsilon(n))$ in the same way we would regular typical sets.

Example: How the sets $T_x^n(\epsilon(n))$ and $T_x^n(\epsilon_0)$ differ. To see how $T_x^n(\epsilon(n))$ and $T_x^n(\epsilon_0)$ differ, consider the definition of $\epsilon(n)$. If $1/\epsilon(n)^2 \notin o(n)$, the set size decreases too quickly in n to contain most of the probability mass. In this case $\lim_{n\to\infty}\Pr[T_x^n(\epsilon(n))] < 1$. In such cases the two sets clearly have very different properties. This follows because for such $\epsilon(n)$ while $T_x^n(\epsilon_0)$ contains most of the probability mass, $T_x^n(\epsilon(n))$ contains almost none. □

C.3 Approach 1: The Dithered Encoding Rule

In [18] a dithered encoding rule that maps an i.i.d. source $\mathbf{y}$ to a codeword $\mathbf{z}$ is proposed for rate-distortion coding. The code used is a standard rate-distortion code $\mathcal{C}$ consisting of independent i.i.d. codewords generated according to $p_{\mathsf{z}}(\mathbf{z}) = \prod_{i=1}^n p_{\mathsf{z}}(z_i)$. We use a test channel $p_{\mathsf{z}|\mathsf{y}}(z|y)$ to relate the random source y to the codewords $\mathbf{z}$ such that $p_{\mathsf{z}|\mathsf{y}}(z|y)$ satisfies $E\left[D(\mathsf{y},\mathsf{z})\right] \le d$.

Instead of joint typicality encoding, we use the following dithered encoding rule:

1. To encode the source $\mathbf{y}$ generate an intermediate random vector $\mathbf{w}$ according to

$$p_{\mathbf{w}|\mathbf{y}}(\mathbf{w}|\mathbf{y}) = \prod_{i=1}^{n} p_{\mathsf{z}|\mathsf{y}}(w_i|y_i),$$

   where $p_{\mathsf{z}|\mathsf{y}}$ is the test channel discussed above.

2. We use the notation $\mathcal{T}(\mathbf{x})$ to denote the empirical distribution (the type) of the sequence $\mathbf{x}$. If $D(\mathcal{T}(\mathbf{y},\mathbf{w})\|p_{\mathsf{y},\mathsf{z}}) > \delta$ an encoding error has occurred, choose index $m = 0$.

3. If $|H(\mathcal{T}(\mathbf{y},\mathbf{w})) - H(\mathsf{y},\mathsf{z})| > \delta$ an encoding error has occurred, choose index $m = 0$.

4. Assuming the above two errors do not occur, list all codewords $\mathbf{z} \in \mathcal{C}$ such that $\mathcal{T}(\mathbf{y},\mathbf{z}) = \mathcal{T}(\mathbf{y},\mathbf{w})$. If this list is of size zero, an encoding error has occurred, choose index $m = 0$.

5. Assuming the list is of non-zero size, randomly and uniformly over the list choose a codeword index m and encoder $\mathbf{y}$ to $\mathbf{z}(m)$.

It can be shown that as n gets very large the probability that this encoding rule is successful (i.e., finds a $\mathbf{z} \in \mathcal{C}$ that satisfies the distortion constraint) can be made arbitrarily close to one. In addition, the following theorem can be proven,

Theorem 10 *Consider any binary valued test, $M[\cdot]$, operating on the pair of sequences $(\mathbf{y}, \tilde{\mathbf{z}})$ where $\tilde{\mathbf{z}}$ is selected according to the conditional distribution $p_{\mathbf{z}|\mathbf{y}}(\tilde{\mathbf{z}}|\mathbf{y}) = \prod_{i=1}^{n} p_{z|y}(\tilde{z}_i|y_i)$. For any $\epsilon > 0$, there exists a $\delta > 0$ and an $n_0 > 0$ such that for all $n > n_0$,*

$$\Pr\{M[\mathbf{y}, \mathbf{z}(m)] \neq M[\mathbf{y}, \tilde{\mathbf{z}}]\} < \epsilon,$$

where $\mathbf{z}(m)$ is an encoding of $\mathbf{y}$ according to the dither encoding rule.

In effect what this theorem says is that the relationship between $\mathbf{y}$ and $\mathbf{z}(m)$ cannot be distinguished from the relationship between the pair $\mathbf{y}$ and $\tilde{\mathbf{z}}$ generated in a pairwise i.i.d. fashion. This relationship guaranteed by this coding structure will allow us to prove the Serial Markov Lemma.

■ C.3.1 Proof Set-up

Say $\mathbf{x}$, $\mathbf{y}$ are jointly typical sample vectors $(\mathbf{x}, \mathbf{y}) \in T_{x,y}^{n}(\delta_{xy})$. Because this implies that $\mathbf{y}$ is marginally typical, we can transcode it into a random vector $\mathbf{z}$ using another code. We will show that using the dithered encoding rule to do the transcoding guarantees that $(\mathbf{x}, \mathbf{z})$ will be jointly typical according to $p(x, z)$. We base our development on that of the Markov Lemma in [8]. The probability that the trancoding does not work is

$$\begin{aligned}
P_{\text{err}} &= \Pr\left[(\mathbf{x}, \mathbf{z}) \notin T_{x,z}^{n}(\epsilon)\right] && \text{(C.9)}\\
&= \Pr\left[\bigcup_{x_0, z_0} \left(|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|\right) \geq n\delta_{xz}|\mathcal{X}\mathcal{Z}|^{-1}\right] && \text{(C.10)}\\
&\leq \sum_{x_0, z_0} \Pr\left[\left(|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|\right) \geq n\delta_{xz}|\mathcal{X}\mathcal{Z}|^{-1}\right]. && \text{(C.11)}
\end{aligned}$$

We will show that for each $(x_0, z_0) \in \mathcal{X} \times \mathcal{Z}$,

$$\lim_{n\to\infty} \Pr\left(\left[|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|\right] \geq n\delta_{xz}|\mathcal{X}\mathcal{Z}|^{-1}\right) = 0 \qquad \text{(C.12)}$$

We will find it useful to rewrite $N(x_0, z_0; \mathbf{x}, \mathbf{z})$ in terms of the dependence on the intermediate vector $\mathbf{y}$:

$$\begin{aligned}
N(x_0, z_0; \mathbf{x}, \mathbf{z}) &= \sum_{y \in \mathcal{Y}} N(x_0, y, z_0; \mathbf{x}, \mathbf{y}, \mathbf{z})\\
&= \sum_{y \in \mathcal{Y}} \left[\sum_{k \in S(x_0, y)} I_k(y, z_0)\right] && \text{(C.13)}
\end{aligned}$$

where $S(x_0, y) = \{i : 1 \leq i \leq n, (x_i, y_i) = (x_0, y)\}$ is a subset of $\{1, \ldots, n\}$, and the $I_k(y, z_0)$ are indicator random variables where $I_k(y, z_0) = 1$ if $y_k = y$ and $z_k = z_0$, and zero otherwise. In the classic Markov Lemma [8] the $I_k(y, z_0)$ are binary i.i.d. random variables with parameter $p_1 = p(z_0|y)$, where $p_1 = \Pr[I_k(y, z_0) = 1] = 1 - \Pr[I_k(y, z_0) =$

0]. This follows because in that situation $\mathbf{z}$ is generated conditionally from $\mathbf{y}$ in a memoryless manner. In the current setting, $\mathbf{z}$ is not generated from $\mathbf{y}$ in a memoryless manner, but we will show that the dithering at the encoder is enough to guarantee the joint typicality of $\mathbf{x}$ and $\mathbf{z}$.

C.3.2 Convergence of the Mean $E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right]$

$$
\begin{aligned}
E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right] &= \sum_{y \in \mathcal{Y}} E_{\mathbf{z}}\left[\sum_{k \in S(x_0,y)} I_k(y, z_0)\right] && \text{(C.14)}\\
&= \sum_{y \in \mathcal{Y}} E_S\left[\sum_{k \in S(x_0,y)} E_{\mathbf{z}|S}\left[I_k(y, z_0)\right]\right] && \text{(C.15)}\\
&\leq \sum_{y \in \mathcal{Y}} E_S\left[\sum_{k \in S(x_0,y)}\right](p(z_0|y) + 2C\delta_{yz}) && \text{(C.16)}\\
&\leq \sum_{y \in \mathcal{Y}} (np(x_0, y) + n\delta_{xy}|\mathcal{X}\mathcal{Y}|^{-1})(p(z_0|y) + 2C\delta_{yz}) && \text{(C.17)}\\
&= \sum_{y \in \mathcal{Y}} n[p(x_0, y, z_0) + p(z_0|y)\delta_{xy}|\mathcal{X}\mathcal{Y}|^{-1} + 2C\delta_{yz}p(x_0, y) + 2C\delta_{xy}\delta_{yz}|\mathcal{Y}\mathcal{Z}|^{-1}] &&\\
&\leq n[p(x_0, z_0) + \delta_{xy}|\mathcal{Z}|^{-1} + \delta_{yz}(2C|\mathcal{Y}| + 2C\delta_{xy}|\mathcal{Z}|^{-1}) &&\\
&\leq np(x_0, z_0) + n\delta_{xy}(|\mathcal{Z}|^{-1} + 2C|\mathcal{Y}| + 2C|\mathcal{Z}|^{-1}) && \text{(C.18)}\\
&= n[x(x_0, z_0) + \delta_{xy}\kappa_1] && \text{(C.19)}
\end{aligned}
$$

Line	Justification
(C.14)	Substituting in (C.13) and the linearity of expectation.
(C.15)	Iterated expectations.
(C.16)	The pair $(\mathbf{y}, \mathbf{z})$ cannot be differentiated from a pair-wise i.i.d. pair, Thm. 10. Hence their first-order statistics must be stationary and will approximate the test channel, $p(z_0|y)$.
(C.17)	$\lvert S(x_0, y)\rvert \leq (np(x_0, y) + n\delta_{xy}\lvert\mathcal{X}\mathcal{Y}\rvert^{-1})$ since $(\mathbf{x}, \mathbf{y}) \in T^n_{x,y}(\delta_{xy})$.
(C.18)	Choose $\delta_{xy} = \delta_{yz}$.
(C.19)	Define $\kappa_1 = \lvert\mathcal{Z}\rvert^{-1} + 2C\lvert\mathcal{Y}\rvert + 2C\lvert\mathcal{Z}\rvert^{-1}$.

We can similarly develop the parallel lower-bound,

$$
E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right] \geq n[p(x_0, z_0) - \delta_{xy}\kappa_1]. \tag{C.20}
$$

From (C.18) and (C.20) we see that the mean of $N(x_0, z_0; \mathbf{x}, \mathbf{z})$ converges to $p(x_0, z_0)$. We next show that the growth in n of $N(x_0, z_0; \mathbf{x}, \mathbf{z})$.

C.3.3 Bounding the Growth of var$(N(x_0, z_0; \mathbf{x}, \mathbf{z}))$

We begin with a useful lemma

Lemma 2 *If* $(\mathbf{x},\mathbf{y}) \in T^n_{\mathsf{x},\mathsf{y}}(\delta_{\mathsf{xy}})$, *then*

$$\sum_{k\in S(\tilde{y})}\sum_{l\in S(\bar{y})} I_k(x_0,\tilde{y})I_l(x_0,\bar{y}) \le n^2\{p(x_0,\tilde{y})p(x_0,\bar{y}) + 3\delta_{\mathsf{xy}}|\mathcal{X}\mathcal{Y}|^{-1}\}. \tag{C.21}$$

Proof:
Because $(\mathbf{x},\mathbf{y}) \in T^n_{\mathsf{x},\mathsf{y}}(\delta_{\mathsf{xy}})$,

$$[N(x_0,\tilde{y}) - np(x_0,\tilde{y})][N(x_0,\bar{y}) - np(x_0,\bar{y})] \le n^2\delta^2_{\mathsf{xy}}|\mathcal{X}\mathcal{Y}|^{-2} \tag{C.22}$$

since the conditions for the joint typicality of $(\mathbf{x},\mathbf{y})$ imply that $|N(x_0,y) - np(x_0,y)| < n\delta_{\mathsf{xy}}|\mathcal{X}\mathcal{Y}|^{-1}$. Rearranging terms gives

$$N(x_0,\tilde{y})N(x_0,\bar{y}) \le n^2[p(x_0,\tilde{y})p(x_0,\bar{y}) + \delta^2_{\mathsf{xy}}|\mathcal{X}\mathcal{Y}|^{-2} + 2\delta_{\mathsf{xy}}|\mathcal{X}\mathcal{Y}|^{-1}] \tag{C.23}$$

$$\sum_{k\in S(\tilde{y})}\sum_{l\in S(\bar{y})} I_k(x_0,\tilde{y})I_l(x_0,\bar{y}) \le n^2\{p(x_0,\tilde{y})p(x_0,\bar{y}) + 3\delta_{\mathsf{xy}}|\mathcal{X}\mathcal{Y}|^{-1}\}, \tag{C.24}$$

where (C.23) follows from $N(x_0,\tilde{y}) < p(x_0,\tilde{y}) + n\delta_{\mathsf{xy}}|\mathcal{X}\mathcal{Y}|^{-1}$, and (C.24) is (C.23) rewritten in terms of indicator functions where $S(\tilde{y}) = \{i : 1 \le i \le n, y_i = \tilde{y}\}$. □

We now use Lemma 2 to show how to bound the variance of $N(x_0,z_0;\mathbf{x},\mathbf{z})$ as a function of n.

