2015-11-17

Quantization and Transmission in Wireless Multi-hop Networks

Behzad Mohammadi Dogahe
University of Miami, behzadmd@gmail.com

UNIVERSITY OF MIAMI

QUANTIZATION AND TRANSMISSION IN WIRELESS MULTI-HOP NETWORKS

By

Behzad Mohammadi Dogahe

A DISSERTATION

Submitted to the Faculty
of the University of Miami
in partial fulfillment of the requirements for
the degree of Doctor of Philosophy

Coral Gables, Florida

December 2015

UNIVERSITY OF MIAMI

A dissertation submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

QUANTIZATION AND TRANSMISSION IN WIRELESS MULTI-HOP NETWORKS

Behzad Mohammadi Dogahe

Approved:

Manohar N. Murthi, Ph.D. Associate Professor of Electrical and Computer Engineering	Xiaodong Cai, Ph.D. Professor of Electrical and Computer Engineering
Kamal Premaratne, Ph.D. Professor of Electrical and Computer Engineering	Shahriar Negahdaripour, Ph.D. Professor of Electrical and Computer Engineering
Dilip Sarkar, Ph.D. Associate Professor of Computer Science	Dean of the Graduate School

MOHAMMADI DOGAHE, BEHZAD (Ph.D., Electrical and Computer Engineering)

Quantization and Transmission in Wireless Multi-hop Networks (December 2015)

Abstract of a dissertation at the University of Miami.

Dissertation supervised by Professor Manohar N. Murthi.
No. of pages in text. (115)

Many applications like streaming audio/video, gaming, distributed or remote computing transmit data over communication networks. The quality-of-service (QoS) of these applications depends on parameters like rate, reliability, delay, power level, etc. Furthermore, these applications need to share network resources and co-exist with other applications which run on the same network.

Since these communication networks are digital communication systems, the transmitter captures the real time data, quantizes and transmits to the receiver. At the receiver not only a reconstruction of the original signal is performed but also there may be a need to classify the signal into several classes. Therefore, reconstruction fidelity and classification are also QoS requirements that many applications may demand.

We are interested in the problem of transmitting data in a layered multi-hop wireless network. The QoS in our system is a function of rates and end-to-end delay from transmitters to the receivers. We are considering wireless multi-hop networks, therefore the capacity of the communication links is a function of power of transmitters and the interference from other links transmitters. Furthermore, we are not only interested in reconstructing the received quantized data but also we would like to perform signal classification on the signals based only on the characteristics of the

received quantized data. We break down this problem into two separate problems to be solved at different layers of communication network as follows

Problem 1 which solves a network optimization problem where QoS depends on the rate, end-to-end delay, and power of transmitters. The solution to this problem essentially entails the rate control (congestion control) in the transport later, and the power control in the physical layer to achieve bounded average queuing delay for the end-to-end transmission of the data.

Problem 2 which focuses on designing a quantizer that guarantees reproduction fidelity of the signals and good classification results based on the information preserved in the reconstructed signal.

Linkage and Inter-connection between Problem 1 and Problem 2 Problem 1 determines the data rate of the sessions. The data rate can be translated into bits/vector rate which is fed into the quantizer that is designed by solving Problem 2. Therefore a communication system is devised that takes advantage of the solutions of Problem 1 and 2 in order to achieve the bigger objective of this dissertation.

Let us now consider the solution to Problem 1. Allocating limited resources such as bandwidth and power in a multi-hop wireless network can be formulated as a Network Utility Maximization (NUM) problem. Researchers have been using NUM in order to develop new network resource allocation algorithms by augmenting the earlier NUM problems. In the NUM framework, sources in the network measure their performance by a utility function. We augment the basic NUM problem with a constraint based on the average queueing delay requirements of the sources. Furthermore, the capacity of

the links in the case of wireless networks depend on the power of transmitters. This augmented NUM formulation turns out to be a non-convex problem. We convert this non-convex problem with high-SIR assumption and a change of variable to a convex problem. We furthermore propose methods to solve this problem in an iterative distributed manner. Simulation results demonstrate the efficacy of the distributed algorithm developed.

To solve Problem 2, we adopt high-rate analysis to design a quantizer that is optimized in the task of reconstruction fidelity as well as classification at the decoder. We deploy symmetric Kullback-Leibler (KL) divergence measure between the conditional probabilities of class given the signal before and after quantization as our distortion measure. We derive the optimum point density of the quantizer for minimizing the symmetric KL divergence. We study tradeoff between classification accuracy and reproduction fidelity. We also derive the effects of a mismatched distortion measure and show that for reduced complexity the original distortion measure can be replaced by a weighted mean square error (WMSE) distortion measure. We examine the performance of these methods on synthetically generated data as well as real data set and observe that our methods are superior in the task of classification of signals at the decoder. The tradeoff is the lower performance in distortion.

The linkage and interrelationship of Problem 1 and 2 is shown at the end. Problem 1 determines the data rate of the sessions. The data rate can be translated into bits/vector rate which is fed into the quantizer that is designed by solving Problem 2. Therefore a communication system is devised that takes advantage of the solutions of Problem 1 and 2 in order to achieve the bigger objective of this dissertation.

to my father

Acknowledgements

I wish to thank everyone who helped me to make this doctoral dissertation what it is. In particular I would like to thank my advisor, Dr. Manohar N. Murthi, who has advised me all through this long process and has provided invaluable guidance, detailed comments and constructive feedback on the results and various drafts that were prepared for conference or journals. I also thank other members of my dissertation committee, Dr. Kamal Premaratne, Dr. Xiaodong Cai, Dr. Shahriar Negahdaripour, and Dr. Dilip Sarkar most of whom I took many classes with, which helped me add to my knowledge and apply to my research.

I would also like to thank other faculty members of the Electrical and Computer Engineering Department, in particular, Dr. James W. Modestino, Dr. Reuven Lask, Dr. Stephen Murrel, Dr. Michael Scordilis, Dr. Saman Zonous, Dr. Mei-Ling Shyu, and Dr. Miroslav Kubat who helped me in one way or another, academically or career-wise. Also Rosamund Coutts, Michelle Perez, Clarisa Alvarez, Kendra Parks, Ivelisse Reyes, Sharon Manjarres, and Lianne Dookie helped me a lot in the administrative tasks of the Electrical and Computer Engineering Department.

I appreciate the discussions I had with Dr. Chris Cosner at the Math Department which helped me outline some of the mathematical proofs in this dissertation. The financial assistance from the grants that this research was supported with is very appreciated.

I would also like to thank Dr. Shihab Asfour for his tremendous support. Also I would like to thank Dr. Asfour's team, members of the Industrial Assessment Center,

in particular Joel Zahlan, Jason Lee Grant, and Mesut Avci with whom I collaborated for parts of my funding.

I thank the following friends whose concerns and thoughts contributed to my experience at the University of Miami: Mohammd H. Mahoor, Mingyu Chen, Zhouyi Xu, Maththondage Chamara Ranasingha, Subasingha Shaminda Subasingha, Steven Cadavid, Murat Aykin, Zifang Huang, Carlos Nocito, Saman Sargolzaei, Guomin Jiang, Hui Lu, Kai Shen, Hossein Shokri, Saeed Khorsandi, Ahmad Seyfi, Mohammad Haghighat, Ali Habashi, Varsha Sainani, and Young-Hun Paik.

Behzad Mohammadi Dogahe

University of Miami

December 2015

Table of Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Many applications like streaming audio/video, gaming, distributed or remote computing transmit data over a telecommunication system. These telecommunication systems are communication networks which can consist of IP based networks, wireless multi-hop networks or other types of networks. See Figure 1.1 which shows the abstraction layers of a telecommunication network.

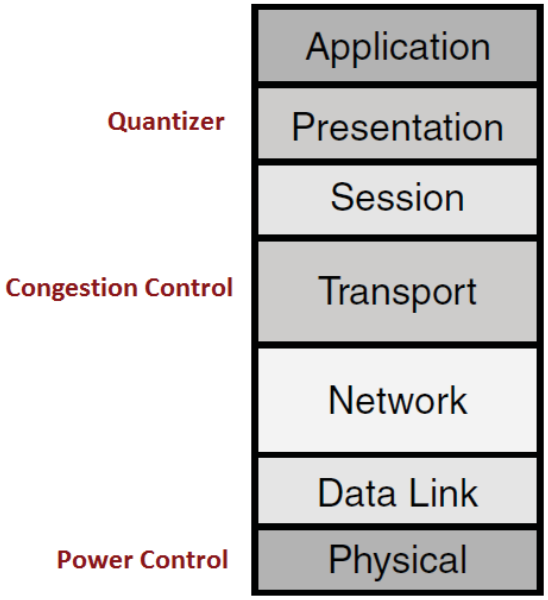


Figure 1.1: Network Layers and the Optimization Algorithms.

An application running on such a network achieves different quality-of-service (QoS) depending on many different parameters. Data in applications are of different forms such as speech, audio, video, or other sensor data. The data is captured, processed, quantized, and transmitted to a receiver. These applications will receive different QoS from the network depending on the rate, reliability, delay, jitter, power level, etc. Each layer of a communication network controls a subset of these variables and also observes a subset of variables and constants from other layers. For some applications, e.g., email or file transfer, the delay between transmitter and receiver is not such a crucial matter as much as it is for an audio/video streaming or gaming applications. For example in a Voice over IP application, voice packets that reach the destination with an unreasonably long delay cannot be of any use. Furthermore, these applications need to co-exist with other applications which run on the same communication network. Therefore they are sharing the network resources. Sharing the network resources brings the necessity to have congestion control algorithms in place to avoid the communication links to become congested, see [KMT98]. In a wireless multi-hop network, the capacity of the links is a function of transmitter powers as well as interference from other transmitters. Consequently, the congestion control in the transport layer and power control in the physical layer need to be jointly designed and distributively implemented, see [Chi05].

Furthermore, since these communication networks are digital communication systems, the transmitter captures the real time data, processes them and generates vector or other multi-dimensional data that need to be quantized and transmitted to the receiver. Thus, the quantization and transmission of vector data is required in many applications. After transmission of the quantized data, a replica of the original

signal is retrieved from the quantized data. Quantizers make an effort to guarantee that the retrieved signal resembles the original signal in one sense or another. For example in sound applications, as long as the de-quantized sound signal sounds very much like the signal before quantization to the human ear, it is considered that the signal recovery was successful. Similarly, when quantizing images, the de-quantized replica of the image should look good to the human eye, or have some low distortion.

At the receiver side of a communication system, not only a reconstruction of the original signal is performed and used but also there may be a need to classify the signal into several classes. For example in a speaker or speech recognition system, an algorithm is in charge of classifying the received signals into say male or female voices, or identifying the person on the other side of the line or recognizing parts of speech automatically, like in many speech automated systems. Therefore, classification is another QoS that many applications may require.

1.1 Problem Statement

We are interested in the problem of transmitting data in a layered multi-hop wireless network. The QoS in our system is not only a function of rates but also our sources may require low end-to-end delay in their transmissions to the receivers. Since we are considering a wireless multi-hop network, the capacity of the communication links is a function of power of transmitters and the interference from other links' transmitters. Furthermore, we are not only interested in reconstructing the received quantized data but also we would like to perform signal classification on the signals based only on the characteristics of the received quantized data.

Figure 1.2 shows the relationship between the parameters involved in this problem, both QoS parameters and the network resources. It also shows the interrelationship between the parameters.

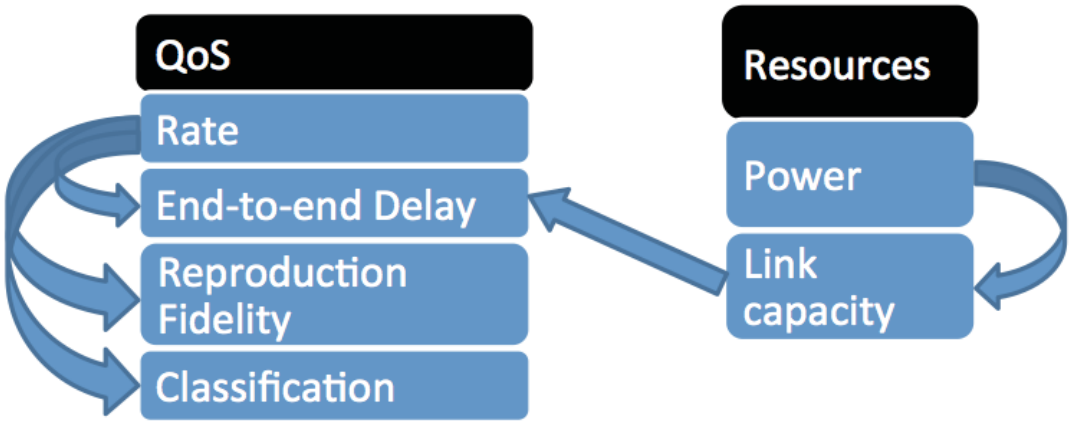


Figure 1.2: QoS parameters, network resources and their relationships.