$$\mathrm{var}(N(x_0,z_0;\mathbf{x},\mathbf{z})) = E_{\mathbf{z}|\mathbf{y}}\{[N(x_0,z_0)]^2\} - E\{N(x_0,z_0)\}^2$$

$$\le E_{\mathbf{z}|\mathbf{y}}\left\{\left[\sum_{\tilde{y}}\sum_{k\in S(\tilde{y})} I_k(x_0,\tilde{y})I_k(\tilde{y},z_0)\right]\left[\sum_{\bar{y}}\sum_{l\in S(\bar{y})} I_l(x_0,\bar{y})I_l(\bar{y},z_0)\right]\right\} - n^2(p(x_0,z_0) - \delta_{\mathsf{xy}}\kappa_1)^2 \tag{C.25}$$

$$= \sum_{\tilde{y}}\sum_{\bar{y}}\left\{E_{\mathbf{z}|\mathbf{y}}\left[\sum_{k\in S(\tilde{y})}\sum_{l\in S(\bar{y})} I_k(x_0,\tilde{y})I_l(x_0,\bar{y})I_k(\tilde{y},z_0)I_l(\bar{y},z_0)\right]\right.$$
$$\left. - n^2p(x_0,\tilde{y},z_0)p(x_0,\bar{y},z_0) + n^2 2\delta_{\mathsf{xy}}\kappa_1|\mathcal{Y}|^{-2}\right\} \tag{C.26}$$

$$= \sum_{\tilde{y}}\sum_{\bar{y}}\left\{\sum_{k\in S(\tilde{y})}\sum_{l\in S(\bar{y})} I_k(x_0,\tilde{y})I_l(x_0,\bar{y})E_{\mathbf{z}|\mathbf{y}}[I_k(\tilde{y},z_0)I_l(\bar{y},z_0)]\right\}$$
$$- n^2p(x_0,\tilde{y},z_0)p(x_0,\bar{y},z_0) + n^2 2\delta_{\mathsf{xy}}\kappa_1|\mathcal{Y}|^{-2}\Big\} \tag{C.27}$$

$$\begin{aligned}
&\leq \sum_{\tilde{y}} \sum_{\bar{y}} \left\{ \left[\sum_{k \in S(\tilde{y})} \sum_{l \in S(\bar{y})} I_k(x_0, \tilde{y}) I_l(x_0, \bar{y}) \right] [p_{z|y}(z_0|\tilde{y}) p_{z|y}(z_0|\bar{y}) + \delta_{yz} |\mathcal{Y}\mathcal{Z}|^{-1}] \right. \\
&\qquad \left. - n^2 p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + n^2 2\delta_{xy} \kappa_1 |\mathcal{Y}|^{-2} \right\} && \text{(C.28)} \\
&\leq \sum_{\tilde{y}} \sum_{\bar{y}} \{ [n^2 p(x_0, \tilde{y}) p(x_0, \bar{y}) + n^2 3\delta_{xy} |\mathcal{X}\mathcal{Y}|^{-2}][p(z_0|\tilde{y}) p(z_0|\bar{y}) + \delta_{\text{dith}}] \\
&\qquad - n^2 p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + n^2 2\delta_{xy} |\mathcal{Y}|^{-2} \} && \text{(C.29)} \\
&\leq \sum_{\tilde{y}} \sum_{\bar{y}} \{ n^2 [p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + \delta_{\text{dith}} + 3\delta_{xy} |\mathcal{X}\mathcal{Y}|^{-1} + 3\delta_{xy} \delta_{\text{dith}} |\mathcal{X}\mathcal{Y}|^{-1} \\
&\qquad - p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + 2\delta_{xy} |\mathcal{Y}|^{-2} \} && \text{(C.30)} \\
&\leq n^2 \bar{\delta} |\text{const}| && \text{(C.31)}
\end{aligned}$$

Line	Justification
(C.25)	Writing out $N(x_0, z_0; \mathbf{x}, \mathbf{z})$ as in (C.13) and expanding $\tilde{S}$ and $\bar{S}$ similarly.
(C.26)	$p(x_0, z_0) = \sum_y p(x_0, y, z_0)$.
(C.27)	The sets $S(\tilde{y})$ and $S(\bar{y})$ and the indicator functions $I_k(x_0, \tilde{y})$ and $I_l(x_0, \bar{y})$ are independent of $\mathbf{z}$.
(C.28)	Because if we use the dither encoding rule of Section C.3 we cannot tell the resultant $\mathbf{z}$ apart from one generated in a memoryless manner conditionally on $\mathbf{y}$. Hence the pairwise statistics factor into a product of test channel statistics.
(C.29)	Lemma 2.
(C.30)	$p(x, y, z) = p(x, y) p(z|y)$.
(C.31)	$\bar{\delta} = \max\{\delta_{xy}, \delta_{\text{dith}}\}$.

■ C.3.4 Mean-Square Convergence

From Chebychev's inequality we now have

$$\begin{aligned}
&\lim_{n \to \infty} \Pr \left[(|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|) \geq n\delta |\mathcal{X}\mathcal{Z}|^{-1} \right] \\
&\qquad \leq \lim_{n \to \infty} \frac{\text{var}(N(x_0, z_0; \mathbf{x}, \mathbf{z}))}{(n\delta |\mathcal{X}\mathcal{Z}|^{-1})^2} \leq \lim_{n \to \infty} \frac{n^2 \bar{\delta} |\text{const}|}{n^2 \delta_{xz}^2 |\mathcal{X}\mathcal{Z}|^{-1}} && \text{(C.32)} \\
&\qquad = \lim_{n \to \infty} \frac{\bar{\delta}}{\delta_{xz}^2} |\text{const}| |\mathcal{X}\mathcal{Z}| && \text{(C.33)}
\end{aligned}$$

We can make the limit (C.33) converge to 0 by picking $\bar{\delta}$ and δ_{xz} to be functions of n, per Section C.2, such that $\lim_{n \to \infty} \frac{\bar{\delta}}{(\delta_{xz})^2} = 0$. This mean-squared convergence holds for any pair of sample values (x_0, z_0). Therefore, for any ϵ we can find a n_0 such that for all $n > n_0$ the $\Pr \left[(\mathbf{x}, \mathbf{z}) \notin T^n_{x,z}(\epsilon) \right] < \epsilon$.

■ C.4 Approach 2: Viswanathan-Berger (VB) Codes

In [75] Viswanathan and Berger introduce a class of codes (which we term VB codes) that have some useful stationary properties. In this section we show that if $\mathbf{x}$ is encoded into $\mathbf{y}$ using a VB code, and then $\mathbf{y}$ is transcoded into $\mathbf{z}$ using a VB code, the pair $(\mathbf{x}, \mathbf{z})$ will be jointly typical. This is a slightly more restrictive serial relationship than the proof of Section C.3 that used the dithered encoding rule. This is because when using VB codes we require both encoding steps to use VB codes, with the dither encoding rule all we required was that $(\mathbf{x}, \mathbf{y})$ were jointly typical. We present this proof in addition because, although it is more involved, it seems that VB codes may have just enough structure to guarantee the joint typicality of $(\mathbf{x}, \mathbf{z})$, while the dither encoding rule may be more powerful than we need.

■ C.4.1 Introduction to VB Codes

Let $\mathbf{y}$ be an i.i.d. sequence of finite-alphabet random variables where $p(y) > 0$ for all $y \in \mathcal{Y}$. Let z be a random variable talking values in $\mathcal{Z}$ with the conditional probability of z given y being $p(z|y)$. Let f^n be a n-length block code from $\mathcal{Y}^n$ to $\mathcal{Z}^n$. The map f^n induces a joint distribution on $\mathbf{y}$ and $\mathbf{z}$ given by

$$\hat{p}(\mathbf{y}, \mathbf{z}) = p(\mathbf{y}) I(f^n(\mathbf{y}) = \mathbf{z}), \tag{C.34}$$

where $I(\cdot)$ is an indicator function, and the corresponding marginals are given by

$$\begin{aligned} \hat{p}(y_i = y, z_i = z) &= E_{p(\mathbf{y})} I(y_i = y, z_i = z) && \text{(C.35)} \\ \hat{p}(z_i = z | y_i = y) &= \frac{\hat{p}(y_i = y, z_i = z)}{p(y_i = y)}. && \text{(C.36)} \end{aligned}$$

In Lemma A.1 of [75] the authors show

Theorem 11 *[75] For every $\delta > 0$, and n sufficiently large, there exists a block code $f^n : \mathcal{Y}^n \to \mathcal{Z}^n$ such that*

1. $f^n(\sigma^k(\mathbf{y})) = \sigma^k f^n(\mathbf{y})$ *where* $\sigma(y_1, y_2, \ldots, y_n) = (y_n, y_1, y_2, \ldots, y_{n-1})$
2. *The range M of f^n is bounded by $M \leq 2^{n(I(y;z)+\delta)}$.*
3. $|\hat{p}_{z_i|y_i}(z|y) - p_{z_i|y_i}(z|y)| \leq 2C\epsilon$ *for all i, where $C = 1/\min_y p_y(y)$ and $\hat{p}(z|y)$ is the conditional distribution between encoder input and codeword induced by the codebook.*

In essence, what Thm. 11 tells us is that the relationship between y_i and z_i is first-order stationary since $p_{z_i|y_i}(y|x)$ is the same for all i. We now show that Thm. 11 can easily be extended to show that source and codeword pairs are strict-sense cyclo-stationarity.

Lemma 3 *The source-symbol codeword-symbol relationship for VB Codes is strict-sense cyclo-stationarity.*

Proof:

$$\begin{aligned}
&\hat{p}(z_t = \tilde{z}, y_t = \tilde{y}, z_s = \bar{z}, y_s = \bar{y}) \\
&\quad = E_{\mathbf{y}}[I(z_t = \tilde{z}, y_t = \tilde{y}, z_s = \bar{z}, y_s = \bar{y})] && \text{(C.37)} \\
&\quad = E_{\mathbf{y}}[I(f_t(\mathbf{y}) = \tilde{z}, y_t = \tilde{y}, f_s(\mathbf{y}) = \bar{z}, y_s = \bar{y})] && \text{(C.38)} \\
&\quad = E_{\mathbf{y}}[I(f_1(\sigma^{-t+1}\mathbf{y}) = \tilde{z}, y_t = \tilde{y}, f_{s-t+1}(\sigma^{-t+1}\mathbf{y}) = \bar{z}, y_s = \bar{y})] && \text{(C.39)} \\
&\quad = E_{\sigma^{-t+1}\mathbf{y}}[I(f_1(\mathbf{y}) = \tilde{z}, y_1 = \tilde{y}, f_{s-t+1}(\mathbf{y}) = \bar{z}, y_{s-t+1} = \bar{y})] && \text{(C.40)} \\
&\quad = E_{\mathbf{y}}[I(f_1(\mathbf{y}) = \tilde{z}, y_1 = \tilde{y}, f_{s-t+1}(\mathbf{y}) = \bar{z}, y_{s-t+1} = \bar{y})] && \text{(C.41)} \\
&\quad = \hat{p}(z_1 = \tilde{z}, y_1 = \tilde{y}, z_{s-t+1} = \bar{z}, y_{s-t+1} = \bar{y}), && \text{(C.42)}
\end{aligned}$$

where $(s-t+1)$ should be taken $\mod n$ to remain in the range $1, \dots n$.[2] Eq. (C.37) follows from the definition of $\hat{p}(\cdot)$, (C.38) since $\mathbf{z} = f(\mathbf{y})$ where the subscript $f_t(\mathbf{y})$ shows particular sample t explicitly, (C.39) from the definition of σ, (C.40) from cycling the $\mathbf{y}$ vector, and (C.41) since $\mathbf{y}$ is i.i.d. □

■ C.4.2 Proof Set-up

Let $\mathbf{x}$ be encoded into $\mathbf{y}$ using a VB code. We show that if we transcode $\mathbf{y}$ into $\mathbf{z}$ using a second VB code then $(\mathbf{x}, \mathbf{z})$ will be jointly typical according to $p(x, z)$ with probability approaching one as n grows to infinity. As in the proof for the dithered encoding rule, the probability that the trancoding does not work is

$$\begin{aligned}
P_{\text{err}} &= \Pr\left[(\mathbf{x}, \mathbf{z}) \notin T^n_{x,z}(\epsilon)\right] && \text{(C.43)} \\
&= \Pr\left[\bigcup_{x_0, z_0} (|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|) \geq n\delta_{xz}|\mathcal{X}\mathcal{Z}|^{-1}\right] && \text{(C.44)} \\
&\leq \sum_{x_0, z_0} \Pr\left[(|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|) \geq n\delta_{xz}|\mathcal{X}\mathcal{Z}|^{-1}\right]. && \text{(C.45)}
\end{aligned}$$

We will show that for each $(x_0, z_0) \in \mathcal{X} \times \mathcal{Z}$,

$$\lim_{n \to \infty} \Pr\left([|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|] \geq n\delta_{xz}|\mathcal{X}\mathcal{Z}|^{-1}\right) = 0 \qquad \text{(C.46)}$$

Again, as in Section C.3, we will find it useful to rewrite $N(x_0, z_0; \mathbf{x}, \mathbf{z})$ in terms of the dependence on the intermediate vector $\mathbf{y}$:

$$\begin{aligned}
N(x_0, z_0; \mathbf{x}, \mathbf{z}) &= \sum_{y \in \mathcal{Y}} N(x_0, y, z_0; \mathbf{x}, \mathbf{y}, \mathbf{z}) \\
&= \sum_{y \in \mathcal{Y}} \left[\sum_{k \in S(x_0, y)} I_k(y, z_0)\right] && \text{(C.47)}
\end{aligned}$$

---

[2] Note, for these purposes $n \mod n = n$ (instead of zero) since the numbering of source symbols traditionally starts with 1 (not 0).

where $S(x_0, y) = \{i : 1 \le i \le n, (x_i, y_i) = (x_0, y)\}$ is a random subset of $\{1, \dots, n\}$, and the $I_k(y, z_0)$ are indicator random variables where $I_k(y, z_0) = 1$ if $y_k = y$ and $z_k = z_0$, and zero otherwise. In the current setting, the relationship between $\mathbf{y}$ and $\mathbf{z}$ is not necessarily i.i.d. but is stationary. We show that this is all that is necessary for joint typicality of $\mathbf{x}$ and $\mathbf{z}$.