We break down this problem into two separate problems to be solved at different layers of communication network as follows

Problem 1 which solves a network optimization problem where QoS depends on the rate of transmission, the end-to-end delay, and the power of transmitters which also determines battery life of the devices. The solution to this problem essentially entails the rate control (congestion control) in the transport later, and the power control in the physical layer to achieve bounded average queuing delay for the end-to-end transmission of the data.

Problem 2 which focuses on designing a quantizer in the application/presentation layer that guarantees not only reproduction fidelity of the signals but also good classi-

fication results based on the information preserved in the reconstructed signal. Figure 1.3 shows a diagram of the problem and how it is broken down into sperate problems to be solved at different layers and eventually combined to achieve the main objective of this dissertation.

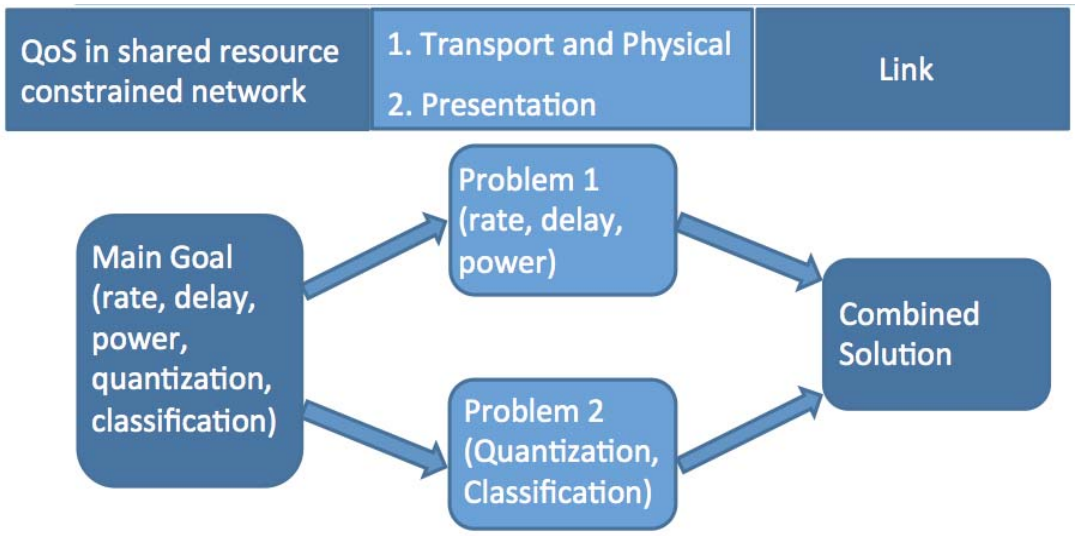


Figure 1.3: Breakdown of the problem.

Linkage and Interrelationship between Problem 1 and Problem 2 Problem 1 determines the data rate of the sessions. The data rate can be translated into bits/vector rate which is fed into the quantizer that is designed by solving Problem 2. Therefore a communication system is devised that takes advantage of the solutions of Problem 1 and 2 in order to achieve the bigger objective of this dissertation. Figure 1.4 shows a diagram explaining how we combine the results of the two problems to achieve our goal.

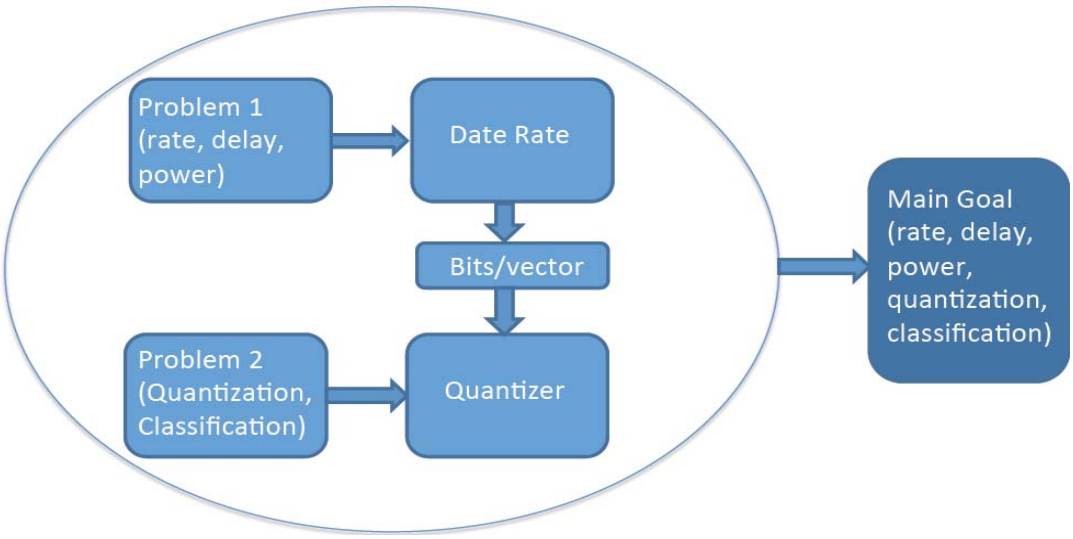


Figure 1.4: Breakdown of the problem.

In the following we present some background to the reader and present some references to related work.

1.2 Background and Related Work

A layered architecture in communication networks adopts a modularized and often distributed approach to network design problems. Each module (layer) controls a subset of variables and also observes a subset of variables and constants from other layers. Each layer in the protocol stack hides the complexity of the lower layer and provides a service to the higher layer. A layered architecture enables a scalable, evolvable, and implementable with lower complexity while introducing limitations in fairness and efficiency. See Figure 1.1.

How does one allocate resources at the various layers of a network shared by multiple users with differing QoS requirements? One can pursue a cross-layer de-

sign approach to the problem of network resource allocation at the expense of higher complexity in communication and computation. Even after crossing the layers, it is desirable to obtain architectural modularity for practical implementation. In the following we will show that we can take advantage of the Network Utility Maximization (NUM) framework to decompose a complex network problem into sub-problems. These sub-problems can be solved at different layers, yet communicate with each other through either implicit or explicit message passing. The implicit messages are events or quantities which are observed or measured through normal behavior of the rate control algorithms, for example they could be the receipt of the acknowledgment packets or the end-to-end delay measured from the time a packet is sent until its acknowledgment packet is received. The explicit messages quantify information that is needed for a specific decomposition.

The problem of allocating limited resources in a multi-hop wireless network can be formulated as a NUM problem. NUM is a mathematical optimization framework that was used to model the working Transmission Control Protocol (TCP) congestion control algorithms [KMT98, LL99, LPW01]. Because of the flexibility of this framework, NUM has been extended from an analytic tool of reverse-engineering to a mathematical theory of a layered network architecture. Researchers have been using it in order to develop new network resource allocation and internet congestion control algorithms by modifying or augmenting the earlier NUM problems. In the NUM framework, sources in the network measure their performance by a utility function. This function is a concave function of usually transmission rate and possibly other parameters. For example the utility function for TCP Vegas is $U_s(x_s) = \log(x_s)$, where x_s is the data rate of source s. The objective of each source is to maximize

its utility and the objective of the network as a whole is to maximize the aggregate utility of all the sources. Clearly, the transmission rate of each link cannot exceed the link capacity. Therefore, NUM is a constrained optimization problem.

Extensions to the NUM problem There are numerous extensions of the basic NUM problem. To mention a few, in [Chi05], joint congestion control and power control is examined; in [LCC06a], an optimization on rate and reliability is considered; in [LCC06b], a joint end-to-end congestion control and per-link medium access control in ad-hoc networks is studied; in [SLGY07], the problem of allocating rate to sessions with bandwidth and delay requirements is addressed; in [LMS04], a joint time-slot and power allocation method for wireless cellular systems is presented; and in [MB02], resource allocation and pricing for the downlink of a wireless network is considered.

Delay in the NUM framework While these papers have made many significant contributions, to the best of our knowledge, the problem of incorporating the delay requirement of the sources into the resource allocation problems in wireless networks with interference-limited link capacities has not been considered in the NUM framework. In terms of handling delay in wireless networks, in [RLM02], a dynamic programming optimization method is introduced to obtain the optimal bit-rate/delay control policy in the downlink for packet transmission in wireless networks with fading channels. In [RLM02], a fixed symbol rate is assumed and different bit rates are achieved by choosing the transmitted symbols from the appropriate signal constellation (adaptive modulation); and in [JCOB02], several extensions of the NUM problem including queuing delay are outlined. In [LCCD09], the authors incorporate the delay

in addition to rate and reliability in the NUM problem. However, they assume fixed capacity links that consist of sub-links with different rate-reliability characteristics.

Through generalized convexity arguments, the study [PDE08] addressed the allocation of power and bandwidth for the general SIR case for the basic NUM problem (2.1). However in this dissertation the constraints in the optimization problem that result from the delay requirements are much more complex than the constraints in [PDE08]. Therefore the series of transformations that the authors proposed in [PDE08] to convert the non-convex problem into a convex problem do not appear to lead to a convex problem in our case. While we focus on the high SIR case ([Chi05], [CB04], [JXB03], [MCLG06]) in this dissertation, the extension to the general SIR case is a matter of further study.

There have been many extensions to the problem of incorporating end-to-end delay requirements in the NUM framework. In [LPCC11], authors incorporated delay to the objective function as a penalty term and studied the stability conditions for delay-independent and delay-dependent cases. By introducing Virtual Link Capacity Margin, which reflects the gap between feasible link capacity and maximum allowable rate over link, the authors in [QBX14] characterized delay as constraint and proposed a joint rate allocation and scheduling scheme in multi-hop wireless networks.

The aforementioned literature covers some of the studies we found related to the first problem we are trying to solve, **Problem 1**. How about the second problem, **Problem 2?**

Vector quantization Vector quantization (VQ) is an extension of scalar quantization for quantizing vectors. Vectors can be a representation of a speech waveform

or an image or they could be parameters that represent a spectral envelope of speech signals, for instance. A vector quantizer consists of a set of reproduction vectors which are also called codepoints, c_i, $i = 1, ...,M$. In the vector quantization process, the reproduction vector c_{i^*} that minimizes the distortion between the source vector s_k and the reproduction vector c_i is selected and its index i^* is transmitted to the decoder, see Figure 1.5. The number of reproduction vectors in the quantizer is dependent on the rate (number of bits) available for the transmission of the quantization index. For example a quantizer with N reproduction vectors needs $log_2 N$ bits to transmit the quantization index.

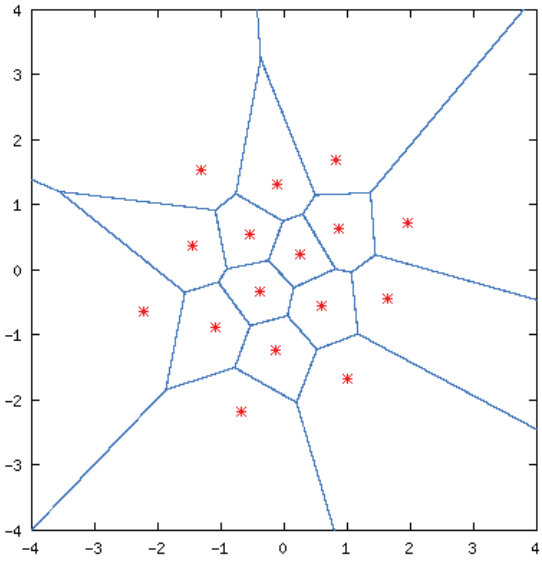


Figure 1.5: An example of 2D VQ - Every pair of numbers falling in each partition are approximated by the red star associated with that partition.

High-Rate Theory In the development of vector quantizers [Gra89] we can use high-rate theory, an asymptotic approximation of a quantizer when the bit rate is

high. A very fundamental assumption of high-rate theory is that for each region where the same reproduction vector is assigned as the quantization vector, the probability density function of the source is uniform. Based on the high-rate theory, we can derive the optimal point density function that minimizes our choice of distortion measure. The point density function determines the number of reproduction vectors in a unit volume of the source data space. The point density function is a function of the probability density function of the source data. Therefore, the knowledge of the probability density function (PDF) of the source is required or one can model the PDF of a signal vector by a Gaussian Mixture Model (GMM).

Quantization of signals has been extensively studied [GN98, GG92]. Several studies have been done in combining detection and quantization with different distortion measures [Kas77, SS77, GH03]. In [VB10], the authors study quantization for detection in a sensor network scenario. In [VV08], the quantization of prior probabilities for hypothesis testing is presented. These studies have made significant contributions to the design of quantizers for the purpose of detection. However, most of them are applicable to binary classification, i.e., binary detection or hypothesis testing, and either are not expandable to multiple class problems or the extension is not straightforward, to the best of our knowledge. There has been some research on combining quantization and classification. In [PPG$^+$96], the authors extend the distortion measure for the Lloyd algorithm by a Bayes risk term which is defined by a weighted cost of classification error. Moreover, in [LJ07], the authors proposed a method for joint quantization and classification in distributed quantizers in which the quantizers sequentially process an input and communicate with each other. Despite the aforementioned contributions, the theory of high-rate quantization for the purpose

of classification still remains somewhat underdeveloped compared to quantization for detection, or quantization theory in general.

1.3 Contributions of Our Work

For **Problem 1** we will examine an augmented NUM formulation in which rate and power control in a wireless network are balanced to achieve bounded average queueing delays for sources. With the additional delay constraints, the augmented NUM problem is non-convex. Therefore, we will present a change of variable to transform the problem to a convex problem and we will develop a solution which results in a distributed rate and power control algorithm tailored to achieving bounded average queueing delays. The solution entails power control in the physical layer and congestion control in the transport layer. See Figure 1.1.

For **Problem 2**, let us remind the reader that the performance of a quantization system is gauged by a measure called the distortion measure which is application specific. Minimizing the distortion measure in quantization guarantees a level of similarity between original and recovered signal in one sense or another. Mean Square Error (MSE) is the distortion measure of choice for many quantization applications. We choose a novel distortion measure defined as the symmetric Kullback-Leibler (KL) divergence measure between the conditional probabilities of class given the signal before and after quantization. Our approach in using the KL divergence measure is different than some other related research, notably [GH03, LJ07]. To be specific, others have tried to minimize the loss in the discrimination between the two classes due to quantization. On the other hand, we try to minimize the aggregate discrimination of the conditional probabilities of classes given the signal before and after quantiza-

tion. Intuitively, it means that we will make sure that the conditional probability of the classes given the data undergoes the least amount of change due to quantization. We will present a high-rate analysis of the quantizer and derive the optimum point density of the quantizer for minimizing the symmetric KL divergence. Furthermore, we will study the tradeoff between classification accuracy and reproduction fidelity and present a point density function for the case of tradeoff. We will also derive the effects of a mismatched distortion measure and show that for reduced complexity the original distortion measure can be replaced by a weighted mean square error (WMSE) distortion measure. We will examine the performance of these methods on synthetically generated data and a real data set and observe that these methods are superior in the task of classification of signals at the decoder with lower reproduction fidelity as a tradeoff.

Furthermore, we will demonstrate that solutions to these two important problems can be combined to devise a communication system that not only optimizes the rate of transmission and power of transmitters to achieve bounded average delay but also deploys a quantizer that is optimized in the task of classification at the decoder as well as reconstruction of the signal.

1.4 Dissertation Outline

The rest of this dissertation is organized into three main chapters as follows:

Chapter 2 attacks the first objective of this dissertation which is incorporating the average delay requirements of the sources into the network resource allocation problem. We construct a NUM problem in a multi-hop wireless network with inter-

ference dependent link capacities. The average delay requirements is introduced as a constraint in the NUM problem. This problem turns to be a non-convex problem. Through a series of transformations we cast the problem into a convex problem and solve it in a distributed manner. We present simulation results on different network topologies and various scenarios of network demands.

Chapter 3 studies the second objective of this dissertation, the problem of quantizing signals for the purpose of achieving good classification at the decoder. We propose a new distortion measure that is optimized for the task of classification at the decoder and design a quantizer based on the high-rate theory. We present results of classifications on both synthetic and real data sets.

Chapter 4 presents simulations where a network such as the one presented in Chapter 2 is in charge of transmitting signals from different classes as is presented in Chapter 3. This chapter demonstrates how the algorithms developed in the previous two chapters can work in tandem to not only satisfy rate, power, and delay requirements of the network as well as better performance of classification at the decoder.

Chapter 5 summarizes the dissertation and outlines some future research directions.

CHAPTER 2

A Distributed Congestion and Power Control Algorithm to Achieve Bounded Average Queuing Delay in Wireless Networks

Allocating limited resources such as bandwidth and power in a multi-hop wireless network can be formulated as a Network Utility Maximization (NUM) problem. In this approach, both transmitting source nodes and relaying link nodes exchange information allowing for the NUM problem to be solved in an iterative distributed manner. Some previous NUM formulations of wireless network problems have considered the parameters of data rate, reliability, and transmitter powers either in the source utility function which measures an application's performance or as constraints. However, delay is also an important factor in the performance of many applications. In this dissertation, we consider an additional constraint based on the average queueing delay requirements of the sources. In particular, we examine an augmented NUM formulation in which rate and power control in a wireless network are balanced to achieve bounded average queueing delays for sources. With the additional delay constraints, the augmented NUM problem is non-convex. Therefore, we present a change of variable to transform the problem to a convex problem and we develop a solution

which results in a distributed rate and power control algorithm tailored to achieving bounded average queueing delays. Simulation results demonstrate the efficacy of the distributed algorithm.

2.1 Overview

Multihop wireless networks are created by a group of nodes that communicate via wireless links possessing interference-limited capacities. Without a fixed infrastructure or centralized administrator, these networks are decentralized and ad-hoc in manner. Each node can act as either a source or a router to forward traffic to the destination. Mobile ad-hoc networks and sensor networks are examples of multihop wireless networks.

Applications like voice, streaming multimedia, gaming, distributed and remote computing, might expect different levels of data rate, reliability, and end-to-end delay from the wireless communication system. Data that is transferred without meeting the requirement of the application, as far as the aforementioned attributes are concerned, might be of little or no use to the application. Therefore, there arises a need for algorithms that accommodate different trade-offs between requirements and the costs to satisfy them. Moreover, wireless resources such as power and bandwidth are far from sufficient unless wise resource utilization techniques are deployed in order to achieve better efficiency. The framework of Network Utility Maximization (NUM) [KMT98] has been used in many network resource allocation and internet congestion control protocols [LCC06a, Chi05, Low03].

In a network of interconnected nodes, associated with each data session, there is a session data rate x_s (bits/sec) and a utility $U_s(x_s)$ for session s. The utility $U_s(x_s)$

is an increasing, strictly concave, and continuously differentiable function of x_s, and is a quality measure as well. In other words, the higher the data rate, the better the quality measure of session s. The total utility of the network is assumed to be the sum of the individual utilities of sessions. The NUM problem, in its most basic formulation, can be stated as a maximization of the aggregate utility of the sources in the network subject to rate constraints on each link:

$$\max_{x_s, \forall s} \sum_s U_s(x_s),$$

$$\text{subject to: } \sum_{s:\ \ell \in L(s)} x_s \leq c_\ell,\ \forall \ell;\ x_s \geq 0,\ \forall s, \tag{2.1}$$

where c_ℓ represents the capacity of link ℓ and $L(s)$ represents the set of links that session s uses in order to transmit the data from its source to its corresponding destination. Hence the routing information is in $L(s)$. In this dissertation we assume that routing is determined by a lower layer mechanism and that the routing changes on a time scale that is much slower than the rate of convergence of the algorithms developed in this dissertation [Chi05]. Therefore, the objective of the NUM formulation is to maximize the aggregate utility of the sessions while satisfying the constraints, viz., session rates are positive and the aggregate data rate on each link does not exceed the link capacity. A standard solution to this problem is based on Lagrangian decomposition (e.g., [LL99]). Among recent papers are numerous extensions of this basic NUM problem. To mention a few, in [Chi05], joint congestion control and power control is examined; in [LCC06a], an optimization on rate and reliability is considered; in [LCC06b], a joint end-to-end congestion control and per-link medium access control in ad-hoc networks is studied; in [SLGY07], the problem of allocating rate to sessions with bandwidth and delay requirements is addressed; in [LMS04], a joint time-slot

and power allocation method for wireless cellular systems is presented; and in [MB02], resource allocation and pricing for the downlink of a wireless network is considered.

While these papers have made many significant contributions, to the best of our knowledge, the problem of incorporating the delay requirement of the sources into the resource allocation problems in wireless networks with

interference-limited link capacities has not been considered in the NUM framework. In terms of handling delay in wireless networks, in [RLM02], a dynamic programming optimization method is introduced to obtain the optimal bit-rate/delay control policy in the downlink for packet transmission in wireless networks with fading channels. In [RLM02], a fixed symbol rate is assumed and different bit rates are achieved by choosing the transmitted symbols from the appropriate signal constellation (adaptive modulation); and in [JCOB02], several extensions of the NUM problem including queuing delay are outlined. In [LCCD09], the authors incorporate the delay in addition to rate and reliability in the NUM problem. However, they assume fixed capacity links that consist of sub-links with different rate-reliability characteristics.

Propagation delay, transmission delay, processing delay and queuing delay are the main components of the total delay in a network. For an application producing bursty data in which the short-term data rate may exceed the capacity of the wireless links, queuing delay can be quite significant [JCOB02]. The lack of available capacity demands buffering, and therefore queuing delay can be the dominant component of total delay. Therefore in this dissertation we only consider the average queuing delay.

In this dissertation, we build on the work in [Chi05] and [SLGY07] to consider balancing powers of the transmitters and rates of the sessions in a multihop wireless network in order to achieve bounded average queuing delays for the sessions with

delay sensitive requirements within the NUM framework. The delay constraints and interference-dependent link capacities cause the problem to become a complex optimization problem with non-convex constraints. With a high SIR assumption we are able to transform the non-convex problem to a convex problem ([BV04, BB99]) by a change of variable. Furthermore, we develop the algorithm in order to solve the problem in a distributed manner. In the solution algorithm, *sessions* adjust their rates to maximize their utility function but take into account the cost they have to incur for using the bandwidth. The *network* takes the responsibility of breaking up the delay requirement of the sessions into delay requirements along each link. The link *transmitters* adjust their power levels to accommodate sufficient data rate along the links.

Through generalized convexity arguments, a recent paper [PDE08] addressed the allocation of power and bandwidth for the general SIR case for the basic NUM problem (2.1). However in this dissertation the constraints in the optimization problem that result from the delay requirements are much more complex than the constraints in [PDE08]. Therefore the series of transformations that the authors proposed in [PDE08] to convert the non-convex problem into a convex problem do not appear to lead to a convex problem in our case. There is further explanation of this issue at the end of Section 2. While we focus on the high SIR case ([Chi05], [CB04], [JXB03], [MCLG06]) in this dissertation, the extension to the general SIR case is a matter of further study.

Our preliminary results on this part of this dissertation appeared in [DFMP07], and [DMFP10]. The rest of the chapter is organized as follows. In Section 2.2, the system model and optimization problem is presented in which the basic NUM problem

is expanded to accommodate the power and delay requirements. In Section 2.3, the algorithm to solve the expanded NUM problem is derived along with its convexity and feasibility discussions. Performance of the new algorithm along with comparisons with previous works are provided in Section 2.4.