■ C.4.3 Convergence of Mean $E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right]$

$$E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right] = \sum_{y \in \mathcal{Y}} E_{\mathbf{x},\mathbf{y},\mathbf{z}}\left[\sum_{k \in S(x_0, y)} I_k(y, z_0)\right] \tag{C.48}$$

$$= \sum_{y \in \mathcal{Y}} E_S\left[\sum_{k \in S(x_0, y)} E_{\mathbf{x},\mathbf{y},\mathbf{z}|S}\left[I_k(y, z_0)\right]\right] \tag{C.49}$$

$$= \sum_{y \in \mathcal{Y}} E_S\left[\sum_{k \in S(x_0, y)} E_{\mathbf{y}|S}\left[I_k(y, z_0)\right]\right] \tag{C.50}$$

$$\le \sum_{y \in \mathcal{Y}} E_S\left[\sum_{k \in S(x_0, y)}\right](p(z_0|y) + 2C\delta_{yz}) \tag{C.51}$$

$$\le \sum_{y \in \mathcal{Y}} (np(x_0, y) + n\delta_{xy}|\mathcal{X}\mathcal{Y}|^{-1})(p(z_0|y) + 2C\delta_{yz}) \tag{C.52}$$

$$= \sum_{y \in \mathcal{Y}} n[p(x_0, y, z_0) + p(z_0|y)\delta_{xy}|\mathcal{X}\mathcal{Y}|^{-1} + 2C\delta_{yz}p(x_0, y) + 2C\delta_{xy}\delta_{yz}|\mathcal{Y}\mathcal{Z}|^{-1}]$$

$$\le n[p(x_0, z_0) + \delta_{xy}|\mathcal{Z}|^{-1} + \delta_{yz}(2C|\mathcal{Y}| + 2C\delta_{xy}|\mathcal{Z}|^{-1})$$

$$\le np(x_0, z_0) + n\delta_{xy}(|\mathcal{Z}|^{-1} + 2C|\mathcal{Y}| + 2C|\mathcal{Z}|^{-1}) \tag{C.53}$$

$$= n[x(x_0, z_0) + \delta_{xy}\kappa_1 \tag{C.54}$$

Line	Justification
(C.48)	Substituting in (C.47) and the linearity of expectation.
(C.49)	Iterated expectations.
(C.50)	Once condition on $S(x_0, y)$, $I_k(y, z_0)$ is independent of $\mathbf{x}$ and $\mathbf{z} = f^n(\mathbf{y})$ deterministically.
(C.51)	First-order stationarity of VB codes and because the code statistics approximate the test channel, $p(z_0|y)$.
(C.52)	$|S(x_0, y)| \le (np(x_0, y) + n\delta_{xy}|\mathcal{X}\mathcal{Y}|^{-1})$ since $(\mathbf{x}, \mathbf{y}) \in T^n_{\mathsf{x},\mathsf{y}}(\delta_{xy})$.
(C.53)	Choose $\delta_{xy} = \delta_{yz}$.
(C.54)	Define $\kappa_1 = |\mathcal{Z}|^{-1} + 2C|\mathcal{Y}| + 2C|\mathcal{Z}|^{-1}$.

We can similarly develop the parallel lower-bound,

$$E\left[N(x_0, z_0; \mathbf{x}, \mathbf{z})\right] \ge n[p(x_0, z_0) - \delta_{xy}\kappa_1]. \tag{C.55}$$

From (C.53) and (C.55) we see that the mean of $N(x_0, z_0; \mathbf{x}, \mathbf{z})$ converges to $p(x_0, z_0)$. We next bound the variance of $N(x_0, z_0; \mathbf{x}, \mathbf{z})$.

C.4.4 Bounding the Growth of var($N(x_0, z_0; \mathbf{x}, \mathbf{z})$)

In order to bound the variance of $N(x_0, z_0; \mathbf{x}, \mathbf{z})$, we would like to have a lemma similar to the following conjecture:

Conjecture 1 *Given that* $\mathbf{y}$ *is an i.i.d. source sequence, and* $\mathcal{C}$ *is a rate-distortion code generated in an i.i.d. manner according to* $p(z)$*, such that* $p(y, z) = p(y)p(z|y)$*, and satisfying a distortion constraint. Then, if* $\mathbf{y}$ *is mapped to a codeword* $\mathbf{z} \in \mathcal{C}$ *using joint typicality decoding, inducing a joint distribution* $\hat{p}(\mathbf{y}, \mathbf{z}) = E_y[I(f^n(\mathbf{y}) = \mathbf{z})]$*, then the following relationship holds for any pair of indices* i*,* j*:*

$$\lim_{n \to \infty} \hat{p}(z_i, z_j | y_i, y_j) = \hat{p}(z_i | y_i) \hat{p}(z_j | y_j).$$

This proposition says that even though on a block level the relationship between $\mathbf{y}$ and $\mathbf{c}$ is deterministic, as the block length grows, the pair-wise statistics look increasingly independent. If we could prove this proposition, then calculation of the variance would be easy as cross terms such as $E[I_k(\cdot,\cdot) I_l(\cdot,\cdot)]$ could be factored in the limit. The dithered encoding rule gives us this property, but the encoding rule for VB Codes does not. We are optimistic, however, that entropy-bounding arguments will be enough to prove the above lemma, but in the meantime, the bounding of $\mathrm{var}(N(x_0, z_0; \mathbf{x}, \mathbf{z}))$ remains more involved for VB Codes.

We first present the following lemma.

Lemma 4 *If* $(\mathbf{x}, \mathbf{y}) \in T^n_{\mathsf{x},\mathsf{y}}(\delta_{xy})$ *and we define* $\tilde{S} = S(x_0, \tilde{y}) = \{i, 1 \le i \le n, (\mathsf{x}_i, \mathsf{y}_i) = (x_0, \tilde{y})\}$ *and* $\bar{S} = S(x_0, \bar{y}) = \{i, 1 \le i \le n, (\mathsf{x}_i, \mathsf{y}_i) = (x_0, \bar{y})\}$*, then if we encode* $\mathbf{y}$ *into* $\mathbf{z}$ *using a VB code, we get*

$$\begin{aligned} \frac{1}{n} \sum_{\tau=1}^{n} E_{\mathbf{y}}[I(\mathsf{z}_k = z_0, \mathsf{z}_{k+\tau} = z_0 | \mathsf{y}_k = \tilde{y}, \mathsf{y}_{k+\tau} = \bar{y})] &= \frac{1}{n} \sum_{\tau=1}^{n} \hat{p}_{\mathsf{z}_1, \mathsf{z}_{1+\tau} | \mathsf{y}_1, \mathsf{y}_{1+\tau}}(z_0, z_0 | \tilde{y}, \bar{y}) \\ &\le p_{\mathsf{z}|\mathsf{y}}(z_0 | \tilde{y}) p_{\mathsf{z}|\mathsf{y}}(z_0 | \bar{y}) + \delta_{xy} \kappa_2 \end{aligned}$$

for some positive constant κ_2*.*

Proof:

$$\begin{aligned}
\sum_{t=1}^{n}\sum_{s=1}^{n}\hat{p}_{z_t,y_t,z_s,y_s}(\tilde{z},\tilde{y},\bar{z},\bar{y}) &= \sum_{t=1}^{n}\sum_{s=1}^{n}E_{\mathbf{y}}\Big\{I[z_t=\tilde{z},y_t=\tilde{y},z_s=\bar{z},y_s=\bar{y}]\Big\} && \text{(C.57)}\\
&= E_{\mathbf{y}}\left\{\sum_{t=1}^{n}\sum_{s=1}^{n}I[z_t=\tilde{z},y_t=\tilde{y},z_s=\bar{z},y_s=\bar{y}]\right\} && \text{(C.58)}\\
&= \sum_{\mathbf{y}\in\mathcal{Y}^n}N(\tilde{z},\tilde{y}|\mathbf{z},\mathbf{y})N(\bar{z},\bar{y}|\mathbf{z},\mathbf{y})p_{\mathbf{y}}(\mathbf{y}) && \text{(C.59)}\\
&\le \sum_{\mathbf{y}\in\mathcal{Y}^n}n^2[p(\tilde{z},\tilde{y})+\delta_{xy}|\mathcal{Y}\mathcal{Z}|^{-1}][p(\bar{z},\bar{y})+\delta_{xy}|\mathcal{Y}\mathcal{Z}|^{-1}]p_{\mathbf{y}}(\mathbf{y}) && \text{(C.60)}\\
&\le n^2[p(\tilde{z},\tilde{y})p(\bar{z},\bar{y})+3\delta_{xy}|\mathcal{Y}\mathcal{Z}|^{-1}]
\end{aligned}$$

Line	Justification
(C.57)	Definition of induced probability distribution.
(C.58)	Linearity of expectation.
(C.59)	Definition of $N(\tilde{z},\tilde{y};\mathbf{z},\mathbf{y})$ and since $\mathbf{z}=f^n(\mathbf{y})$ deterministically.
(C.60)	Upper bounds on $N(\tilde{z},\tilde{y};\mathbf{z},\mathbf{y})$ since encoding assumed to be successful.

As shown in Lemma 3, $\hat{p}$ is only a function of $(t-s)$. Since the block code is of length n, there are n possible differences $\tau=(t-s)$, and n shifts of each difference. Therefore, we can simplify the above to

$$p(\tilde{z},\tilde{y})p(\tilde{z},\tilde{y})+3\delta_{xy}|\mathcal{Y}\mathcal{Z}|^{-1} \ge \frac{1}{n}\sum_{\tau=1}^{n}\hat{p}_{z_1,y_1,z_{1+\tau},y_{1+\tau}}(\tilde{z},\tilde{y},\bar{z},\bar{y}) \tag{C.61}$$

$$\begin{aligned}
&= \frac{1}{n}\sum_{\tau=1}^{n}\hat{p}_{z_1,z_{1+\tau}|y_1,y_{1+\tau}}(\tilde{z},\bar{z}|\tilde{y},\bar{y})\hat{p}_{y_1,y_{1+\tau}}(\tilde{y},\bar{y})\\
&= \frac{1}{n}\sum_{\tau=1}^{n}\hat{p}_{z_1,z_{1+\tau}|y_1,y_{1+\tau}}(\tilde{z},\bar{z}|\tilde{y},\bar{y})p_{y_1,y_{1+\tau}}(\tilde{y},\bar{y})
\end{aligned} \tag{C.62}$$

$$p_{z|y}(\tilde{z}|\tilde{y})p_{z|y}(\bar{z}|\bar{y})p_y(\tilde{y})p_y(\bar{y})+3\delta_{xy}|\mathcal{Y}\mathcal{Z}|^{-1} \ge \frac{1}{n}\sum_{\tau=1}^{n}\hat{p}_{z_1,z_{1+\tau}|y_1,y_{1+\tau}}(\tilde{z},\bar{z}|\tilde{y},\bar{y})p_y(\tilde{y})p_y(\bar{y}) \tag{C.63}$$

$$p(\tilde{z}|\tilde{y})p(\bar{z}|\bar{y})+3\delta_{xy}|\mathcal{Y}\mathcal{Z}|^{-1}C^2 \ge \frac{1}{n}\sum_{\tau=1}^{n}\hat{p}_{z_1,z_{1+\tau}|y_1,y_{1+\tau}}(\tilde{z},\bar{z}|\tilde{y},\bar{y}) \tag{C.64}$$

$$p(\tilde{z}|\tilde{y})p(\bar{z}|\bar{y})+\delta_{xy}\kappa_2 = \frac{1}{n}\sum_{\tau=1}^{n}E_{\mathbf{y}}[I(z_k=z_0,z_{k+\tau}=z_0|y_k=\tilde{y},y_{k+\tau}=\bar{y})] \tag{C.65}$$

Line	Justification
(C.61)	In the double sum each of the n shifts appears n times.
(C.62)	$\hat{p}_{\mathbf{y}}(\mathbf{y}) = p_{\mathbf{y}}(\mathbf{y})$.
(C.63)	$\mathbf{y}$ is i.i.d.
(C.64)	$C = 1/\min_y p_y(y)$.
(C.65)	Definition of $\hat{p}$ and $\kappa_2 \equiv 3C^2|\mathcal{Y}\mathcal{Z}|^{-1}$.