2.2 System Model and Optimization Problem

Consider a wireless network with N nodes which are connected to one another via L links. In wireless networks, the link capacities are dependent on the signal to interference ratios of the corresponding links. The signal-to-interference ratio (SIR) on each link depends on the power of the transmitter on that link as well as the consequent interference due to other links' transmitter powers and the noise on the link [Gol05]. Therefore, the constant capacity assumption in wired networks is not valid in the wireless case anymore. Due to interference in wireless networks, capacities of the links are interdependent. The algorithm that we develop in this dissertation detects the bottleneck link implicitly in a distributed and adaptive manner, as in [Chi05], and hence by power re-distribution optimally moves the bottleneck link around in order to achieve the delay constraints. Assuming $\mathbf{P}$ to be the transmit power vector whose ℓ-th element P_ℓ is the power level of the transmitter of link ℓ, the capacity of the wireless link ℓ can be expressed as

$$c_\ell(\mathbf{P}) = B \log(1 + K \, \mathrm{SIR}_\ell(\mathbf{P})), \tag{2.2}$$

where K is a constant that depends on the modulation and the required bit error rate [Gol05]. Let B be the bandwidth of the channel that can be assumed to be one

unit without loss of generality [Gol05]. Then, the SIR for link ℓ may be expressed as

$$\mathrm{SIR}_\ell(\mathbf{P}) = \frac{P_\ell G_{\ell\ell}}{\sum_{k\neq\ell} P_k G_{\ell k} + n_\ell}, \tag{2.3}$$

where $G_{\ell\ell}$ and $G_{\ell k}$ are path gain and path loss, respectively, and n_ℓ is the thermal noise power on the receiver of link ℓ, [Bam98]. Note that c_ℓ can be approximated as $\log(\mathrm{SIR}_\ell)$ assuming that $K\,\mathrm{SIR} \gg 1$ (i.e., high SIR scenarios) and by taking the effect of K into $G_{\ell\ell}$, i.e.,

$$c_\ell(\mathbf{P}) \approx \log(\mathrm{SIR}_\ell(\mathbf{P})). \tag{2.4}$$

This research proposes and analyzes a distributed algorithm for joint optimization of end-to-end congestion control, power control, and average queuing delay requirements of the sessions based on the NUM framework for the high SIR case of (2.4). Consider the requirement of the average queuing delay on a link to be less than a specific amount, i.e., $E(T_\ell) \leq d_\ell$, where d_ℓ is the local average delay allowed on link ℓ and T_ℓ is the queuing delay on link ℓ. Let us assume a queuing delay model based on the assumption of general packet length distribution (with mean $1/\mu$ and variance σ^2 on every link) and each link modeled as an $M/G/1$ queue [SLGY07, LCCD09, BG92, GK77]. Therefore the average queuing delay on each link is equal to (see [SLGY07] or [GK77])

$$E(T_\ell) = \frac{(1-\beta)/\mu}{c_\ell} + \frac{\beta/\mu}{c_\ell - \sum_{s:\ \ell\in L(s)} x_s}, \tag{2.5}$$

where $\beta = (1+\mu^2\sigma^2)/2$. As is also mentioned in [SLGY07], (2.5) is equivalent to the *Pollaczek-Khinchin* (P-K) *formula* [BG92, page 186] for average delay in M/G/1 queues. Therefore, substituting for the expression of $E(T_\ell)$ in the delay requirement

of $E(T_\ell) \leq d_\ell$, it follows that

$$\frac{(1-\beta)/\mu}{c_\ell} + \frac{\beta/\mu}{c_\ell - \sum_{s:\ \ell \in L(s)} x_s} \leq d_\ell. \tag{2.6}$$

Following some simple algebraic manipulations (2.6) modifies to

$$\sum_{s:\ \ell \in L(s)} x_s \leq c_\ell - \frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell}}. \tag{2.7}$$

Note that the aforementioned delay model (2.5) is based on packet-by-packet transmission. On the other hand, the NUM framework formulation is based on a fluid model for transmission. Mixing packet-by-packet transmission models with fluid models is an approximation but very common in the NUM literature [SLGY07, LCCD09]. Now consider the requirement of the aggregate data rate of all sessions going through a link to be positive and less than the capacity of the link, i.e., $0 \leq \sum_{s:\ \ell \in L(s)} x_s \leq c_\ell$. We now observe that this requirement is equivalent to $E(T_\ell) \geq 1/\mu\, c_\ell$. Indeed, note that

$$\frac{1}{\mu\, c_\ell} \leq E(T_\ell) \leq \infty. \tag{2.8}$$

Substituting for the expression for $E(T_\ell)$ from (2.5), we get

$$\frac{1}{\mu\, c_\ell} \leq \frac{(1-\beta)/\mu}{c_\ell} + \frac{\beta/\mu}{c_\ell - \sum_{s:\ \ell \in L(s)} x_s} \leq \infty. \tag{2.9}$$

Next, apply the following algebraic manipulations:

- Move the first term in the middle to the other side of the inequalities and multiply by μ:

$$\frac{1}{c_\ell} - \frac{1-\beta}{c_\ell} \leq \frac{\beta}{c_\ell - \sum_{s:\ \ell \in L(s)} x_s} \leq \infty. \tag{2.10}$$

- Combine the terms on the left hand side and divide by β:

$$\frac{1}{c_\ell} \leq \frac{1}{c_\ell - \sum_{s:\ \ell \in L(s)} x_s} \leq \infty. \tag{2.11}$$

- Invert the terms:

$$0 \leq \sum_{s:\ \ell \in L(s)} x_s \leq c_\ell. \tag{2.12}$$

Therefore considering the power dependent link capacities (using $c_\ell(\mathbf{P})$ instead of c_ℓ), the NUM problem can be expressed as follows:

$$\begin{aligned} &\max_{\mathbf{x},\mathbf{P},\mathbf{d}} \sum_s U_s(x_s) \\ &\text{subject to: } \sum_{s:\ \ell \in L(s)} x_s \leq c_\ell(\mathbf{P}) - \frac{\beta}{\mu\, d_\ell - \dfrac{1-\beta}{c_\ell(\mathbf{P})}},\ \forall \ell; \\ &d_\ell \geq \frac{1}{\mu\, c_\ell(\mathbf{P})},\ \forall \ell; \\ &\sum_{\ell \in L(s)} d_\ell \leq D_s,\ \forall s;\ x_s \geq R_s,\ \forall s. \end{aligned} \tag{2.13}$$

where the objective function is the aggregate sum of the utilities of the sessions. The first constraint is the result of the requirement that $E(T_\ell) \leq d_\ell$ which yields (2.7); the second constraint is the result of $0 \leq \sum_{s:\ \ell \in L(s)} x_s \leq c_\ell$; the third constraint is simply the total average queuing delay requirement for session s, which is the sum of per-link queuing delays (d_ℓ) for the links that session s uses, and D_s is the requirement that session s imposes on its own queuing delay; and the fourth constraint is the minimum rate requirement for each session. Here the optimization variables are $\mathbf{x}$, $\mathbf{P}$, and $\mathbf{d}$ as opposed to just $\mathbf{x}$ in the basic NUM problem. The assumption of positive power and a maximum allowable power for each transmitter is obvious and taken into account in the optimization problem, even if it is not explicitly indicated hereafter as in [Chi05].

As we mentioned in the introduction, in [PDE08] the authors solve the basic NUM problem (2.1) to allocate power and bandwidth for the general SIR case. Specifically in [PDE08], the constraint in the optimization problem is stated as

$$\sum_{s:\ \ell\in L(s)} x_s \leq c_\ell(\mathbf{P}),\ \forall \ell. \tag{2.14}$$

The authors in [PDE08] use several results and theorems (i.e., *Results 2-4, and Theorems 1-2* in [PDE08]), to prove that the constraint (2.14) will be convex using the following transformation

$$\log\left(\sum_{s:\ \ell\in L(s)} \exp(\tilde{x_s})\right) \leq \log\left(c_\ell(\tilde{\mathbf{P}})\right),\ \forall \ell, \tag{2.15}$$

where $\tilde{P} = \log P$, and $\tilde{x_s} = \log x_s$. Following similar steps to those in [PDE08] do not appear to convert the non-convex constraint in (2.13) into a convex constraint for the general SIR case. So in this dissertation we focus on the high SIR case. We prove the convexity of the optimization problem (2.13) for the high SIR case in the next section.

2.3 Distributed Resource Allocation Algorithm

To solve this NUM problem, we use the dual decomposition approach [PC06]. In a convex optimization problem with zero duality gap, solving the dual problem will result in an optimal solution for the primal problem. But in the case of non-convex optimization, the duality gap is non-zero. A non-zero duality gap means that the standard dual-decomposition algorithm may lead to suboptimal solutions. The primal problem (2.13) is a non-convex problem since the first and second constraints are not convex. But if we use the high-SIR approximation of the link capacities, i.e.,

(2.4), then problem (2.13) is a convex optimization problem if the change of variable $\tilde{\mathbf{P}} = \log \mathbf{P}$ is applied. We summarize this in the following observation.

Observation: If the utility functions $U_s(x_s)$ are concave in x_s, and the high SIR approximation of link capacities (i.e., (2.4)) is used, and the change of variable $\tilde{\mathbf{P}} = \log \mathbf{P}$ is performed, then the optimization problem (2.13) is convex in $\mathbf{x}, \tilde{\mathbf{P}}, \mathbf{d}$.

Proof: In order to achieve a unique optimum in the problem of maximizing a concave function with a constraint of the form $f(x) \leq 0$, the constraint function $f(x)$ needs to be convex. With a log transformation $\tilde{\mathbf{P}} = \log \mathbf{P}$, the term $c_\ell(\exp(\tilde{\mathbf{P}}))$ is strictly concave (see [Chi05]). Therefore $\frac{1}{\mu c_\ell(\exp(\tilde{\mathbf{P}}))}$ is strictly convex and consequently the second constraint in (2.13) is a convex constraint. Furthermore, the second constraint in (2.13) and the fact that $\beta \geq 0$ ensure that $\mu d_\ell \geq \frac{1-\beta}{c_\ell(\exp(\tilde{\mathbf{P}}))}$. This inequality and the convexity of $c_l(\exp(\tilde{\mathbf{P}}))$ prove that $\frac{\beta}{\mu d_\ell - \frac{1-\beta}{c_\ell(\exp(\tilde{\mathbf{P}}))}}$ is a convex function in d_ℓ, and $\tilde{\mathbf{P}}$. Therefore the first constraint is convex. The rest of the constraints are linear constraints. Hence, the problem in (2.13) is a convex optimization problem.□

For simplicity we carry out the equations in the $\mathbf{P}$ domain. To proceed, we introduce Lagrange multipliers $\boldsymbol{\lambda} > 0$ to formulate the Lagrange dual function corresponding to the primal problem (2.13) as

$$L_1(\mathbf{x}, \mathbf{P}, \mathbf{d}, \boldsymbol{\lambda}) \;=\; \sum_s U_s(x_s) + \sum_\ell \lambda_\ell \left(c_\ell(\mathbf{P}) - \frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell(\mathbf{P})}} - \sum_{s:\ \ell \in L(s)} x_s \right). \quad (2.16)$$

Note that in this Lagrangian function, we just relaxed the first constraint in

problem (2.13). The Lagrange dual function is

$$Q_1(\boldsymbol{\lambda}) = \max_{\mathbf{x},\mathbf{P},\mathbf{d}} L_1(\mathbf{x},\mathbf{P},\mathbf{d},\boldsymbol{\lambda})$$
$$\text{with: } d_\ell \geq \frac{1}{\mu\, c_\ell(\mathbf{P})},\ \forall \ell;\ \sum_{\ell \in L(s)} d_\ell \leq D_s,\ \forall s;\ x_s \geq R_s,\ \forall s. \tag{2.17}$$

The dual problem can be formulated as

$$\min Q_1(\boldsymbol{\lambda}) \text{ subject to: } \boldsymbol{\lambda} > 0. \tag{2.18}$$

To solve the dual problem, we first try to solve (2.17). Because of the linearity of the differentiation operator, we decompose (2.17) into two separate parallel problems as follows:

- The *source* problem:

$$\max_{\mathbf{x}} \sum_s U_s(x_s) - \sum_\ell \sum_{s:\ \ell \in L(s)} \lambda_\ell x_s \text{ subject to: } x_s \geq R_s,\ \forall s. \tag{2.19}$$

- The *link* problem:

$$\max_{\mathbf{P},\mathbf{d}} \sum_\ell \lambda_\ell \left(c_\ell(\mathbf{P}) - \frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell(\mathbf{P})}} \right)$$
$$\text{subject to: } d_\ell \geq \frac{1}{\mu\, c_\ell(\mathbf{P})},\ \forall \ell;\ \sum_{\ell \in L(s)} d_\ell \leq D_s,\ \forall s. \tag{2.20}$$

Then, the dual problem (2.18) can be solved by using the gradient projection algorithm [BB99] as

$$\lambda_\ell = \left[\lambda_\ell - \kappa_1 \left(c_\ell - \frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell}} - \sum_{s:\ \ell \in L(s)} x_s \right) \right]^+, \tag{2.21}$$

where $[x]^+ = \max\{x, 0\}$, and κ_1 is the step size for the gradient method. Different time-dependent step-sizes, e.g., $\kappa_1 = \kappa_0/t$, or $\kappa_1 = \kappa_0/\sqrt{t}$ can guarantee the convergence of λ_ℓ as the iteration time, t, increases, [Chi05]. We utilize this approach to

step-sizes throughout the paper. The *source* problem in (2.19) can itself be separated to be solved at each source independently by just the feedback information of $\sum_\ell \lambda_\ell$, i.e.,

$$\max_{x_s \geq R_s} U_s(x_s) - x_s \sum_{\ell \in L(s)} \lambda_\ell. \tag{2.22}$$

The *link* problem in (2.20) is the more difficult problem because it is coupled in two variables $\mathbf{P}$, and $\mathbf{d}$. To proceed, we decouple the problem in (2.20) into two separate convex problems. The derived congestion-control algorithm, that is named `Algorithm I` later in this section, will be distributed in the *power allocation* with some message passing, yet centralized in partitioning the end-to-end delay requirements to per-link delays. We furthermore present an approach to de-centralize the solution of allocating per-link delays, leading to `Algorithm II`.

Now let us consider decomposing the link problem in (2.20) into two separate optimization problems. Each optimization corresponds to only one optimization variable, assuming the other one as a constant. The rationale behind this assumption is that the two optimization variables ($\mathbf{P}$, and $\mathbf{d}$) vary in value at different time scales. To be more specific, the rate at which the transmitters' powers ($\mathbf{P}$) and therefore the power-dependent capacities ($c_l(\mathbf{P})$) change is much lower than the rate at which the delay distribution ($\mathbf{d}$) on links changes. The first problem which assumes $\mathbf{P}$ and therefore c_l's are relatively constant is

- the *delay distribution* problem:

$$\min_{\mathbf{d}} \sum_{\ell} \lambda_\ell \left(\frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell}} \right)$$
$$\text{subject to: } d_\ell \geq \frac{1}{\mu\, c_\ell},\ \forall \ell;\ \sum_{\ell \in L(s)} d_\ell \leq D_s,\ \forall s. \tag{2.23}$$

The second problem is

- the *power allocation* problem

$$\max_{\mathbf{P}} \sum_{\ell} \lambda_\ell \left(c_\ell(\mathbf{P}) - \frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell(\mathbf{P})}} \right), \tag{2.24}$$

which is equivalent to

$$\max_{\mathbf{P}} \sum_{\ell} \lambda_\ell c_\ell(\mathbf{P}) \left(\frac{\mu\, d_\ell c_\ell(\mathbf{P}) - 1}{\mu\, d_\ell c_\ell(\mathbf{P}) - 1 + \beta} \right). \tag{2.25}$$

In the following two sub-sections we introduce two solutions to the *power allocation* problem.

2.3.1 Exact Power Allocation Solution

We use the gradient method [BB99], to solve the *power allocation* problem as follows

$$P_\ell(t+1) = P_\ell(t) + \kappa_p\, \nabla_\ell I(\mathbf{P}), \tag{2.26}$$

where

$$I(\mathbf{P}) = \sum_{\ell} \lambda_\ell c_\ell(\mathbf{P}) \left(\frac{\mu\, d_\ell c_\ell(\mathbf{P}) - 1}{\mu\, d_\ell c_\ell(\mathbf{P}) - 1 + \beta} \right) \tag{2.27}$$

$\nabla_\ell(.)$ is the derivative with respect to P_ℓ, and κ_p is the step size for the gradient method. Following [Chi05], the derivative with respect to $\tilde{P}_\ell$ is simply a scaled version of the derivative with respect to P_ℓ with the scaling factor of P_ℓ. Hence the power

update in either $\mathbf{P}$ or $\tilde{\mathbf{P}}$ domain will work. To derive the expression for $\nabla_\ell I(\mathbf{P})$, we substitute for the relationship between SIR_ℓ and $\mathbf{P}$ from (2.3). We also use the approximation $c_\ell(\mathbf{P}) \approx \log(\mathrm{SIR}_\ell(\mathbf{P}))$, therefore

$$\nabla_\ell I(\mathbf{P}) = \frac{\lambda_\ell}{P_\ell}\left(\frac{\mu d_\ell c_\ell(\mathbf{P})-1}{\mu d_\ell c_\ell(\mathbf{P})-1+\beta}+\frac{\beta\mu d_\ell c_\ell(\mathbf{P})}{\Big(\mu d_\ell c_\ell(\mathbf{P})-1+\beta\Big)^2}\right)- \sum_{j\neq\ell}\frac{\lambda_j G_{j\ell}}{\sum_{k\neq j} G_{jk}P_k+n_j}\left(\frac{\mu d_j c_j(\mathbf{P})-1}{\mu d_j c_j(\mathbf{P})-1+\beta}+\frac{\beta\mu d_j c_j(\mathbf{P})}{\Big(\mu d_j c_j(\mathbf{P})-1+\beta\Big)^2}\right). \quad (2.28)$$

Note that calculating $\nabla_\ell I(\mathbf{P})$ is computationally extensive. Therefore in the following we develop an approximate solution. Using the approximate solution entails less extensive computations and it uses less amount of message passing. Comparing the simulation results from the developed exact solution [(2.26), and (2.27)] with the results of the approximate solution shows that the effect of this approximation is negligible. Therefore in all practical applications the approximate solution is recommended.

2.3.2 Approximate Power Allocation Solution

To proceed to develop the approximate solution let us consider problem (2.25) in which $\frac{\mu\, d_\ell c_\ell(\mathbf{P})-1}{\mu\, d_\ell c_\ell(\mathbf{P})-1+\beta}$ is a linear rational function ($f(z)=\frac{z}{z+\beta}$) of variable $z = \mu\, d_\ell c_\ell(\mathbf{P})-1$. This function can be approximated as a constant for values of z that are 3-4 times larger than β. This approximation is a valid assumption for the cases in which the delay requirements of the sessions are not set too tight (e.g., $z \geq 4\beta$ or $d_l \geq \frac{4\beta+1}{\mu c_l(\mathbf{P})}$). Therefore problem (2.25) can be approximated as

$$\max_{\mathbf{P}} \sum_{\ell} \lambda_\ell c_\ell(\mathbf{P}). \tag{2.29}$$

And again by deploying gradient method the power update equation will be

$$P_\ell(t+1) = P_\ell(t) + \kappa_p \, \nabla_\ell I(\mathbf{P}), \tag{2.30}$$

where

$$I(\mathbf{P}) = \sum_{\ell} \lambda_\ell c_\ell(\mathbf{P}), \tag{2.31}$$

and

$$\nabla_\ell I(\mathbf{P}) = \frac{\lambda_\ell}{P_\ell} - \sum_{j \neq \ell} \frac{\lambda_j G_{j\ell}}{\sum_{k \neq j} G_{jk} P_k + n_j}. \tag{2.32}$$

2.3.3 Discussion and Summarization of the Algorithm

Note that the exact expression for $\nabla_\ell I(\mathbf{P})$, (2.28), is a function of link delays (d_ℓ's, d_j's) unlike the approximate expression for $\nabla_\ell I(\mathbf{P})$, (2.32). Therefore, this is the reason why the message passing overhead is lower in the approximate case. In other words, the information of the delay distribution on each link (d_ℓ's, d_j's) does not need to be available to the transmitters to update their power allocation (P_ℓ).

Using the power allocation that results from the *power allocation* problem, and calculating c_ℓ, $\forall \ell$, the *delay distribution* problem, which is the minimization in (2.23) where the optimization variable is $\mathbf{d}$, can be performed. The *delay distribution* problem in (2.23), with the assumption of constant c_ℓ, is a convex optimization problem and therefore there exists a unique global optimum.

Therefore in order to solve the *link* problem (2.20), we first solve the *power allocation* problem, and then the *delay distribution* problem in (2.23), using iterative methods.

We summarize the algorithm below for the approximate *power allocation* case. For the case of exact *power allocation*, the algorithm can be similarly derived.

Algorithm I: Rate-Power-Delay I

Initialize $\lambda_\ell, \forall \ell$, and $t = 0$.

Repeat the following steps until convergence:

(1) $t = t + 1$. Every session finds its rate x_s by maximizing its own net utility:

$$\max_{x_s \geq R_s} U_s(x_s) - x_s \sum_{\ell \in L(s)} \lambda_\ell(t). \tag{2.33}$$

(2) Each transmitter calculates a message $m_j(t)$ based on values that can be determined at each node and passes it to all other transmitters by a flooding protocol:

$$m_j(t) = \frac{\lambda_j(t)\mathtt{SIR}_j(t)}{P_j(t)G_{jj}}. \tag{2.34}$$

(3) Each transmitter updates its power based on

$$P_\ell(t+1) = P_\ell(t) + \kappa_p \left(\frac{\lambda_\ell(t)}{P_\ell(t)} - \sum_{j \neq \ell} G_{j\ell} m_j(t) \right). \tag{2.35}$$

(4) Solve the centralized network problem:

$$\min_{\mathbf{d}} \sum_\ell \lambda_\ell \left(\frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell}} \right)$$
$$\text{subject to:} \quad d_\ell \geq \frac{1}{\mu\, c_\ell l}, \ \forall \ell; \ \sum_{\ell \in L(s)} d_\ell \leq D_s, \ \forall s. \tag{2.36}$$

(5) Each link updates its link price as

$$\lambda_\ell(t\ +\ 1)\ \ =\ \ \left[\lambda_\ell(t)\ -\ \kappa_1\left(c_\ell(t)\ -\ \frac{\beta}{\mu\, d_\ell(t) - \frac{1-\beta}{c_\ell(t)}}\ -\ \sum_{s:\ \ell\in L(s)} x_s(t)\right)\right]^+ . \quad (2.37)$$

This and other decentralized algorithms (e.g., [Chi05], [PDE08]) are achieved at the expense of some message passing which might not always be very desirable in some implementations. But one could develop algorithms with lower message passing overhead at the expense of losing optimality (e.g., see [PDE08]). These sub-optimal approximate algorithms are beyond the scope of this dissertation.

2.3.4 Decentralizing the Network Problem

Now let us consider how to decentralize Step (4) of

`Algorithm I`. This will lead to `Algorithm II`. We use the dual-decomposition method in order to solve the *delay distribution* problem in (2.36). The dual decomposition method will result in an optimum solution in the primal problem (2.36) because of the zero duality gap. We introduce Lagrange multipliers $\boldsymbol{\nu} > 0$ to formulate the Lagrange dual function corresponding to the primal problem in (2.36) as follows:

$$L_2(\mathbf{d}, \boldsymbol{\nu}) = \sum_\ell \lambda_\ell \left(\frac{\beta}{\mu\, d_\ell - \frac{1-\beta}{c_\ell}}\right) + \sum_s \nu_s \left(\sum_{\ell\in L(s)} d_\ell - D_s\right). \quad (2.38)$$

Again note that, in this Lagrangian function, we just relaxed the second constraint in (2.36). The Lagrange dual function is

$$Q_2(\boldsymbol{\nu}) = \min_{d_\ell \geq 1/\mu\, c_\ell, \forall \ell} L_2(\mathbf{d}, \boldsymbol{\nu}). \quad (2.39)$$

The dual problem can be formulated as

$$\min Q_2(\boldsymbol{\nu}) \text{ subject to: } \boldsymbol{\nu} > 0. \tag{2.40}$$

To solve the dual problem, we first solve (2.39). Because of linearity of the differentiation operator, we decompose this problem to be solved on each link separately:

$$\min_{d_\ell \geq \frac{1}{\mu c_\ell}} \frac{\lambda_\ell \beta}{\mu d_\ell - \frac{1-\beta}{c_\ell}} + d_\ell \sum_{s \in S(\ell)} \nu_s. \tag{2.41}$$

Then, the dual problem in (2.40) is solved by using the gradient projection algorithm as

$$\nu_s(t+1) = \left[\nu_s(t) - \kappa_2 \left(D_s - \sum_{\ell \in L(s)} d_\ell \right) \right]^+, \tag{2.42}$$

where κ_2 is the step size.

We summarize the new algorithm with the decentralized delay distribution below.

`Algorithm II: Rate-Power-Delay II`

`Initialize` $\lambda_\ell, \forall \ell$, `and` $t = 0$.

`Repeat the following steps until convergence:`

`Steps (1-3) same as steps (1-3) in Algorithm I`

`(4) Solve the decentralized network problem for the delay distribution:`

`Initialize` $\nu_s, \forall s$, `and` $k = 0$.

`Repeat the following steps until convergence:`

- `(4-1)` $k = k + 1$. `Every link updates its own delay share by`

$$\min_{d_\ell \geq \frac{1}{\mu c_\ell}} \frac{\lambda_\ell \beta}{\mu d_\ell - \frac{1-\beta}{c_\ell}} + d_\ell \sum_{s \in S(\ell)} \nu_s. \tag{2.43}$$

- `(4-2) The delay price is updated as`

$$\nu_s(k+1) = \left[\nu_s(k) - \kappa_2 \left(D_s - \sum_{\ell \in L(s)} d_\ell(k)\right)\right]^+ . \tag{2.44}$$

`(5) Each link updates its link price as`

$$\lambda_\ell(t\ +\ 1) \quad = \quad \left[\lambda_\ell(t)\ -\ \kappa_1\left(c_\ell(t)\ -\ \frac{\beta}{\mu\, d_\ell(t) - \frac{1-\beta}{c_\ell(t)}}\ -\ \sum_{s:\ \ell \in L(s)} x_s(t)\right)\right]^+ . \tag{2.45}$$

2.3.5 Concluding Remarks on the Algorithms

Note that, there are similarities between some of the steps of `Algorithm I` and `Algorithm II` and the algorithms that have been developed in [Chi05] and [SLGY07]. However, combining the two problems in the two aforementioned papers and decentralizing the delay distribution problem

(`Algorithm II`) have not been developed before. It is also worth mentioning that Step (4) of `Algorithm II` is done at a faster time-scale than the other steps since the convergence of Step (4) is required before moving to Step (5). In other words, the link share of delay distribution in Step (4-1), and the corresponding price update in Step (4-2) are done more frequently than the congestion control and power control. On the other hand, the time needed for the channel coding to achieve c_ℓ is assumed to be faster than all the mentioned time scales. Furthermore, it is assumed that the network topology and routing will not change any faster than all the other mentioned time scales.