This theorem says that the shift-average of the empirical probability density is roughly equal to the factored density. □

We now use Lemma 4to bound the variance of $N(x_0, z_0; \mathbf{x}, \mathbf{z})$ when using VB codes.

$$\begin{aligned}
\operatorname{var}(N(x_0, z_0; \mathbf{x}, \mathbf{z})) &= E_{\mathbf{x},\mathbf{z}}\{[N(x_0, z_0; \mathbf{x}, \mathbf{z})]^2\} - E_{\mathbf{x},\mathbf{z}}\{N(x_0, z_0; \mathbf{x}, \mathbf{z})\}^2 \\
&\le E_{\mathbf{x},\mathbf{y}}\left\{\left[\sum_{\tilde{y}} \sum_{k \in S(x_0,\tilde{y})} I_k(\tilde{y}, z_0)\right]\left[\sum_{\bar{y}} \sum_{l \in S(x_0,\bar{y})} I_l(\bar{y}, z_0)\right]\right\} - n^2(p(x_0, z_0) - \delta_{xy}\kappa_1)^2
\end{aligned} \tag{C.66}$$

$$\begin{aligned}
=\sum_{\tilde{y}}\sum_{\bar{y}}\Bigg\{ & E_{\mathbf{y},\tilde{S},\bar{S}}\left[\sum_{k \in S(x_0,\tilde{y})} \sum_{l \in S(x_0,\bar{y})} I_k(\tilde{y}, z_0) I_l(\bar{y}, z_0)\right] \\
& - n^2 p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + n^2 2\delta_{xy}\kappa_1|\mathcal{Y}|^{-2}\Bigg\}
\end{aligned} \tag{C.67}$$

$$\begin{aligned}
=\sum_{\tilde{y}}\sum_{\bar{y}}\Bigg\{ & E_{\tilde{S},\bar{S}}\left[\sum_{k \in S(x_0,\tilde{y})} \sum_{l \in S(x_0,\bar{y})} E_{\mathbf{y}|\tilde{S},\bar{S}}[I_k(\tilde{y}, z_0) I_l(\bar{y}, z_0)]\right] \\
& - n^2 p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + n^2 2\delta_{xy}\kappa_1|\mathcal{Y}|^{-2}\Bigg\}
\end{aligned} \tag{C.68}$$

$$\begin{aligned}
=\sum_{\tilde{y}}\sum_{\bar{y}}\Bigg\{ & E_{\tilde{S},\bar{S}}\left[\sum_{k \in S(x_0,\tilde{y})} \sum_{l \in S(x_0,\bar{y})} E_{\mathbf{y}}[I_k(\tilde{y}, z_0) I_l(\bar{y}, z_0)]\right] \\
& - n^2 p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + n^2 2\delta_{xy}\kappa_1|\mathcal{Y}|^{-2}\Bigg\}
\end{aligned} \tag{C.69}$$

$$\begin{aligned}
=\sum_{\tilde{y}}\sum_{\bar{y}}\Bigg\{ & E_{\tilde{S},\bar{S}}\left[\sum_{k \in S(x_0,\tilde{y})} \sum_{l \in S(x_0,\bar{y})} \hat{p}_{\mathsf{z}_k,\mathsf{z}_l|\mathsf{y}_k,\mathsf{y}_l}(z_0, z_0|\tilde{y}, \bar{y})\right] \\
& - n^2 p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + n^2 2\delta_{xy}\kappa_1|\mathcal{Y}|^{-2}\Bigg\}
\end{aligned} \tag{C.70}$$

$$\begin{aligned}
\le\sum_{\tilde{y}}\sum_{\bar{y}}\{ & (np(x_0, \tilde{y}) + n\delta_{xy}|\mathcal{X}\mathcal{Y}|^{-1})(np(x_0, \bar{y}) + n\delta_{xy}|\mathcal{X}\mathcal{Y}|^{-1})(p(z_0|\tilde{y})p(z_0|\bar{y}) + \delta_{xy}\kappa_2) \\
& - n^2 p(x_0, \tilde{y}, z_0) p(x_0, \bar{y}, z_0) + n^2 2\delta_{xy}\kappa_1|\mathcal{Y}|^{-2}\}
\end{aligned} \tag{C.71}$$

$$
\begin{aligned}
\leq & \sum_{\tilde{y}} \sum_{\bar{y}} \{ n^2 [p(x_0,\tilde{y}) p(x_0,\bar{y}) + n^2 3\delta_{xy} |\mathcal{X}\mathcal{Y}|^{-1}] (p(z_0|\tilde{y}) p(z_0|\bar{y}) + \delta_{xy}\kappa_2) \\
& \qquad - n^2 p(x_0,\tilde{y},z_0) p(x_0,\bar{y},z_0) + n^2 2\delta_{xy}\kappa_1 |\mathcal{Y}|^{-2} \} \\
\leq & \sum_{\tilde{y}} \sum_{\bar{y}} \{ n^2 [p(x_0,\tilde{y},z_0) p(x_0,\bar{y},z_0) + 3\delta_{xy} |\mathcal{X}\mathcal{Y}|^{-1} + \delta_{xy}\kappa_2 + 3\delta_{xy}^2 \kappa_2 |\mathcal{X}\mathcal{Y}|^{-1} \\
& \qquad - p(x_0,\tilde{y},z_0) p(x_0,\bar{y},z_0) + 2\delta_{xy}\kappa_1 |\mathcal{Y}|^{-2}] \} \qquad \text{(C.72)} \\
\leq & n^2 \bar{d} |\text{const}|. \qquad \text{(C.73)}
\end{aligned}
$$

Line	Justification
(C.66)	Writing out $N(x_0, z_0; \mathbf{x}, \mathbf{z})$ as in (C.47), and $E[N(x_0, z_0; \mathbf{x}, \mathbf{z})] \simeq p(x_0, z_0)$.
(C.67)	$p(x_0, z_0) = \sum_y p(x_0, y, z_0)$; and the expectation is over $\mathbf{y}, \tilde{S}, \bar{S}$ since the only dependence on $\mathbf{x}$ is through the sets $\tilde{S}, \bar{S}$ and $\mathbf{z} = f^n(\mathbf{y})$ deterministically.
(C.68)	Iterated expectations.
(C.69)	The conditioning on $\tilde{S}$, $\bar{S}$ can be dropped because the joint statistics of $(\mathbf{y}, \mathbf{z})$ are fully determined by $\mathbf{y}$, the sets $\tilde{S}$ and $\bar{S}$ are relevant only for knowing which indices to sum over.
(C.69)	Definition of $\hat{p}$.
(C.70)	The expectation over the sets $\tilde{S}$, $\bar{S}$ turns the sum over $\hat{p}$ into a shift average. Hence we get the shift average given by (C.21) times the sizes of the sets, e.g., $|S(x_0, \tilde{y})| \leq (np(x_0, \tilde{y}) + n\delta |\mathcal{X}\mathcal{Y}|^{-1})$.
(C.71)	$p(x, y, z) = p(x, y) p(z\|y)$.
(C.72)	$\bar{d} = \max\{\delta_{xy}, \kappa_2\}$.

■ C.4.5 Mean-Square Convergence

From Chebychev's inequality we now have

$$
\begin{aligned}
& \lim_{n\to\infty} \Pr\left[(|N(x_0, z_0; \mathbf{x}, \mathbf{z}) - np(x_0, z_0)|) \geq n\delta |\mathcal{X}\mathcal{Z}|^{-1} \right] \\
& \leq \quad \lim \operatorname{var} \frac{N(x_0, z_0; \mathbf{x}, \mathbf{z})}{(n\delta |\mathcal{X}\mathcal{Z}|^{-1})^2} \leq \lim \frac{n^2 \bar{\delta} |\text{const}|}{n^2 \delta_{xz}^2 |\mathcal{X}\mathcal{Z}|^{-1}} \qquad \text{(C.74)} \\
& = \quad \lim_{n\to\infty} \frac{\bar{\delta}}{\delta_{xz}^2} |\text{const}| |\mathcal{X}\mathcal{Z}| \qquad \text{(C.75)}
\end{aligned}
$$

We can make the limit (C.75) converge to 0 by picking $\bar{\delta}$ and δ_{xz} to be functions of n, per Section C.2, such that $\lim_{n\to\infty} \frac{\bar{\delta}}{(\delta_{xz})^2} = 0$. This mean-squared convergence holds for any pair of sample values (x_0, z_0). Therefore, for any ϵ we can find a n_0 such that for all $n > n_0$ the $\Pr\left[(\mathbf{x}, \mathbf{z}) \notin T^n_{x,z}(\epsilon)\right] < \epsilon$.

Appendix D

Derivations: CEO Problems

D.1 Serial CEO Problem

In this section we derive the results for the serial CEO problem presented in Chapter 4. We first derive a general achievability region, and then specialize to the quadratic-Gaussian case.

D.1.1 Achievability

In this section we demonstrate our successive coding approach to the serial CEO problem. Agent l has a source estimate $\hat{\mathbf{x}}_l$ such that $(\hat{\mathbf{x}}_l, \mathbf{x}) \in T^n_{\mathsf{x},\hat{\mathsf{x}}_l}(\epsilon)$. Since $\hat{\mathbf{x}}$ is marginally typical we can treat it as a new source vector, and transcode it into a new random codebook, and communicate the appropriate index m to Agent $l+1$. Agent $l+1$ receives m and in addition has side information $\mathbf{y}$ generated from $\mathbf{x}$ through the memoryless channel law $p(y|x)$.

- **Codebooks:** Associate $p(u_l|\hat{x}_l)$ with agent l. Let $\tilde{R}_l = I(\hat{\mathsf{x}}_l; \mathsf{u}_l) + \epsilon$ and $R_l = I(\hat{\mathsf{x}}_l; \mathsf{u}_l) - I(\mathsf{u}_l; \mathsf{y}_{l+1}) + 2\epsilon$. Construct a random codebook $\mathcal{C}_l$ with $2^{n\tilde{R}_l}$ codewords $\mathbf{u}(s_l)$ each generated in an i.i.d. manner $p_{\mathbf{u}_l}(\mathbf{u}_l(s_l)) = \prod_{i=1}^n p_{\mathsf{u}_l}(u_{l,i}(s_l))$. Label these codewords $\mathbf{u}_l(s_l)$ where $s_l \in \mathcal{S}_l = \{1, 2, \ldots, 2^{n\tilde{R}_l}\}$. Subdivide $\mathcal{C}_l$ into 2^{nR_l} subcodes or "bins". Assign the first $2^{n(I(\mathsf{u}_l;\mathsf{y}_{l+1})-\epsilon)}$ codewords to bin 1, the next $2^{n(I(\mathsf{u}_l;\mathsf{y}_{l+1})+\epsilon)}$ to bin 2, and so on up to bin 2^{nR_l}. Let $B_l(i)$ denote the codewords assigned to agent l's ith bin.

- **Encoding:** Use the dither encoding rule of Sec. C.3 to map $\hat{\mathbf{x}}_l$ to a codeword $\mathbf{u}_l(s_l) \in \mathcal{C}_l$. Send the message m such that $\mathbf{u}_l(s_l) \in B(m)$. If there is no such codeword set $m = 0$. If there is more than one, pick any one.

- **Decoding:** Agent $l+1$ searches the bin $B(m)$ for a s_l such that $(\mathbf{u}_l(s_l), \mathbf{y}_{l+1}) \in T^n_{\mathsf{u}_l,\mathsf{y}_{l+1}}(\epsilon)$. If there is a unique satisfactory s_l, the CEO calculates $\hat{\mathbf{x}}_{l+1} = g_{l+1}(\mathbf{u}_l(s_l), \mathbf{y}_{l+1})$. If there is not a satisfactory s_l, the CEO declares an error.

- **Probability of Error:**

  1. The sequences $(\mathbf{x}, \hat{\mathbf{x}}_l) \notin T^n_{\mathsf{x},\hat{\mathsf{x}}_l}(\epsilon)$. The probability of this is small by assumption.

2. The sequence $(\mathbf{x}, \mathbf{u}_l(s_l)) \notin T^n_{x,u_l}(\epsilon)$. The probability of this event is small by the Serial Markov Lemma.
3. The sequence $(\mathbf{u}_l(s_l), \mathbf{y}_{l+1}) \notin T^n_{u_l,y_{l+1}}(\epsilon)$. If $u \leftrightarrow x \leftrightarrow y_l$, the probability of this event is small by the regular Markov Lemma since $(\mathbf{x}, \mathbf{u}_l(s_l)) \in T^n_{x,u_l}(\epsilon)$ by step 2 and $(\mathbf{x}, \mathbf{y}_{l+1})$ are pairwise i.i.d.
4. There exists a $\tilde{s} \neq s_l$ such that $\mathbf{u}_l(\tilde{s}) \in B(m)$ but such that $(\mathbf{y}_{l+1}, \mathbf{u}(\tilde{s})) \notin T^n_{y_{l+1},u_l}(\epsilon)$, yet $(\mathbf{u}_l(s_l), \mathbf{y}_{l+1}) \in T^n_{u_l,y_{l+1}}(\epsilon)$.

   $$\Pr \leq 2^{n(\tilde{R}_l - R_l)} 2^{-n(I(u_l;y_{l+1}))} = 2^{-n\epsilon}.$$

5. Given that $\mathbf{u}(s_l)$ is decoded correctly, the empirical distribution can be made as close as we want to the chosen distribution $p(x, u_l, y_{l+1})$. This implies that we achieve a distortion $d_{l+1} = E\left[D(x, g_{l+1}(u_{l+1}, y_{l+1}))\right]$.
   the distortion constraint is met.