The delay requirements of the sessions are reflected in the parameters D_s and in the constraint $\sum_{\ell \in L(s)} d_\ell \leq D_s$. This requirement cannot be too tight since the

condition $d_\ell \geq 1/\mu\, c_\ell(\mathbf{P})$ should also hold. Moreover, for the approximate solution to the *power allocation* problem to be accurate, $d_l \geq \frac{4\beta+1}{\mu c_l(\mathbf{P})}$ should hold. Therefore, the d_ℓ values cannot be lowered arbitrarily without any bound. Also, increasing the power of the transmitters without any bounds will not result in unbounded capacity, because the capacities of the links are interference limited.

2.4 Simulations

In the simulations, the utility function for all sessions s is taken as $U_s(x_s) = \alpha_s \log x_s$, which is an increasing, strictly concave function of the session rate x_s. It is also the utility function of TCP Vegas. We examine the performance of `Algorithm II` in some simple wireless network configurations, comparing the resulting rate and power allocations with the allocations provided by the algorithm in [Chi05] which does not include average queueing delay constraints. In addition, we explore how the power and rate allocations change as the average queueing delay constraints are varied. In congestion control with TCP Vegas, the utility function's parameter α_s is typically set to the same constant for all sessions in order to achieve proportional fairness of network rates. However, proportional fairness no longer remains true in the NUM problem (2.13) formulation because of the inclusion of the sessions' average queueing delay as constraints. Therefore, in the simulations, we compare cases of all sessions possessing the same value of α_s with cases in which some sessions change their value of α_s in order to reflect their preference for high data rate. In particular, we examine cases of greedy sources desiring both high rate and low delay, sources possessing a preference for low delay, and sources possessing a preference for high rate. In these comparisons, we consider two different simple wireless network configurations with 5

nodes and 4 sessions. See Figs. 2.1 and 2.2.

In particular, in Set A of the simulations, we simulate a case of sessions with no delay requirements and another case with different delay requirements. The results are then compared and the impact of the delay requirements are studied. In Set B of the simulations, we study how different α_s values of different sessions affect the rate-power allocation. In Set C of the simulations, the relationship between the rate-power allocations and flow topology is studied. In the captions of the tables, $\boldsymbol{\alpha_s}$ represents the vector of α_s pertaining to different sessions, i.e., $\boldsymbol{\alpha_s} = [\alpha_1, \alpha_2, \alpha_3, \alpha_4]$.

2.4.1 Set A: Changing the Delay Requirement

Here we consider the effect of delay requirements on the rate and power allocation for the network and flow topology I in Fig. 2.1. We start with a scenario — Scenario 1 — in which all the sessions impose the same queuing delay requirement and have the same preference for data rate in terms of their utility function. Therefore, the average queueing delay requirement D_s for all of the sessions is initially set at 10ms; $\alpha_s = 1$ for all the sessions. A maximum power of 0.12W is set as a constraint as well. We now compare the results of `Algorithm II` with the case of no delay requirements, i.e., the algorithm derived in [Chi05], to see how the delay requirements would change the rate-power allocations.

Fig. 2.3 shows the convergence of the rates of different sessions both for the case of the algorithm with and without delay constraints. Comparing the rates of each individual session obtained by `Algorithm II` (with delay constraints) with the rates obtained by the algorithm in [Chi05], it is clear that the sessions running `Algorithm II` reduced their data rates in order to achieve lower queueing delays based on the con-

straints. Fig. 2.4 shows the transmitter powers at each of the links for the case with delay constraints (`Algorithm II`) and the case of no delay constraints. As can be seen from these plots, for the case with delay constraints (`Algorithm II`), the transmitters at link 2 and link 4 are consuming slightly more power to handle the delay requirements of the sessions.

Fig. 2.5 shows the achieved delay for each session. The requirement of 10ms delay is achieved by all the sessions. The numerical values of these simulations are shown in Scenario 1 in Table 2.1. Three other different delay requirements for Session 2 (8ms, 5ms, and 2ms) are also simulated and the results are presented as Scenarios 2, 3, and 4 in Table 2.1. Session 2 is chosen for this experiment since it flows through the largest number of links. It is also notable that the total queuing delay for Session 2 is the sum of the queuing delays of Sessions 1 and 3 because of the flow and link topology. Therefore, Session 2 imposes the most difficult queuing delay requirement. We see that, in order to achieve lower queuing delay for Session 2, link 2 increases its transmission power ($0.111 \rightarrow 0.112 \rightarrow 0.115 \rightarrow 0.120$) to increase its capacity. Moreover, link 4 decreases its transmission power ($0.060 \rightarrow 0.059 \rightarrow 0.058 \rightarrow 0.051$) to lower its interference on links 2 and 3 since the delay requirement of Session 4 is already achieved and lowering the capacity of link 4 will not violate the delay requirement of Session 4. In addition, we see that all the sessions in Scenarios 2-4 have decreased their sending rate in order to obtain a lower queuing delay for Session 2.

When the exact *power allocation* algorithm is used, the achieved data rates are only slightly different from the rates achieved by the approximate power algorithm in `Algorithm II`. Notice the comparison in Scenarios 1 and 2 of Table 2.1. The power

and delay for the exact power allocation algorithm are essentially the same as those in the approximate algorithm for these scenarios. We do not repeat these numbers for space reasons. The condition $d_l \geq \frac{4\beta+1}{\mu c_l(\mathbf{P})}$ that is required for the approximate power allocation to be more accurate holds in these scenarios.

2.4.2 Set B: Changing α_s Together with the Delay Requirement

In the previous set of simulations, all of the sources kept $\alpha_s = 1$ in their utility function. Now we consider the cases of allowing a utility to change its value of α_s in order to reflect its preference for a high data rate. In particular, we examine the performance of `Algorithm II` as both the α_s values and the delay constraints are changed. First, we consider a situation in which Session 2 emphasizes a preference for higher rate than the rest of the sessions. Specifically, we use $\boldsymbol{\alpha_s} = [1, 2, 1, 1]$. As Session 2 tightens its delay requirement in the Scenarios 5-8 depicted in Table 2.2, it turns itself into a greedy source which requires both high rate and low delay. Comparing the rate and power allocation results of Scenarios 5-8 with the rate and power allocation results from Scenarios 1-4, one notices that Session 2 achieves a higher data rate at the expense of lower data rates for other sessions and higher power for the transmitter on link 2, while also satisfying its low delay requirement. Furthermore, in Scenario 9, we consider a case of two sessions — Sessions 3-4 — requiring lower delay than they require in Scenario 5. But since they set their α_s values to 1, they are not greedy as far as expecting high rate as well. Comparing the results in Scenario 9 with those in Scenario 5, we see that Sessions 3-4, that require lower delay, achieve it at the expense of a lower rate, and also a reduced rate for

Session 2.

In Scenario 10 in Table 2.3, we consider the case where two sessions — Sessions 3-4 — require a higher rate but not very low delay and another session — Session 2 — requires lower delay but not a higher rate. Therefore Sessions 3-4 set their α_s value at 2 while the rest of the sessions keep it at 1. Comparing the results of Scenario 10 with those of Scenario 3, we infer that the higher rate requirements of Sessions 3-4 are achieved at the expense of lower rates for other sessions while the delay requirement of Session 2 is also met.

2.4.3 Set C: A Different Flow Topology

Simulations similar to Table 2.1 are generated for the network and flow topology II in Fig. 2.2. The corresponding results are shown in Table 2.4. The same observations that were made in Set A of the simulations can also be made here, i.e., we see that tightening the delay requirement of Session 2 in Scenarios 11-14 imposes lower rates on all the sessions and different power allocation for the transmitters. We also emphasize another point by the network and flow topology II: the result of rate-power allocation for any particular session is highly interdependent with the topology of the other flows. For instance, Session 1 in flow topology II resembles Session 4 in flow topology I. But the results of the power and rate allocations for Session 4 in the flow topology I (see Table 2.1) is very different from the results for Session 1 in the flow topology II (see Table 2.4).

In summary, these simulations demonstrate the potential benefits of `Algorithm II` which provides distributed rate and power control for a multi-hop wireless network while also providing average queueing delay bounds for sessions. Moreover, the

simulations also show the potential for different sessions obtaining their preferences for high rate or low delay. Our current research work involves the study of convexity issues if the general SIR scenario is considered as well as the fairness and feasibility of the algorithm developed here.

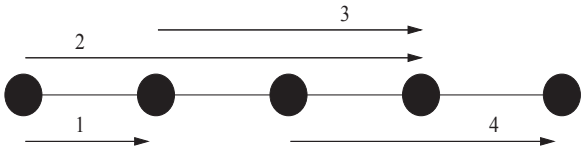


Figure 2.1: Network and flow topology I.

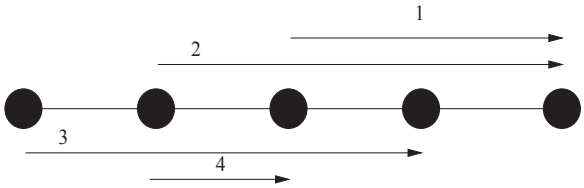


Figure 2.2: Network and flow topology II.

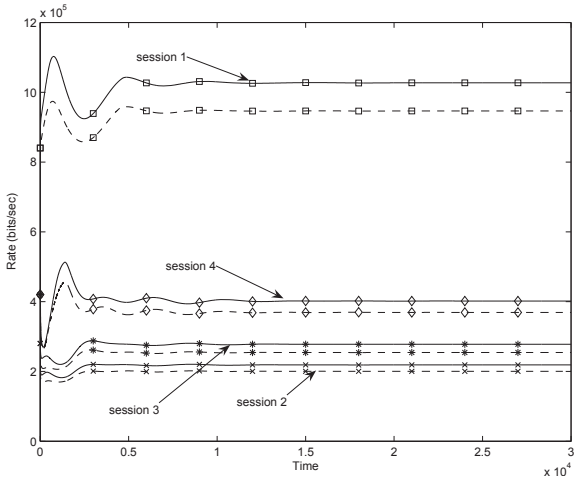


Figure 2.3: Convergence for the rates of 4 sessions. Solid curves are for the case of no delay constraint and dashed curves are for the case with delay constraints.

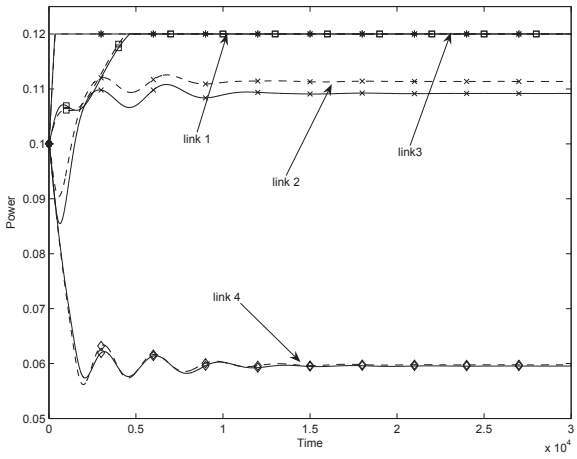


Figure 2.4: Convergence for the transmission power of 4 links. Solid curves are for the case of no delay constraint and dashed curves are for the case with delay constraints.

2.5 Remarks

We presented the problem of allocating resources of bandwidth and power in a multi-hop wireless network in a NUM framework, specifically incorporating an average queuing delay requirement of the sessions into the NUM problem. We transformed the non-convex problem to a convex problem by a change of variable and assuming a high SIR scenario. We presented a distributed iterative algorithm solving the NUM problem. In particular, in our solution algorithm, both sources and links exchange information allowing for the NUM problem to solve for the session rate, power of transmitters, and delay share of each link in an iterative distributed manner. The simulations showed the performance of the solution and comparisons with the previously developed algorithms and demonstrated that average queuing delay requirement of the sessions were achieved.