D.1.2 Quadratic-Gaussian Case

At agent $l-1$'s encoder, $\hat{x}_{l-1}$ and x are jointly typical by assumption that at all decoding and data fusion steps were all accomplised without error earlier in the chain of agents. In addition, whether we use the dither encoding rule or VB Codes, the marginals approximate the test channel. Therefore, we can use an innovations form to rewrite the relationship between $\hat{x}_{l-1}$ and x as $\hat{x}_{l-1} = \alpha x + \tilde{v}_{l-1}$, where $\alpha = \left(1 - \frac{d_{l-1}}{\sigma_x^2}\right)$ and $\tilde{v}_{l-1} \sim \mathcal{N}(0, \alpha d_{l-1})$. For the purpose of encoding, define agent $l-1$'s source observation to be

$$z_{l-1} = \frac{\hat{x}_{l-1}}{\alpha} = x + \frac{\tilde{v}_{l-1}}{\alpha}. \tag{D.2}$$

Think of agent $l - l$'s observation z_{l-1} as the source in additive white Gaussian noise, $\frac{1}{\alpha}\tilde{v}_{l-1}$ of variance $\frac{\sigma_x^2 d_{l-1}}{\sigma_x^2 - d_{l-1}}$. Then consider of agent l's observation y_l as decoder side information. This is the noisy Wyner-Ziv problem. Now, rewrite the noisy Wyner-Ziv rate distortion function (3.5) in the distortion-rate form

$$d = \sigma^2_{x|y_0,y_1} + (\sigma^2_{x|y_0} - \sigma^2_{x|y_0,y_1}) 2^{-2R}. \tag{D.3}$$

In the current context the encoder observation $y_1 = z_{l-1}$, the decoder side information, $y_0 = y_l$, the rate $R = R_{l-1}$, and the distortion $d = d_l$. Substituting these values into (D.3) results in

$$d_l = \sigma^2_{y_l,z_{l-1}} + (\sigma^2_{x|y_l} - \sigma^2_{x|y_l,z_{l-1}}) 2^{-2R_{l-1}}. \tag{D.4}$$

To simplify the result, we make the following calculations:

$$\sigma^2_{y_l,z_{l-1}} = \frac{1}{\frac{\sigma_x^2 - d_{l-1}}{\sigma_x^2 d_{l-1}} + \frac{1}{N_l} + \frac{1}{\sigma_x^2}} = \frac{N_l d_{l-1}}{N_l + d_{l-1}} \tag{D.5}$$

$$\begin{aligned} \sigma^2_{x|y_l} - \sigma^2_{y_l,z_{l-1}} &= \frac{N_l \sigma_x^2}{N_l + \sigma_x^2} - \frac{N_l d_{l-1}}{N_l + d_{l-1}} = \frac{N_l \sigma_x^2}{N_l + \sigma_x^2}\left[\frac{1 - \frac{d_{l-1}}{\sigma_x^2}}{1 + \frac{d_{l-1}}{N_l}}\right] \\ &= \sigma^2_{x|y_l}\left[\frac{1 - \frac{d_{l-1}}{\sigma_x^2}}{1 + \frac{d_{l-1}}{N_l}}\right]. \end{aligned} \tag{D.6}$$

Substituting (D.5) and (D.6) into (D.4) gives

$$d_l = \frac{N_l\, d_{l-1}}{N_l + d_{l-1}} + \sigma^2_{x|y_l} \frac{\left(1 - \frac{d_{l-1}}{\sigma_x^2}\right)}{\left(1 + \frac{d_{l-1}}{N_l}\right)} 2^{-2R_{l-1}},$$

which is the iterative distortion-rate function for the serial CEO problem presented in (4.7).

D.2 Parallel CEO Problem

In this section we derive the results for the parallel CEO problem presented in Chapter 4. We first derive a general achievability region, and then specialize to the quadratic-Gaussian case.

D.2.1 Achievability

We now justify our sequential approach to the regular CEO problem. Let the source $\mathbf{x}$ be i.i.d., $p_{\mathbf{x}}(\mathbf{x}) = \prod_{i=1}^n p_x(x_i)$ let $d_l = E\left[D(\mathbf{x}, \hat{\mathbf{x}}_l)\right]$ be the average distortion measure that we want to minimize. Let agents $1, \ldots, L$ have observations $\mathbf{y}_1, \ldots, \mathbf{y}_L$ jointly distributed with the source as $p_{\mathbf{x},\mathbf{y}_1,\ldots,\mathbf{y}_L}(\mathbf{x}, \mathbf{y}_1, \ldots, \mathbf{y}_L) = \prod_{i=1}^n p_x(x_i) \prod_{l=1}^L p_{y_l|x}(y_{l,i}|x_i)$.[1]

- **Codebooks:** Associate codebook $\mathcal{C}_l$ and test channel $p(u_l|y_l)$ with agent l. Let $\tilde{R}_l = I(y_l; u_l) + \epsilon/L$ be the rate of $\mathcal{C}_l$ and define $R_l = I(y_l; u_l) - I(u_l; u_1^{l-1}) + \epsilon$. In [40, 45] the authors show the Generalized Markov Lemma which states that conditionally independent source observations $\mathbf{y}_1, \ldots, \mathbf{y}_L$, of the sort we have here, can be independently encoded into codewords $\mathbf{u}_1, \ldots, \mathbf{u}_L$ that are jointly typical with each other and the source according to $p(x, u_1, \ldots, u_L)$. In [40] Han and Kobayashi show this for finite-alphabet sources and in [45] Oohama extends their result to Gaussian sources. We assume that codebooks $\mathcal{C}_1, \ldots \mathcal{C}_L$ are generated in an appropriate manner so that the Generalized Markov Lemma holds.

---

[1] If the CEO has its own source observation $\mathbf{y}_{\text{CEO}}$, this can be accounted for through an $L+1$st agent who has no rate constraint, i.e., $R_{L+1} = \infty$, and so can communicate $\mathbf{y}_{\text{CEO}}$ to the CEO losslessly.

Next, bin the codewords of each codebook. For example, consider the codewords $\mathbf{u}_l(s_l) \in \mathcal{C}_l$ at agent l where $s_l \in \mathcal{S}_l = \{1, 2, \ldots, 2^{n\tilde{R}_l}\}$. Assign the first $2^{nI(u_l;u_1^{l-1})}$ codewords to bin 1, the next $2^{nI(u_l;u_1^{l-1})}$ to bin 2, and so on up to bin 2^{nR_l}. Let $B_l(m)$ denote the set of codewords assigned to Agent l's mth bin.

- **Encoding:** Given an observation $\mathbf{y}_l$, agent l encodes it into the appropriate sequence $\mathbf{u}_l(s_l) \in \mathcal{C}_l$. If there is no such codeword set $s_l = 0$. If there is more than one, pick any one. The encoder transmits to the CEO the bin index m such that $\mathbf{u}_l(s_l) \in B_l(m)$.

- **Decoding:** The decoding is done by the CEO in L steps starting with agent 1. Note that for agent 1 there is only one codeword per bin (since $I(u_1; u_1^0) = 0$), so the CEO simply maps the received index back into a codeword sequence, as in a regular source code. At step $l > 1$ the CEO looks for a $\mathbf{u}_l(s_l)$ such that $s_l \in B_l(m)$ and $(\mathbf{u}_1(s_1), \ldots, \mathbf{u}_{l-1}(s_{l-1}), \ldots, \mathbf{u}_l(s_l)) \in T^n_{u_1,u_2,u_l}(\epsilon)$. If there is a unique satisfactory s_l, the CEO calculates $\hat{\mathbf{x}}_l = g_l(\mathbf{u}_1(s_1), \ldots, \mathbf{u}_l(s_l))$. If there is not a satisfactory s_l, the CEO declares an error.

- **Probability of Error:**

  1. The set of sequences $(\mathbf{x}, \mathbf{y}_1, \ldots \mathbf{y}_L) \notin T^n_{x,y_1,\ldots,y_L}(\epsilon)$. The probability of this event is small by the weak law of large numbers.
  2. The sequence $\mathbf{y}_l$ is typical, but there does not exist a s_l such that $(\mathbf{y}_l, \mathbf{u}_l(s_l)) \in T^n_{y_l,u_l}(\epsilon)$. The probability of this event is small if $\tilde{R}_l > I(y_l; u_l)$.
  3. The sequences $(\mathbf{x}, \mathbf{u}_1(s_1), \mathbf{u}_2(s_2), \ldots, \mathbf{u}(s_l)) \notin T^n_{x,u_1,\ldots,u_l}(\epsilon)$. The probability of this event is small for all l by the Generalized Markov Lemma.
  4. There exists another $\mathbf{u}(\tilde{s}_l)$ with the same bin index as the correct $\mathbf{u}(s_l)$, but such that $(\mathbf{y}_l, \mathbf{u}(\tilde{s}_l)) \notin T^n_{y_l,u_l}(\epsilon)$, yet $(\mathbf{x}, \mathbf{u}(s_1), \mathbf{u}(s_2), \ldots, \mathbf{u}(\tilde{s}_l)) \in A_\epsilon^{*(n)}$.

  $$\Pr \le 2^{n(\tilde{R}_l - R_l)} 2^{-n(I(u_l;u_1^{l-1})+\epsilon)} = 2^{-n\epsilon/l}.$$

  5. Given indices $s_1, s_2, \ldots s_l$ are decoded correctly, the empirical distribution is close to the original distribution $p(x)\prod_{i=1}^{l} p(u_i|x)$, and hence $(\mathbf{x}, \hat{\mathbf{x}}_l)$ will have a joint distribution close to the distribution that achieves distortion d_l.

- **Iterated Statistics:** We now show that in the quadratic-Gaussian case $I(u_l; u_1^{l-1}) = I(u_l; \hat{x}_{l-1})$. This fact is important because it allows us to derive the achievable distortion-rate region in the quadratic-Gaussian case in an iterative manner. This means that the generation of single-letter statistics can be done sequentially. This helps make the analysis of the finite-L region more tractable.

First, assume that the estimate $\hat{x}$ of x can be defined recursively recursively, i.e., $\hat{x}_{l-1} = g_{l-1}(u_{l-1}, g_{l-2}(u_{l-2}, \ldots))$, but overall is a deterministic function of $u_1, \ldots u_{l-1}$. For the moment, assume that $\hat{x}_{l-1}$ is the minimum mean-squared error estimate of x given $\{u_1, u_2, \ldots, u_{l-1}\}$. This assumption will be justified below. We have the following Markov chain relationships,

$$\begin{array}{ccccccc} u_l & \leftrightarrow & u_1^{l-1} & \leftrightarrow & \hat{x}_{l-1}, & & \end{array} \tag{D.9}$$

$$\begin{array}{ccccccc} x & \leftrightarrow & \hat{x}_{l-1} & \leftrightarrow & u_1^{l-1}, & & \end{array} \tag{D.10}$$

$$\begin{array}{ccccccc} u_l & \leftrightarrow & x & \leftrightarrow & \hat{x}_{l-1} & \leftrightarrow & u_1^{l-1}, \end{array} \tag{D.11}$$

$$\begin{array}{ccccccc} u_l & \leftrightarrow & \hat{x}_{l-1} & \leftrightarrow & u_1^{l-1}. & & \end{array} \tag{D.12}$$

Eq. (D.9) holds because $\hat{x}_{l-1}$ is a deterministic function of $\{u_1, \ldots, u_{l-1}\}$. Equation (D.10) holds by the orthogonality properties of the minimum mean-squared error estimator, i.e. the error is independent of the data so $p(x|u_1^{l-1}) = p(x|\hat{x}_{l-1}, u_1^{l-1}) = p(x - \hat{x}_{l-1}|\hat{x}_{l-1}, u_1^{l-1}) = p(x - \hat{x}_{l-1}|\hat{x}_{l-1}) = p(x|\hat{x}_{l-1})$. Since u_l is only dependent on y_l which, conditioned on x, is independent of y_1^{l-1} and therefore of $(u_1^{l-1}, \hat{x}_{l-1})$, we put this fact together with (D.10) to get (D.11). We can marginalizing out x in (D.11) to get (D.12) because $\sum_x p(x, \hat{x}_{l-1}, u_1, \ldots u_l) = \sum_x p(x, \hat{x}_{l-1}, u_l) p(u_1, \ldots, u_{l-1}|\hat{x}_{l-1}) = p(u|\hat{x}_{l-1}) p(\hat{x}_{l-1}) p(u_1, \ldots, u_{l-1}|\hat{x}_{l-1})$. Finally, putting together (D.9) and (D.12) with the data processing inequality tells us that

$$I(u_l; u_1^{l-1}) = I(u_l; \hat{x}_{l-1}). \tag{D.13}$$

We now confirm that $\hat{x}_{l-1}$ is the minimum mean-squared error estimate of x given $\{u_1, u_2, \ldots, u_{l-1}\}$. What we need to show is that the minimum mean-squared error estimate $\hat{x}_l$ can be constructed iteratively as $\hat{x}_l = g_l(u_l, \hat{x}_{l-1})$. The proof is inductive. We know that $\hat{x}_1$ is the minimum mean-squared error estimate since there is only 1 bin and $\hat{x}_1 = u_1$. Assume that $\hat{x}_{l-1} = g_{l-1}(u_{l-1}, g_{l-2}(u_{l-2}, g_{l-3}(u_{l-3}, \ldots)))$ is the minimum mean-squared error estimate of x given $u_1, \ldots u_{l-1}$. Then we have

$$\begin{aligned} p(x|u_1^{l-1}, u_l) &= p(x|u_1^{l-1}, u_l, \hat{x}_{l-1}) && \text{(D.14)} \\ &= \frac{p(x, u_1^{l-1}, u_l, \hat{x}_{l-1})}{p(u_1^{l-1}, u_l, \hat{x}_{l-1})} \\ &= \frac{p(u_l|x, u_1^{l-1}, \hat{x}_{l-1}) p(x|u_1^{l-1}, \hat{x}_{l-1})}{p(u_l|u_1^{l-1}, \hat{x}_{l-1})} \\ &= \frac{p(u_l|x, \hat{x}_{l-1}) p(x|\hat{x}_{l-1})}{p(u_l|\hat{x}_{l-1})} && \text{(D.15)} \\ &= \frac{p(u_l|x, \hat{x}_{l-1}) p(\hat{x}_{l-1}, x)}{p(u_l, \hat{x}_{l-1})} \\ &= p(x|\hat{x}_{l-1}, u_l). \end{aligned}$$

Eq. (D.14) follows because $\hat{x}_{l-1}$ is a function of u_1^{l-1}, and (D.15) follows from the Markov relationships (D.10), (D.11), and (D.12).

D.2.2 Quadratic-Gaussian Case

When the decoder decodes the message from agent l he has the side information $\hat{x}$. We use innovations form to rewrite the relationship between $\hat{x}_{l-1}$ and x as $\hat{x}_{l-1} = \alpha x + \tilde{v}_{l-1}$, where $\alpha = (1 - d_{l-1}/\sigma_x^2)$ and $\tilde{v}_{l-1} \sim \mathcal{N}(0, \alpha d_{l-1})$. For the purpose of decoding, define the CEO's side information as

$$z_{l-1} = \frac{\hat{x}_{l-1}}{\alpha} = x + \frac{\tilde{v}_{l-1}}{\alpha}. \tag{D.16}$$

where $\frac{1}{\alpha}\tilde{v}_{l-1}$ has variance $\frac{\sigma_x^2 d_{l-1}}{\sigma_x^2 - d_{l-1}}$. Agent l is the encoder and measures $x + v_l$ where $v_l \sim \mathcal{N}(0, N_l)$. The CEO is the decoder and has side information given by z_{l-1}. This is a version of the noisy Wyner-Ziv problem. We again evaluate the distortion-rate form of the Wyner-Ziv function (D.3), but make different substitutions: the encoder observation $y_1 = y_l$, decoder side information $y_0 = z_{l-1}$, rate $R = R_l$, and distortion $d = d_{l-1}$. These substitutions result in

$$d_l = \sigma^2_{z_{l-1}, y_l} + (\sigma^2_{x|z_{l-1}} - \sigma^2_{x|z_{l-1}, y_l})2^{-2R_l}. \tag{D.17}$$

We have already calculated $\sigma^2_{z_{l-1}, y_l} = \frac{N_l d_{l-1}}{N_l + d_{l-1}}$ in (D.5), and $\sigma^2_{x|z_{l-1}} = d_{l-1}$ because z_{l-1} encapsulates the information the CEO knows before agent l reports in. Substituting these values into (D.17) gives

$$d_l = \frac{N_l \, d_{l-1}}{N_l + d_{l-1}} + \frac{d_{l-1}^2}{N_l + d_{l-1}} 2^{-2R_l}$$

which is the iterative distortion-rate function for the parallel CEO problem presented in (4.14).

D.3 Parallel CEO Problem with Large Numbers of Agents

- **Lower bounding (4.36)**
  To lower bound (4.36) we assume that $R_{l+1} > 0$ so that $x_{l+1} - x_l < 0$, i.e., the distortion decreases at each step. Rearranging (4.34) we get

$$\begin{aligned} -\frac{1 + x_l}{x_l^2}(x_{l+1} - x_l) &= 1 - e^{-2R_{l+1}} && \text{(D.19)} \\ \frac{1 + x_l}{x_l^2}\Delta x &= 1 - \left[1 - 2R_{l+1} + \frac{(2R_{l+1})^2}{2!} - \frac{(2R_{l+1})^3}{3!} + \ldots\right] && \text{(D.20)} \\ \frac{1 + x_l}{2x_l^2}|\Delta x| &\leq R_{l+1}, && \text{(D.21)} \end{aligned}$$

where we expand e^{-2R_l} in a power series in (D.20), and define $x_l - x_{l-1} = \Delta x \equiv$

$\frac{1}{L}\frac{\sigma_x^2-D}{N} > 0$.[2] For R_l small (e.g., $R_l < 1$), the higher order terms in the power series sum to a positive constant, yielding an upper bound (D.21). Summing (D.21) over l gives us

$$\sum_{l=0}^{L-1} R_{l+1} \geq \sum_{l=0}^{L-1} \frac{1+x_l}{2x_l^2}\Delta x, \tag{D.22}$$

where $x_0 = \frac{\sigma_x^2}{N}$ and $x_l = x_0 - l\Delta x = x_0 - l\left[\frac{\sigma_x^2-D}{LN}\right]$. This is a Riemann sum that lower-bounds the integral in (4.36).[3]

- **Upper bounding (4.36)**
  To upper bound (4.36) iterate (4.33) *backwards* through the same set of $\{x_0, x_1, \ldots, x_L\}$, starting now with $x_L = D/N$. All we are doing is upper bounding (4.36) by a second Riemann sum, but we can think of the backwards iteration as using a 'negative rate' $\tilde{R}_{l+1} < 0$ at each step. At each step we start with $x_{l+1} = x_0 - (l+1)\Delta x$ and use (D.19) to determine the 'negative rate' it takes to get to $x_l > x_{l+1}$.

$$\begin{aligned}
-\frac{1+x_{l+1}}{x_{l+1}^2}(x_l - x_{l+1}) &= 1 - e^{-2\tilde{R}_{l+1}} && \text{(D.23)}\\
-\frac{1+x_{l+1}}{x_{l+1}^2}\Delta x &= 1 - \left[1 - 2\tilde{R}_{l+1} + \frac{(2\tilde{R}_{l+1})^2}{2!} - \frac{(2\tilde{R}_{l+1})^3}{3!} + \ldots\right] \\
&= 1 - \left[1 + 2|\tilde{R}_{l+1}| + \frac{(2\tilde{R}_{l+1})^2}{2!} + \frac{(2|\tilde{R}_{l+1}|)^3}{3!} + \ldots\right] && \text{(D.24)}\\
&\leq -2|\tilde{R}_{l+1}| && \text{(D.25)}\\
\frac{1+x_{l+1}}{2x_{l+1}^2}\Delta x &\geq |\tilde{R}_{l+1}|, && \text{(D.26)}
\end{aligned}$$

where (D.24) follows because $\tilde{R}_{l+1} < 0$, and (D.25) by dropping the higher-order negative terms. Summing (D.26) over l we get

$$\sum_{l=0}^{L-1} |\tilde{R}_{l+1}| \leq \sum_{l=0}^{L-1} \frac{1+x_{l+1}}{2x_{l+1}^2}\Delta x, \tag{D.27}$$

Both (D.22) and (D.27) are Riemann sums approximating the integral of (4.36). The difference is that the height of the steps in the former are evaluated at the beginning of each interval, while those in the latter are evaluated at the end of each interval. Together these two approximations sandwich the integral in (4.36). We next relate R_{l+1} to $\tilde{R}_{l+1}$.

---

[2] Note that while in (4.35) we held $R_l = \bar{R}/L$ constant, we have now re-written (4.33) in terms of constant Δx. This will allow us to approximate (4.36) in terms of Riemann sums. Expressing (4.33) in this form is okay, as long as Δx is small. Clearly, for example, Δx must be less than σ_x^2/N. Since $\lim_{L\to\infty} \Delta x = 0$, and we are investigating the large L regime, we are in the small Δx region.

[3] The difference between the signs of the integrand in (4.36) and the summand in (D.22) occurs because $dx < 0$ in (4.36), but we chose $\Delta x > 0$ in (D.22).

- **Showing $R_{l+1} \leq |\tilde{R}_{l+1}| + O(\Delta x)$**
  To show that $R_{l+1} \leq |\tilde{R}_{l+1}| + O(\Delta x)$ think about decreasing the distortion from x_l to x_{l+1}, requiring rate R_{l+1}, and then increasing the distortion back from x_{l+1} to x_l, requiring 'negative rate' $\tilde{R}_{l+1}$.

  Solving (D.19) for R_{l+1} and (D.23) for $\tilde{R}_{l+1}$ gives us

$$\begin{aligned} R_{l+1} &= -\frac{1}{2}\log\left[1 - \frac{1+x_l}{x_l^2}\Delta x\right] && \text{(D.28)} \\ \tilde{R}_{l+1} &= -\frac{1}{2}\log\left[1 + \frac{1+x_{l+1}}{x_{l+1}^2}\Delta x\right] \leq 0. && \text{(D.29)} \end{aligned}$$

Subtract (D.28) from the absolute value of (D.29) to find the difference between the magnitude of these rates:

$$\begin{aligned} |\tilde{R}_{l+1}| - R_{l+1} &= \frac{1}{2}\log\left[\left(1 + \frac{1+x_{l+1}}{x_{l+1}^2}\Delta x\right)\left(1 - \frac{1+x_l}{x_l^2}\Delta x\right)\right] \\ &= \frac{1}{2}\log\left[1 - \frac{(\Delta x)^2}{x_l^2 x_{l+1}^2}\right] && \text{(D.30)} \\ &\geq \frac{1}{2}\log\left[1 - \frac{(\Delta x)^2}{(D/N)^4}\right] && \text{(D.31)} \\ R_{l+1} &\leq |\tilde{R}_{l+1}| + \frac{1}{2}\log\left[\frac{(D/N)^4}{(D/N)^4 - (\Delta x)^2}\right] \\ &\leq |\tilde{R}_{l+1}| + \frac{1}{2}\left(\frac{(\Delta x)^2}{(D/N)^4 - \Delta x^2}\right), && \text{(D.32)} \end{aligned}$$

where in (D.30) we use $x_l - x_{l+1} = \Delta x$, in (D.31) we use $x_l \geq D/N$, and in (D.32) we use $\log(s) \leq s - 1$. So, summing (D.32) over l gives us

$$\begin{aligned} \sum_{l=0}^{L-1} R_{l+1} &\leq \sum_{l=0}^{L-1} |\tilde{R}_{l+1}| + \frac{L}{2}\left(\frac{(\Delta x)^2}{(D/N)^4 - \Delta x^2}\right) \\ &= \sum_{l=0}^{L-1} |\tilde{R}_{l+1}| + \frac{1}{2}\left(\frac{\sigma_X^2 - D}{N\Delta x}\right)\left(\frac{(\Delta x)^2}{(D/N)^4 - \Delta x^2}\right) && \text{(D.33)} \\ &\leq \sum_{l=0}^{L-1} \frac{1+x_{l+1}}{2x_{l+1}^2}\Delta x + |\text{const}|\Delta x. && \text{(D.34)} \end{aligned}$$

In (D.33) we express L in terms of Δx, $L = \frac{1}{\Delta x}\frac{\sigma_X^2 - D}{N}$, and in (D.34) we substitute in (D.27) and explicitly showed that the second term is Δx times some positive constant.

- **Upper and Lower bounding $\bar{R} = \sum R_{l+1}$**
  Putting together (D.22) and (D.34) we can now upper and lower bound $\bar{R} = \sum R_{l+1}$:

  $$\sum_{l=0}^{L-1} \frac{1+x_l}{2x_l^2}\Delta x \leq \sum_{l=0}^{L-1} R_{l+1} =\leq \sum_{l=0}^{L-1} \frac{1+x_l}{2x_l^2}\Delta x + |\text{const}|\Delta x. \tag{D.35}$$

  Taking into account the $|\text{const}|\Delta x$ in (D.34), the two Riemann sums lower and upper bound $\int_{\frac{D}{N}}^{\frac{\sigma_X^2}{N}} \left(\frac{1}{2x^2} + \frac{1}{2x}\right) dx$, and converge as $L \to \infty$. Since they also lower and upper bound $\bar{R}$, in the limit $\bar{R}$ and the integral must be equal. This justifies (4.36), so the rate distortion bound (4.37) is achieved in the limit.

Thus, the sequential codes derived herein achieve the lower-bound on the rate distortion function of [45]. Note that the limits in (4.36) can be set arbitrarily, which means that the bound on the rate distortion function of [45] is not achievable only in the limit as $L \to \infty$, but at each data fusion step, as long as the extra data to be integrated at each step is asymptotically small.