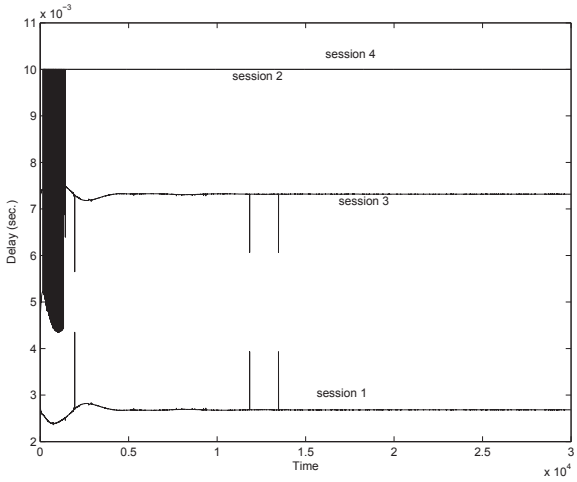


Figure 2.5: Convergence for the queueing delay of four sessions.

Table 2.1: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 1, 1, 1]$

	Session #			
	1	2	3	4
Scenario 1				
Delay Requirement (ms)	10.0	10.0	10.0	10.0
Delay Achieved (ms)	2.7	10.0	7.3	10.0
Data Rate (Kb/s)	946.5	200.4	254.3	369.2
Data Rate (Kb/s) (exact power allocation)	**946.6**	**200.5**	**254.4**	**368.9**
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.111	0.120	0.060
Scenario 2				
Delay Requirement (ms)	10.0	8.0	10.0	10.0
Delay Achieved (ms)	2.1	8.0	5.9	10.0
Data Rate (Kb/s)	921.9	195.7	248.4	362.5
Data Rate (Kb/s) (exact power allocation)	**922.1**	**195.7**	**248.5**	**362.5**
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.112	0.120	0.059
Scenario 3				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	1.3	5.0	3.7	10.0
Data Rate (Kb/s)	843.5	179.9	228.7	338.8
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.115	0.120	0.058
Scenario 4				
Delay Requirement (ms)	10.0	2.0	10.0	10.0
Delay Achieved (ms)	0.5	2.0	1.5	10.0
Data Rate (Kb/s)	413.4	83.9	105.3	199.5
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.116	0.120	0.120	0.051

Table 2.2: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 2, 1, 1]$

	Session #			
	1	2	3	4
Scenario 5				
Delay Requirement (ms)	10.0	10.0	10.0	10.0
Delay Achieved (ms)	2.6	10.0	7.4	10.0
Data Rate (Kb/s)	786.0	317.6	199.0	295.1
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.118	0.120	0.056
Scenario 6				
Delay Requirement (ms)	10.0	8.0	10.0	10.0
Delay Achieved (ms)	2.1	8.0	5.9	10.0
Data Rate (Kb/s)	764.7	310.2	194.6	289.9
	Link 1	Link 2	Link 3	Link 4
Power	0.120	0.118	0.120	0.056
Scenario 7				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	1.3	5.0	3.7	10.0
Data Rate (Kb/s)	700.0	285.1	179.0	272.6
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.120	0.120	0.055
Scenario 8				
Delay Requirement (ms)	10.0	2.0	10.0	10.0
Delay Achieved (ms)	0.5	2.0	1.5	10.0
Data Rate (Kb/s)	326.1	133.4	83.9	168.3
	Link 1	Link 2	Link 3	Link 4
Power	0.112	0.120	0.120	0.050
Scenario 9				
Delay Requirement (ms)	10.0	10.0	5.0	5.0
Delay Achieved (ms)	5.0	10.0	5.0	5.0
Data Rate (Kb/s)	835.4	305.6	187.0	268.8
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.118	0.120	0.058

Table 2.3: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 1, 2, 2]$

	Session #			
	1	2	3	4
Scenario 10				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	1.2	5.0	3.8	10.0
Data Rate (Kb/s)	773.1	119.9	283.8	381.2
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.105	0.108	0.120	0.060

Table 2.4: Simulation results for network and flow topology II with $\boldsymbol{\alpha_s} = [1, 1, 1, 1]$

	Session #			
	1	2	3	4
Scenario 11				
Delay Requirement (ms)	10.0	10.0	10.0	10.0
Delay Achieved (ms)	6.5	10.0	10.0	3.5
Data Rate (Kb/s)	347.3	192.9	197.3	433.9
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.054	0.120	0.120	0.073
Scenario 12				
Delay Requirement (ms)	10.0	8.0	10.0	10.0
Delay Achieved (ms)	5.0	8.0	10.0	3.0
Data Rate (Kb/s)	334.2	188.8	194.8	434.1
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.052	0.120	0.120	0.074
Scenario 13				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	3.2	5.0	10.0	1.8
Data Rate (Kb/s)	297.9	173.5	181.7	415.7
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.051	0.120	0.120	0.077
Scenario 14				
Delay Requirement (ms)	10.0	2.0	10.0	10.0
Delay Achieved (ms)	1.3	2.0	10.0	0.7
Data Rate (Kb/s)	125.7	83.3	92.9	247.1
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.046	0.120	0.120	0.092

CHAPTER 3

Quantization for Classification Accuracy in High-Rate Quantizers

Quantization of signals is required for many transmission, storage and compression applications. The original signal is quantized at the encoder side. At the decoder side, a replica of the original signal that should resemble the original signal in some sense is recovered. Present quantizers make an effort to reduce the distortion of the signal in the sense of reproduction fidelity. Consider scenarios in which signals are generated from multiple classes. The encoder focuses on the task of quantizing the data without any regards to the class of the signal. The quantized signal reaches the decoder where not only the recovery of the signal should take place but also a decision should be made on the class of the signal based on the quantized version of the signal only. In this dissertation, we study the design of such a vector quantizer that is optimized for the task of classification at the decoder. We define the distortion to be the symmetric Kullback-Leibler (KL) divergence measure between the conditional probabilities of class given the signal before and after quantization. A high-rate analysis of the quantizer is presented and the optimum point density of the quantizer for minimizing the symmetric KL divergence is derived. Furthermore, tradeoff between classification

accuracy and reproduction fidelity is studied and a point density function for the case of tradeoff is presented. Effects of a mismatched distortion measure is also derived and shown that for reduced complexity the original distortion measure can be replaced by a weighted mean square error (WMSE) distortion measure. The performance of these methods on synthetically generated data and real data set is examined and observed to be superior in the task of classification of signals at the decoder with lower reproduction fidelity as tradeoff.

3.1 Overview of Quantization and Classification

Quantization of signals has been extensively studied [GN98, GG92]. The quality of a quantizer is measured by the similarity of the reproduction signal with the original signal. This similarity is measured by a distortion measure which is defined differently for different purposes. Some of the most common distortion measures are Mean Squared Error (MSE), weighted mean square error (WMSE) and log spectral distortion (LSD) [GR95]. These measures usually guarantee the fidelity of the reproduction but heed less attention to other utilizations of the reproduction signal, for instance detection or classification of the signal at the decoder. Several studies have been done in combining detection and quantization with different distortion measures [Kas77, SS77, GH03]. In [VB10], the authors study quantization for detection in a sensor network scenario. In [VV08], the quantization of prior probabilities for hypothesis testing is presented. These studies have made significant contributions to the design of quantizers for the purpose of detection. However, most of them are applicable to binary classification, i.e., binary detection or hypothesis testing, and either are not expandable to multiple class problems or the extension is not straightforward, to the

best of our knowledge. There has been some research on combining quantization and classification. In [PPG+96], the authors extend the distortion measure for the Lloyd algorithm by a Bayes risk term which is defined by a weighted cost of classification error. Moreover, in [LJ07], the authors proposed a method for joint quantization and classification in distributed quantizers in which the quantizers sequentially process an input and communicate with each other. Despite the aforementioned contributions, the theory of high-rate quantization for the purpose of classification still remains somewhat underdeveloped compared to quantization for detection, or quantization theory in general.

In this dissertation, we consider scenarios in which signals are generated from multiple classes. The encoder focuses on the task of quantizing the data without any regards to the class of the signal. The quantized signal reaches the decoder where not only the recovery of the signal should take place but also a decision is to be made on the class of the signal based on the quantized version of the signal only. We study the design of such vector quantizer that is optimized for the task of classification at the decoder. We adopt a high-rate quantizer design approach [GH03, GR95, VB10]. A high-rate quantizer is an optimal quantizer in the sense of minimizing the distortion with unlimited quantization levels. Therefore an optimal point density function which represents how codepoints are distributed in the space is derived. We choose the distortion to be the symmetric Kullback-Leibler (KL) divergence measure between the conditional probabilities of class given the signal before and after quantization. Our approach in using the KL divergence measure is different than some other related research, notably [GH03, LJ07]. To be specific, others have tried to minimize the loss in the discrimination between the two classes due to quantization. On the other hand,

we try to minimize the aggregate discrimination of the conditional probabilities of classes given the signal before and after quantization, i.e., equation (3.2). Intuitively, it means that we will make sure that the conditional probability of the classes given the data undergoes the least amount of change due to quantization. The tradeoff between the reproduction fidelity and classification accuracy at the decoder is also examined.

In the following we derive the optimum point density function using high-rate theory for a quantizer with the new distortion measure in Section 3.2. In Section 3.3, we design a quantizer that performs a tradeoff between classification accuracy and reproduction fidelity. In Section 3.5, for some signal examples we will compare the optimal point densities developed in this research with the MSE point density and compare the classification results at the decoder. We observe that the quantizer based on the KL divergence point density outperforms the traditional quantizer based on the MSE point density with regards to the classification accuracy. Some of the results of this chapter appeared in [DM11].

3.2 Classification Based Quantizer Design

Consider the input to an encoder, $\mathbf{x}$, to be an n dimensional signal in $D \subset R^n$ with probability density function $p(\mathbf{x})$. The output of the encoder is $Q(\mathbf{x})$ which maps $\mathbf{x}$ to an output vector $\hat{\mathbf{x}}$ which takes values from $\hat{\mathbf{X}} = \{\hat{\mathbf{x}}_1, ...\hat{\mathbf{x}}_{2^B}\}$ where B is the number of bits in the quantizer. The expected distortion that is introduced by the quantizer can be expressed as

$$E_d(Q) = \int_{\mathbf{x}\in D} d(\mathbf{x}, \hat{\mathbf{x}}) p(\mathbf{x}) d\mathbf{x}, \tag{3.1}$$

where d is the distortion measure. The problem of designing a quantizer is to minimize the expected distortion between the original signal, $\mathbf{x}$, and the reconstructed signal, $\hat{\mathbf{x}}$. Since our main objective is to design a quantizer that improves the classification accuracy of the signal at the decoder, our distortion measure is defined as the symmetric KL divergence measure between the conditional probabilities of class given the signal before and after quantization. The symmetric KL divergence is always non-negative, i.e., $d(\mathbf{x}, \hat{\mathbf{x}}) \geq 0$. Therefore d is defined as

$$d(\mathbf{x}, \hat{\mathbf{x}}) = D\Big(P(C|\mathbf{x})||P(C|\hat{\mathbf{x}})\Big) + D\Big(P(C|\hat{\mathbf{x}})||P(C|\mathbf{x})\Big) = \sum_{j=1}^{C}\left[P(C_j|\mathbf{x})\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})} + P(C_j|\hat{\mathbf{x}})\log\frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\right], \quad (3.2)$$

where $P(C|\mathbf{x})$ and $P(C|\hat{\mathbf{x}})$ are conditional probabilities of the classes given the data before and after quantization, respectively. We follow high-rate theory analysis in which an optimal quantizer in the sense of minimizing the defined distortion is derived. The optimal quantizer is represented by an optimal point density function which determines where in the space should the codepoints lie. This approach is similar to the one in [GR95]. If we perform a Taylor series expansion of $d(\mathbf{x}, \hat{\mathbf{x}})$ about $\mathbf{x} = \hat{\mathbf{x}}$ and ignore all the terms but the first three terms, we conclude

$$d(\mathbf{x}, \hat{\mathbf{x}}) \approx d(\hat{\mathbf{x}}, \hat{\mathbf{x}}) + \mathbf{d}(\hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}}) + \frac{1}{2}(\mathbf{x} - \hat{\mathbf{x}})^T\mathbf{D}(\hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}}) \quad (3.3)$$

where $\mathbf{d}(\hat{\mathbf{x}})$ is an n-element row vector whose elements are defined by

$$d_j(\hat{\mathbf{x}}) = \left.\frac{\partial d(\mathbf{x}, \hat{\mathbf{x}})}{\partial x_j}\right|_{\mathbf{x}=\hat{\mathbf{x}}} \quad (3.4)$$

and $\mathbf{D}(\hat{\mathbf{x}})$ is an $n \times n$ element matrix whose elements are

$$D_{j,k}(\hat{\mathbf{x}}) = \left.\frac{\partial^2 d(\mathbf{x}, \hat{\mathbf{x}})}{\partial x_j \partial x_k}\right|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad (3.5)$$