List of Figures

1.1 An abbreviated diagram of the protocol stack in a traditional layered architecture, focusing on the layers relevant to the work in this thesis. . 14
1.2 A general sensor network with finite rate links and tree-structured communications. 15
1.3 Possible applications of the buffering and distribution protocols developed in this thesis include infrastructure gateway managers. An objective is to design algorithms that can adapt to unpredictable system loads caused by users entering and exiting the network, as well as by unpredictable traffic patterns. 18

2.1 Distributed source coding with two switches, (a) and (b). 22
2.2 Design intuition for scalar quantizers with decoder side information. Fig. 2.2-a plots the prior $p_{\mathsf{x}}(x)$ versus the quantization regions of a standard two-bit quantizer. Fig. 2.2-b plots the posterior $p_{\mathsf{x}|\mathsf{y}}(x|y_a)$ for the source x given the side information is y_a. Most of the posterior distribution is supported within the single quantization region labeled 0. Fig. 2.2-c plots two possible posteriors for x versus the quantization regions of a non-standard scalar quantizer. In this design the side information y is used to resolve the ambiguity in quantizer region. For example, if x was quantizer to 0, and we observed y_a we would guess that x is in the left-most region labeled 0. If, on the other hand y_b were measured, then we would guess that **x** is in the third region from the left labeled 0. . . . 23
2.3 Each choice of $p(u|y)$ and $f(\cdot,\cdot)$ defines one achievable rate-distortion point (d,R). The rate-distortion function $R(d)$ is the lower convex envelope of all achievable points and is indicated by the dashed curve. 27

2.4 The rate distortion function for Wyner-Ziv source coding with side information in the low distortion-to-noise ($d/\sigma^2_{x|y}$) regime can be found via sphere covering arguments. The dotted circle correspond to the source uncertainty given the side information, $\sigma^2_{x|y}$. The solid circles correspond to the radius-$\sqrt{nd}$ quantization regions at the encoder. The labeling of these quantization regions is not unique, e.g., two are labeled 'a', and two 'b' in the picture. 29
2.5 Channel coding with state side information at the encoder. 30
2.6 Capacity for information embedding region for a Gaussian i.i.d. host of variance σ^2_x can be found in the high distortion-to-noise d/N regime via sphere packing arguments. The inner dotted circle correspond to the power constraint, the source cannot be moved further than this without violating the distortion constraint. The outer circle indicates the extra distortion incurred by the host due to channel noise. The solid circles correspond the the embedding messages, which must be spaced at least $\sqrt{nN}$ apart to be immune to the channel noise. 32

3.1 Wyner-Ziv source coding with noisy encoder observations. The signals $\mathbf{x}$, $\mathbf{y}_0$, $\mathbf{y}_1$ and m are, respectively, the source, side information, encoder observation, and message. 36
3.2 The noisy Wyner-Ziv rate distortion function, $R^{\mathrm{WZ}}_{\mathrm{I}}(d)$, for the binary Hamming case is shown with the solid line. The function $g(d)$ is shown by the dotted curve, and the point $\tilde{d}$ is indicated by d_{tilde}. 39
3.3 The noisy Wyner-Ziv rate distortion function, $R^{\mathrm{WZ}}_{\mathrm{I}}(d)$, for the quadratic-Gaussian case. 40
3.4 Information embedding with noisy host observations. The signals $\mathbf{x}$, $\mathbf{y}$, m, $\mathbf{e}$, $\mathbf{w}$ and $\mathbf{z}$ are, respectively, the host, host observation, message, embedding signal, composite signal, and channel output. 41
3.5 The rate distortion function for noisy Wyner-Ziv source coding can be found via sphere covering arguments. The dotted circle correspond to the source uncertainty given the side information, $\sigma^2_{x|y}$. The solid circles correspond to the quantization regions at the encoder ($d - \sigma^2_{x|y_0,y_1}$) that must be scaled up by ($1 + N_1/\sigma^2_{x|y_0,y_1}$) because of the extra uncertainty at the encoder caused by the noisy observations. 44
3.6 The capacity for noisy information embedding can be derived from sphere packing arguments. The three concentric dotted circles correspond to: 1) uncertainty in state, $\sigma^2_{x|y}$, 2) state uncertainty + allowable introduced distortion, d, and 3) state uncertainty + distortion + channel noise, N_1. We cover the largest sphere with smaller (solid) spheres that correspond to codewords. Each small sphere is of radius $\sqrt{n(\sigma^2_{x|y} + N_1)}$ to ensure reliable decoding. 45

4.1 A general sensor network with finite-rate links and tree-structured communications. 48

4.2 The parallel CEO problem with additive noise observations. The signals $\mathbf{x}$, $\mathbf{y}_l$, and $\mathbf{v}_l$ are the source, observation, and noise vectors, respectively. Agent l communicates to the CEO at rate R_l. The CEO fuses the data to produce a source estimate $\hat{\mathbf{x}}$. Generally the CEO has its own observation observation $\mathbf{y}_{\text{CEO}}$. If the CEO does not have an observation, set $\mathbf{y}_{\text{CEO}} = 0$. 50

4.3 The serial CEO problem with additive noise observations. The signals $\mathbf{x}$, $\mathbf{y}_l$, and $\mathbf{v}_l$ are the source, observations, and noise vectors, respectively. Agent $l-1$ communicates to agent l at rate R_{l-1}. The CEO has his own observation $\mathbf{y}_{\text{CEO}}$. 51

4.4 The parallel CEO problem, quadratic-Gaussian case. Estimate-and-quantize performance is plotted with the dash-dotted curve. A (generally loose) lower bound given by $\sigma^2_{x|y_{\text{CEO}},y_1,\ldots,y_l}$ is plotted with the dashed curve. The performance of the successively structured codes proposed in Section 4.4 is plotted with the solid curve. 55

4.5 The serial CEO problem, quadratic-Gaussian case. Estimate-and-quantize performance is plotted with the dash-dotted curve. A (generally loose) lower bound given by $\sigma^2_{x|y_1,\ldots,y_l}$ is plotted with the dashed curve. The performance of the successively structured codes proposed in Section 4.4 is plotted with the solid curve. 56

4.6 The serial CEO problem can visualized as a succession of L noisy Wyner-Ziv stages. At stage l, $\hat{x}_{l-1}$ is the encoder observation, and y_l is the side information. 57

4.7 The encoding and successive decoding algorithms for the parallel CEO problem can be visualized as a succession of L noisy Wyner-Ziv stages. At stage l, y_l is the encoder observation, and $\hat{x}_{l-1}$ is the side information. In this diagram we have not indicated the CEO's direct observation $\mathbf{y}_{\text{CEO}}$. 58

4.8 Serial CEO estimation error d_l improves with agent number. In this example $\sigma^2_x = 4$, $N = 1$, and $R = 1, 2, 3, \infty$. Solid curves plot the evolution of the estimation error d_l for the different finite rates. Dotted lines plot the lower bound on performance for each finite rate, approached asymptotically as the number of agents l tends toward infinity. The dashed curve is $\sigma^2_{x|y_1,\ldots,y_l}$ the minimum mean-squared error in x given $y_1, \ldots, y_l$ which can be approached as the per-agent rate R grows toward infinity. 60

4.9 Parallel CEO estimation performance, d_l, improves with agent number. In this example $\sigma^2_x = 4$, $N = 1$, and $R = 0.25, 0.75, 1.5, \infty$. Solid curves plot the evolution of the estimation error d_l for the different finite rates. The dashed curve plots $\sigma^2_{x|y_{\text{CEO}},y_1,\ldots,y_l}$, approached as R grows to infinity. 63

4.10 Normalized distortion dynamics as a function of R_l. 65

4.11 The fraction of the total rate $R_{\text{tot}} = \bar{R}$ assigned to each agent is a function of both R_{tot} and SNR. 68
4.12 The percentage of extra distortion incurred by constraining the agents to equal rates. For the three examples presented $\bar{R} = 10$, and the SNRs of the L agents were identical within each example. 70
4.13 Certain classes of relay channel communication problems can be approached via the parallel CEO solution by wrapping a channel encoder and decoder around the distributed source estimator. 72

5.1 Internal structure of memory interface. 77
5.2 We designed the memory interface of a basic RAM of size M_{tot} to manage the buffering and distribution of distortion-tolerant data. Each signal arrival is quantized by the interface and stored in the RAM. Each departure is a packet of size M_{pac}. The arrival and departure statistics are modeled as a $M/M/1$ queuing process with arrival rate λ and departure rate μ_{pac}. 78
5.3 Bound on achievable average distortion for $d = \exp(-0.1R)$ as a function of system utilization. 80
5.4 Basic idea of baseline protocol. Information is stored at constant fidelity. If new signals arrive when the queue is full they are lost. Transmissions should focus on sending the stored signals one at a time. 82
5.5 Basic idea of adaptive protocol. Signals are stored at variable fidelities. If a new signal arrives when the queue is full, all signals are "squeezed" to fit at lower fidelities. Transmissions should send first the most important information stored in memory, determined across all stored signals. . . . 83
5.6 Determine packet contents by "water-filling" to the dashed line, which satisfies $\sum_{i=1}^{m-1} \delta_{\text{pac},i} = M_{\text{pac}}$. Shaded rectangles indicate $R_{\text{dec},i}$ and white rectangles $R_{\text{qu},i}$. 85
5.7 Determine memory re-allocations by "water-filling" to the dashed line, which satisfies $\sum_{i=1}^{m-1}(R_{\text{qu},i} - \delta_{\text{qu},i}) = M_{\text{tot}}$. Shaded rectangles indicate $R_{\text{dec},i}$ and white rectangles $R_{\text{qu},i}$. 86
5.8 The Markov chain denoting the number of items in memory. 88
5.9 Expected distortion versus ρ_{pac}, $M_{\text{tot}} = 1200$, $M_{\text{pac}} = 30$, and $\alpha = 0.1$. The expected distortion for the low-rate ($M_{\text{tot}}/\kappa_{\text{mem}} = 1200/200 = 6$) and high-rate ($M_{\text{tot}}/\kappa_{\text{mem}} = 1200/80 = 15$) quantization cases are plotted as dotted and dash-dotted curves. Experimental confirmation of the analysis is plotted as +'s and ×'s, respectively. The choices of $\kappa_{\text{mem}} = 80, 200$ are optimal for $\rho_{\text{pac}} \simeq 5$ and $\rho_{\text{pac}} \simeq 2$, respectively. The performance of the optimized baseline and the lower bound on all algorithms are plotted as dashed and solid curves, respectively. The experimental performance of the adaptive algorithm, indicated by o's, closely approximates the lower bound. 91

5.10 The percentage by which system resources must increase so that baseline algorithm performance is guaranteed to match adaptive algorithm performance for fixed $M_{\text{tot}} = 1200$, $\tau_{\text{emp}} = 40$, $\alpha = 0.1$, and $\kappa_{\text{mem}} = 200$ (low quantization rate) and $\kappa_{\text{mem}} = 80$ (high quantization rate). The transition from memory-constrained to communication-constrained operation occurs at $\rho_{\text{sig}} \simeq 1$. 96

5.11 The percentage by which system resources must increase so that baseline algorithm performance is guaranteed to match adaptive algorithm performance for fixed $M_{\text{tot}} = 1200$, $\tau_{\text{emp}} = 40$, $\alpha_{\text{base}} = 0.1$, $\alpha_{\text{adapt}} = 0.09$ and $\kappa_{\text{mem}} = 200$ (low quantization rate) and $\kappa_{\text{mem}} = 80$ (high quantization rate). 99

B.1 Channel coding with noisy side information, Gaussian case. The signal y is viewed as the perfectly known channel state. The random quantities m, $\mathbf{e}$, $\mathbf{w}$, and $\mathbf{z}$ are respectively the message, embedding signal, composite signal, and channel output. The two unknown sources of noise are $\tilde{\mathbf{v}}_0$ and $\mathbf{v}_1$ which are the estimation error and channel noise, respectively. 121

List of Tables

2.1 Relationship between problems and parameters of Fig. 2.1. Column 2 indicates whether the y-encoder can losslessly communicate **y** to the decoder ($R_y \geq H(y)$). This is equivalent to the decoder observing **y** directly. If $R_y \leq H(y)$ the y-encoder cannot losslessly communicate **y** to the decoder. In the first two problems there is no **y** to encode. Columns 3 and 4 indicate whether the respective switches are open (○) or closed (●). Columns 5 and 6 indicate the distortion requirements on the decoded source estimates. A non-zero value indicates a rate distortion (lossy) problem while $d_y = \infty$ means that the decoder is not concerned with estimating **y** in this problem. 22

Notation

Symbol	Definition
x	random variable (sans-serif)
x	sample value (serifed)
$\boldsymbol{\mathsf{x}}$	random vector (bold sans-serif)
$\mathbf{x}$	sample vector (bold serifed)
x_i	ith component of $\boldsymbol{\mathsf{x}}$
$\boldsymbol{\mathsf{x}}_i^j$	subvector consisting of ith through jth components of $\boldsymbol{\mathsf{x}}$
$\boldsymbol{\mathsf{x}}^j$	subvector consisting of 1st through jth components of $\boldsymbol{\mathsf{x}}$
$\boldsymbol{\mathsf{x}}_i$	ith random vector in an indexed set, e.g., $\{\boldsymbol{\mathsf{x}}_1, \boldsymbol{\mathsf{x}}_2, \ldots, \boldsymbol{\mathsf{x}}_L\}$
$\mathcal{X}$	domain of random variable x
$p_{\mathsf{x}}(x)$	probability distribution of x
$p(x)$	probability distribution of x (distribution supressed)
$H(\mathsf{x})$	entropy of x
$H(\mathsf{x}\|\mathsf{y})$	conditional entropy of x given y
$h(\mathsf{x})$	differential entropy of x
$h(\mathsf{x}\|\mathsf{y})$	conditional differential entropy of x given y
$I(\mathsf{x};\mathsf{y})$	mutual information between x and y
$I(\mathsf{x};\mathsf{y}\|\mathsf{z})$	conditional mutual information between x and y given z
$N(x_0;\mathbf{x})$	cardinality of number of indices $\{i, 1 \leq i \leq n\}$ such that $x_i = x_0$
$T^n_{p_{\mathsf{x}}(x)}(\epsilon)$	strongly typical set: $T^n_{\mathsf{x}}(\epsilon) = \{\mathbf{x} : \|N(x_0;\mathbf{x}) - np_{\mathsf{x}}(x_0)\| < n\epsilon\}$ for all $x_0 \in \mathcal{X}$ s.t. $p_{\mathsf{x}}(x_0) > 0$ and $N(x_0;\mathbf{x}) = 0$ if $p_{\mathsf{x}}(x_0) = 0$
$T^n_{\mathsf{x}}(\epsilon)$	strongly typical set (distribution of x supressed)
$\mathcal{T}(\boldsymbol{\mathsf{x}})$	empirical distribution (type) of $\boldsymbol{\mathsf{x}}$
log	base-2 or base-e, as indicated in the text

Bibliography

[1] R. Ahlswede and I. Csiszár, "Hypothesis testing with communication constraints," *IEEE Trans. Inform. Theory*, vol. 32, pp. 533–542, July 1986.

[2] R. Ahlswede and J. Körner, "Source coding with side information and a converse for degraded broadcast channels," *IEEE Trans. Inform. Theory*, vol. 21, pp. 629–637, Nov. 1975.

[3] A. Albanese, J. Blömer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1737–1744, Nov. 1996.

[4] R. J. Barron, *Systematic Hybrid Analog/Digital Signal Coding.* PhD thesis, Mass. Instit. of Tech., 2000.

[5] R. J. Barron, B. Chen, and G. W. Wornell, "The duality between information embedding and source coding with side information and its implications and applications," *Submitted to IEEE Trans. Info. Theory*, Jan. 2000.

[6] R. J. Barron, B. Chen, and G. W. Wornell, "The duality between information embedding and source coding with side information and some applications," in *Proc. Int. Symp. Inform. Theory*, (Washington, DC), p. 300, June 2001.

[7] T. Berger, *Rate Distortion Theory.* Prentice-Hall, 1971.

[8] T. Berger, "Multiterminal source coding," in *The Information Theory Approach to Communications* (G. Longo, ed.), ch. 4, Springer-Verlag, 1977.

[9] T. Berger, Z. Zhang, and H. Viswanathan, "The CEO problem," *IEEE Trans. Inform. Theory*, vol. 42, pp. 887–902, May 1996.

[10] R. E. Blahut, *Principles and Practice of Information Theory.* Addison-Wesley, 1987.

[11] R. S. Blum, S. A. Kassam, and H. V. Poor, "Distributed detection with multiple sensors: Part II – Advanced topics," *Proc. IEEE*, vol. 85, pp. 64–79, Jan. 1997.

[12] G. Caire and S. Shamai, "On achievable rates in a multi-antenna Gaussian broadcast channel," in *Proc. Int. Symp. Inform. Theory*, (Washington, DC), p. 147, June 2001.

[13] B. Chen, S. C. Draper, and G. W. Wornell, "Information embedding and related problems: Recent results and applications," in *Proc. 39th Allerton Conf. on Communication, Control and Computing*, Oct. 2001.

[14] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1423–1443, May 2001.

[15] M. Chiang and T. M. Cover, "Unified duality of channel capacity and rate distortion with state information," in *Proc. Int. Symp. Inform. Theory*, (Washington, DC), p. 300, June 2001.

[16] J. Chou, S. S. Pradhan, L. El Ghaoui, and K. Ramchandran, "A robust optimization solution to the data hiding problem using distributed source coding principles," in *Proc. SPIE Conf. on Elec. Imaging: Image and Video Comm. and Proc.*, pp. 301–310, Jan. 2000.

[17] J. Chou, S. S. Pradhan, and K. Ramchandran, "On the duality between source coding and data hiding," in *Proc. 33rd Asilomar Conf. on Signals, Systems and Comp.*, Nov. 1999.

[18] A. Cohen, S. C. Draper, E. Martinian, and G. W. Wornell, "Stealing bits from a quantized source," in *Proc. Int. Symp. Inform. Theory*, (Lausanne, Switzerland), July 2002.

[19] A. S. Cohen and A. Lapidoth, "The Gaussian watermarking game," *IEEE Trans. Inform. Theory*, vol. 42, June 2002.

[20] M. H. Costa, "Writing on dirty paper," *IEEE Trans. Inform. Theory*, vol. 29, pp. 439–441, May 1983.

[21] T. M. Cover and A. El Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Inform. Theory*, vol. 25, pp. 572–584, Sept. 1979.

[22] T. M. Cover, A. El Gamal, and M. Salehi, "Multiple access channels with arbitrarily correlated sources," *IEEE Trans. Inform. Theory*, vol. 26, pp. 648–657, Nov. 1980.

[23] T. M. Cover and C. S. K. Leung, "An achievable rate region for the multiple-access channel with feedback," *IEEE Trans. Inform. Theory*, vol. 27, pp. 292–298, May 1981.

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

[25] R. L. Dobrushin and B. S. Tsybakov, "Information transmission with additional noise," *IEEE Trans. Inform. Theory*, vol. 8, pp. 293–304, 1962.

[26] A. A. El Gamal and T. M. Cover, "Achievable rates for multiple descriptions," *IEEE Trans. Inform. Theory*, vol. 28, pp. 851–857, Nov. 1982.

[27] W. H. Equitz and T. M. Cover, "Successive refinement of information," *IEEE Trans. Inform. Theory*, vol. 37, pp. 269–275, Mar. 1991.

[28] N. Farvardin and J. W. Modestino, "Adaptive buffer-instrumented entropy-coded quantizer performance for memoryless sources," *IEEE Trans. Inform. Theory*, vol. 32, pp. 9–22, Jan. 1986.

[29] R. G. Gallager, *Information Theory and Reliable Communication.* John Wiley and Sons, 1968.

[30] M. Gastpar, "Multiple-relay channel: Coding and asymptotic capacity." Wireless Networking Seminar, MIT, Cambridge, MA, Oct. 2001.

[31] S. I. Gel'fand and M. S. Pinsker, "Coding for channels with random parameters," *Problems of Control and Information Theory*, vol. 9, pp. 19–31, 1980.

[32] A. Gersho and R. Gray, *Vector Quantization and Signal Compression.* Kluwer Academic Press, 1992.

[33] V. K. Goyal and J. Kovacevic, "Generalized multiple description coding with correlating transforms," *IEEE Trans. Inform. Theory*, vol. 47, pp. 2199–2224, Sept. 2001.

[34] R. M. Gray, "Conditional rate-distortion theory," tech. rep., Stanford Electronics Laboratories, No. 6502-2, 1972.

[35] R. M. Gray, "A new class of lower bounds to information rates of stationary sources via conditional rate-distortion functions," *IEEE Trans. Inform. Theory*, vol. 19, pp. 480–489, 1973.

[36] R. M. Gray and A. D. Wyner, "Source coding for a simple network," *Bell Syst. Tech. J.*, vol. 53, pp. 1681–1721, Nov. 1974.

[37] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad-hoc wireless networks," *To appear IEEE/ACM Trans. Net.*

[38] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, pp. 388–404, 2000.

[39] T. S. Han and S. Amari, "Statistical inference under multiterminal data compression," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2300–2324, Oct. 1998.

[40] T. S. Han and S. Verdu, "A unified achievable rate region for a general class of multiterminal source coding systems," *IEEE Trans. Inform. Theory*, vol. 26, pp. 277–288, May 1980.

[41] D. Harrison and J. W. Modestino, "Analysis and further results on adaptive entropy-coded quantization," *IEEE Trans. Inform. Theory*, vol. 36, pp. 1069–1088, Sept. 1990.

[42] C. Heegard and A. El Gamal, "On the capacity of computer memory with defects," *IEEE Trans. Inform. Theory*, vol. 29, pp. 731–739, Sept. 1983.

[43] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *To appear: IEEE Trans. Wireless Comm.*

[44] F. R. Kschischang, B. J. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, pp. 498–519, Feb. 2001.

[45] Y. Oohama, "The rate-distortion fuction for the quadratic Gaussian CEO problem," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1057–1070, May 1998.

[46] Y. Oohama, "Rate-distortion theory for Gaussian multiterminal source coding systems with several side informations at the decoder," *Submitted to IEEE Trans. Info. Theory*, 2001.

[47] L. H. Ozarow, "On a source coding problem with two channels and three receivers," *Bell Syst. Tech. J.*, vol. 59, pp. 1909–1921, Dec. 1980.

[48] L. H. Ozarow, "The capacity of the white Gaussian multiple access channel with feedback," *IEEE Trans. Inform. Theory*, vol. 30, pp. 623–629, July 1984.

[49] J. Pearl, *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann, 1988.

[50] B. Prabhakar, E. Uysal-Biyikoglu, and A. El Gamal, "Energy-efficient transmission over a wireless link via lazy packet scheduling," in *Proc. IEEE INFOCOMM*, pp. 386–394, Apr. 2001.

[51] S. S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense microsensor network," *IEEE Signal Proc. Mag.*, pp. 51–60, Mar. 2002.

[52] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): Design and construction," in *Proc. Data Compression Conf.*, pp. 158–167, Mar. 1999.

[53] S. S. Pradhan and K. Ramchandran, "Distributed source coding: Symmetric rates and applications to sensor networks," in *Proc. Data Compression Conf.*, pp. 363–372, Mar. 2000.

[54] R. Puri, K. Lee, and K. Ramchandran, "An integrated source transcoding and congestion control paradigm for video streaming in the internet," *IEEE Trans. Multimedia*, vol. 3, pp. 18–32, Mar. 2001.

[55] R. Puri, S. S. Pradhan, and K. Ramchandran, "n-channel multiple descriptions: Theory and construction," *Proc. Data Compression Conf.*, 2002.

[56] K. Ramchandran and M. Vetterli, "Multiresolution joint source-channel coding," in *Wireless Communications: Signal Processing Perspectives* (H. V. Poor and G. W. Wornell, eds.), ch. 7, pp. 282–329, 1998.

[57] B. Rimoldi, "Successive refinement of information: Characterization of the achievable rates," *IEEE Trans. Inform. Theory*, vol. 40, pp. 253–259, Jan. 1994.

[58] A. Sahai, *Anytime Information Theory.* PhD thesis, Mass. Instit. of Tech., 2001.

[59] B. Schein and R. Gallager, "The Gaussian parallel relay channel," in *Proc. Int. Symp. Inform. Theory*, (Sorrento, Italy), p. 22, June 2000.

[60] B. E. Schein, *Distributed Coordination in Network Information Theory.* PhD thesis, Mass. Instit. of Tech., 2001.

[61] H. M. H. Shalaby and A. Papamarcou, "Multiterminal detection with zero-rate data compression," *IEEE Trans. Inform. Theory*, vol. 38, pp. 254–267, Mar. 1992.

[62] S. Shamai, S. Verdu, and R. Zamir, "Systematic lossy source/channel coding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 564–579, Mar. 1998.

[63] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 623–656, 1948.

[64] C. E. Shannon, "Channels with side information at the transmitter," *IBM Journal of Research and Development*, vol. 2, pp. 289–293, Oct. 1958.

[65] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE National Conv. Rec., part 4*, pp. 142–163, 1959.

[66] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.

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

[68] S. Tatikonda, *Control Under Communication Constraints.* PhD thesis, Mass. Instit. of Tech., 2001.

[69] R. R. Tenney and N. S. Sandell, "Detection with distributed sensors," *IEEE Trans. Aerospace Elec. Syst.*, vol. 17, pp. 501–510, July 1981.

[70] S. Tilak, N. B. Abu-Ghazaleh, and W. Heinzelman, "A taxonomy of wireless micro-sensor network models," *To appear: ACM Mobile Comp. and Comm. Review (MC2R).*

[71] D. Tse, R. Gallager, and J. Tsitsiklis, "Optimal buffer control for variable-rate lossy compression," in *Proc. 31st Allerton Conf. on Communication, Control and Computing*, Sept. 1993.

[72] D. N. C. Tse, *Variable-rate Lossy Compression and its Effects on Communication Networks.* PhD thesis, Mass. Instit. of Tech., 1994.

[73] J. N. Tsitsiklis, "Decentralized detection," in *Advances in Statistical Signal Processing, 2*, 1993.

[74] S. Tung, *Multiterminal Source Coding.* PhD thesis, Cornell University, 1978.

[75] H. Viswanathan and T. Berger, "The quadratic Gaussian CEO problem," *IEEE Trans. Inform. Theory*, vol. 43, pp. 1549–1559, Sept. 1997.

[76] R. Viswanathan and P. K. Varshney, "Distributed detection with multiple sensors: Part I – Fundamentals," *Proc. IEEE*, vol. 85, pp. 54–63, Jan. 1997.

[77] M. J. Wainwright, T. Jaakkola, and A. S. Willsky, "Tree-based reparameterization for approximate inference on loopy graphs," in *Advances in Neural Information Processing Systems 14*, 2001.

[78] J. Wolf, A. Wyner, and J. Ziv, "Source coding for multiple descriptions," *Bell Syst. Tech. J.*, vol. 59, pp. 1417–1426, Oct. 1980.

[79] J. K. Wolf and J. Ziv, "Transmission of noisy information to a noisy receiver with minimum distortion," *IEEE Trans. Inform. Theory*, vol. 16, pp. 406–411, July 1970.

[80] A. D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inform. Theory*, vol. 20, pp. 2–10, Jan. 1974.

[81] A. D. Wyner, "The common information of two dependent random variables," *IEEE Trans. Inform. Theory*, vol. 21, pp. 163–179, Mar. 1975.

[82] A. D. Wyner, "On source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 21, pp. 294–300, May 1975.

[83] A. D. Wyner, "The rate-distortion function for source coding with side information at the decoder–II: General sources," *Information and Control*, vol. 38, pp. 60–80, 1978.

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

[85] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Understanding belief propogation and its generalizations," tech. rep., Mitsubishi Electric Research Laboratories, No. TR-2001-22, 2001.

[86] R. Zamir, "The rate loss in the Wyner-Ziv problem," *IEEE Trans. Inform. Theory*, vol. 42, pp. 2073–2084, Nov. 1996.

[87] R. Zamir and T. Berger, "Multiterminal source coding with high resolution," *IEEE Trans. Inform. Theory*, vol. 45, pp. 106–117, Jan. 1999.

[88] R. Zamir and S. Shamai, "Nested linear/lattice codes for Wyner-Ziv encoding," in *1998 Information Theory Workshop, Kilarney, Ireland*, pp. 92–93, June 1998.

[89] R. Zamir, S. Shamai, and U. Erez, "Nested codes: an algebraic binning scheme for noisy multiterminal networks," *IEEE Trans. Inform. Theory*, vol. 42, June 2002.

[90] Z. Zhang and T. Berger, "Estimation via compressed information," *IEEE Trans. Inform. Theory*, vol. 34, p. 198, 1998.