Based on the definition of the distortion d in (3.2) the first term in (3.3) is equal to zero. Furthermore, by calculations similar to the ones in Section 3.2.1 it can easily be seen that the second term is also equal to zero (See Appendix A). If we substitute the non-zero term from (3.3) into (3.1) we obtain

$$E_d(Q) \approx \int_{\mathbf{x}\in D} \frac{1}{2}(\mathbf{x}-\hat{\mathbf{x}})^T \mathbf{D}(\hat{\mathbf{x}})(\mathbf{x}-\hat{\mathbf{x}})p(\mathbf{x})d\mathbf{x}. \tag{3.6}$$

Let us define $S_i = \{\mathbf{x} \in D | Q(\mathbf{x}) = \hat{\mathbf{x}}_i\}$. Therefore, each data point that lies in the region S_i is mapped into $\hat{\mathbf{x}}_i$ as a result of the quantization. Therefore (3.6) can be written as

$$E_d(Q) \approx \sum_{\hat{\mathbf{x}}_i \in \hat{\mathbf{X}}} \int_{\mathbf{x}\in S_i} \frac{1}{2}(\mathbf{x}-\hat{\mathbf{x}}_i)^T \mathbf{D}(\hat{\mathbf{x}}_i)(\mathbf{x}-\hat{\mathbf{x}}_i)p(\mathbf{x})d\mathbf{x}. \tag{3.7}$$

Assuming a high-rate quantizer, we reason that $p(\mathbf{x}) \approx p(\hat{\mathbf{x}}_i); \forall \mathbf{x} \in S_i$. Hence

$$E_d(Q) \approx \sum_{\hat{\mathbf{x}}_i \in \hat{\mathbf{X}}} p(\hat{\mathbf{x}}_i) \int_{\mathbf{x}:\mathbf{x}+\hat{\mathbf{x}}_i \in S_i} \frac{1}{2}\mathbf{x}^T \mathbf{D}(\hat{\mathbf{x}}_i)\mathbf{x}d\mathbf{x}. \tag{3.8}$$

The volume of the Voronoi region S_i can be expressed as $\mathrm{vol}(S_i) = \int_{\mathbf{x}\in S_i} d\mathbf{x}$. And if the probability of vector $\mathbf{x}$ lying in region S_i is P_i then

$$P_i = \text{Probability}(\mathbf{x} \in S_i) \approx p(\hat{\mathbf{x}}_i)\mathrm{vol}(S_i). \tag{3.9}$$

Using this definition and the following equality from [GR95]

$$\int_{\mathbf{x}\in S_i} \mathbf{x}^T \mathbf{D}\mathbf{x}d\mathbf{x} = \mathrm{vol}(S_i)\frac{n}{n+2}\left(\frac{\mathrm{vol}(S_i)^2}{\kappa_n^2}|\mathbf{D}|\right)^{1/n}, \tag{3.10}$$

where κ_n is the volume of n-dimensional unit sphere, equation (3.8) can be expressed as

$$E_d(Q) \approx \sum_{\hat{\mathbf{x}}_i \in \hat{\mathbf{X}}} \frac{P_i}{\mathrm{vol}(S_i)} \left(\mathrm{vol}(S_i) \frac{n}{2(n+2)} \left(\frac{\mathrm{vol}(S_i)^2}{\kappa_n^2} |\mathbf{D}(\hat{\mathbf{x}}_i)| \right)^{1/n} \right) =$$

$$\sum_{\hat{\mathbf{x}}_i \in \hat{\mathbf{X}}} \frac{n}{2(n+2)} P_i \left(\frac{\mathrm{vol}(S_i)^2}{\kappa_n^2} |\mathbf{D}(\hat{\mathbf{x}}_i)| \right)^{1/n}$$

Let us define the point density function

$$\lambda(\mathbf{x}) = \lim_{B \to \infty} \frac{1}{2^B \mathrm{vol}(S_i)}, \tag{3.11}$$

which represents the limiting density of the codebook vectors around $\mathbf{x}$, please refer to [GR95, Ger79]. Therefore

$$E_d(Q) \approx \frac{n}{2(n+2)} \sum_{\hat{\mathbf{x}}_i \in \hat{\mathbf{X}}} p(\hat{\mathbf{x}}_i) \mathrm{vol}(S_i) \left(\frac{|\mathbf{D}(\hat{\mathbf{x}}_i)|}{(2^B \lambda(\hat{\mathbf{x}}_i) \kappa_n)^2} \right)^{1/n} \approx$$

$$\frac{n 2^{-2B/n} \kappa_n^{-2/n}}{2(n+2)} \int_{\mathbf{x} \in D} \lambda^{-2/n}(\mathbf{x}) |\mathbf{D}(\mathbf{x})|^{1/n} p(\mathbf{x}) d\mathbf{x}. \tag{3.12}$$

In order to minimize the approximation in (3.12) with respect to $\lambda(\mathbf{x})$ and subject to the unit integral property of $\lambda(\mathbf{x})$ (see [Ger79]) we use Holder's inequality to derive the optimum point density function as

$$\lambda_{\mathrm{KL}}(\mathbf{x}) = \frac{\left(|\mathbf{D}(\mathbf{x})|^{1/n} p(\mathbf{x}) \right)^{n/(n+2)}}{\int_{\mathbf{x} \in D} \left(|\mathbf{D}(\mathbf{x})|^{1/n} p(\mathbf{x}) \right)^{n/(n+2)} d\mathbf{x}} \tag{3.13}$$

3.2.1 Derivation of Components of $\mathbf{D}(\mathbf{x})$

Based on the derivation of optimal point density function $\lambda_{\mathrm{KL}}(\mathbf{x})$ from the previous section, we need to calculate the terms of $\mathbf{D}(\mathbf{x})$ for our choice of distortion measure,

in (3.2). For the elements of $\mathbf{D}(\mathbf{x})$ we have

$$D_{i,k}(\hat{\mathbf{x}}) \;=\; \frac{\partial}{\partial x_i}\frac{\partial}{\partial x_k}\sum_{j=1}^{C}\left[P(C_j|\mathbf{x})\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}+\; P(C_j|\hat{\mathbf{x}})\log\frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad (3.14)$$

We calculate each of the two terms in the summation in (3.14) separately as follows

$$\begin{aligned}
&\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_k}\left(P(C_j|\mathbf{x})\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}\right) = \\
&\qquad\frac{\partial}{\partial x_i}\left[\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}\frac{\partial}{\partial x_k}P(C_j|\mathbf{x}) + P(C_j|\mathbf{x})\frac{\partial}{\partial x_k}\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}\right] = \\
&\qquad\qquad\frac{\partial}{\partial x_i}\left[\left(\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})\right)\left(\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}+1\right)\right] = \\
&\left(\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})\right)\left(\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}+1\right)+\frac{1}{P(C_j|\mathbf{x})}\left(\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})\right)\left(\frac{\partial}{\partial x_i}P(C_j|\mathbf{x})\right),
\end{aligned} \quad (3.15)$$

and

$$\begin{aligned}
&\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_k}\left(P(C_j|\hat{\mathbf{x}})\log\frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\right) = \\
&\quad\frac{P(C_j|\hat{\mathbf{x}})}{(P(C_j|\mathbf{x}))^2}\left(\left(\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})\right)\left(\frac{\partial}{\partial x_i}P(C_j|\mathbf{x})\right)-\; P(C_j|\mathbf{x})\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})\right).
\end{aligned} \quad (3.16)$$

Now, by substituting (3.15) and (3.16) into (3.14) we conclude

$$D_{i,k}(\hat{\mathbf{x}}) \qquad = \qquad \sum_{j=1}^{C}\left[\frac{2}{P(C_j|\mathbf{x})}\left(\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})\right)\left(\frac{\partial}{\partial x_i}P(C_j|\mathbf{x})\right)\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad (3.17)$$

In this dissertation, we assume a generative model for the classifier. Hence, $p(\mathbf{x})$ and $p(\mathbf{x}|C_j)$ are known a priori. Therefore, we use Bayes rule, $P(C_j|\mathbf{x}) = \frac{P(C_j)p(\mathbf{x}|C_j)}{p(\mathbf{x})}$, to

calculate the terms $\frac{\partial}{\partial x_i}P(C_j|\mathbf{x})$ and $\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})$ in (3.17) as follows

$$\frac{\partial}{\partial x_i}P(C_j|\mathbf{x}) = \frac{\partial}{\partial x_i}\frac{P(C_j)p(\mathbf{x}|C_j)}{p(\mathbf{x})} = P(C_j)\frac{\frac{\partial}{\partial x_i}p(\mathbf{x}|C_j).p(\mathbf{x}) - p(\mathbf{x}|C_j)\frac{\partial p(\mathbf{x})}{\partial x_i}}{(p(\mathbf{x}))^2}, \quad (3.18)$$

and similarly for $\frac{\partial}{\partial x_k}P(C_j|\mathbf{x})$. By substituting this into (3.17) and calculating at $\mathbf{x} = \hat{\mathbf{x}}$ we will have

$$D_{i,k}(\hat{\mathbf{x}}) = \sum_{j=1}^{C} P(C_j)\left[2\frac{\frac{\partial}{\partial x_i}p(\mathbf{x}|C_j).\frac{\partial}{\partial x_k}p(\mathbf{x}|C_j)}{p(\mathbf{x})p(\mathbf{x}|C_j)} - \right.$$

$$\left.\frac{2\frac{\partial}{\partial x_i}p(\mathbf{x}|C_j).\frac{\partial}{\partial x_k}p(\mathbf{x}) - 2\frac{\partial}{\partial x_k}p(\mathbf{x}|C_j).\frac{\partial}{\partial x_i}p(\mathbf{x})}{(p(\mathbf{x}))^2} + 2\frac{\frac{\partial}{\partial x_i}p(\mathbf{x}).\frac{\partial}{\partial x_k}p(\mathbf{x}).p(\mathbf{x}|C_j)}{(p(\mathbf{x}))^3}\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad (3.19)$$

Hence for the diagonal elements of $\mathbf{D}(\mathbf{x})$, we obtain

$$D_{i,i}(\hat{\mathbf{x}}) =$$

$$\sum_{j=1}^{C} P(C_j)\left[2\frac{\left(\frac{\partial}{\partial x_i}p(\mathbf{x}|C_j)\right)^2}{p(\mathbf{x})p(\mathbf{x}|C_j)} - 4\frac{\frac{\partial}{\partial x_i}p(\mathbf{x}|C_j).\frac{\partial}{\partial x_i}p(\mathbf{x})}{(p(\mathbf{x}))^2} + 2\frac{p(\mathbf{x}|C_j)\left(\frac{\partial}{\partial x_i}p(\mathbf{x})\right)^2}{(p(\mathbf{x}))^3}\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad (3.20)$$

Eventually, to conclude the final expression for the point density we need to calculate the determinant $|\mathbf{D}(\mathbf{x})|$ and substitute it in (3.13).

3.3 Tradeoff between Classification Accuracy and Reproduction Fidelity

We observed in the previous section that by defining a new distortion measure we were able to derive an optimal quantizer in the sense of minimizing the KL di-

vergence between conditional probabilities of classes given the data before and after quantization. In this section we address the tradeoff between reproduction fidelity and classification accuracy. For this purpose we define the distortion measure as follows

$$d(\mathbf{x}, \hat{\mathbf{x}}) = \alpha d_{\text{MSE}}(\mathbf{x}, \hat{\mathbf{x}}) + (1-\alpha) d_{\text{KL}}(\mathbf{x}, \hat{\mathbf{x}}) \tag{3.21}$$

where $d_{\text{MSE}} = \|\mathbf{x} - \hat{\mathbf{x}}\|^2$, i.e., reproduction fidelity measure, and d_{KL} is defined as in the previous section, i.e., the symmetric KL divergence measure. Furthermore, α is a value between zero and one that determines how much weight is put on each measure. Following similar steps as in Section 3.2 we conclude that the optimal point density function is

$$\lambda_{\text{Tradeoff}}(\mathbf{x}) = \frac{\left(\left(\alpha|\mathbf{D}_{\text{MSE}}(\mathbf{x})|^{1/n} + (1-\alpha)|\mathbf{D}_{\text{KL}}(\mathbf{x})|^{1/n}\right)p(\mathbf{x})\right)^{n/(n+2)}}{\int_{\mathbf{x}\in D}\left(\left(\alpha|\mathbf{D}_{\text{MSE}}(\mathbf{x})|^{1/n} + (1-\alpha)|\mathbf{D}_{\text{KL}}(\mathbf{x})|^{1/n}\right)p(\mathbf{x})\right)^{n/(n+2)} d\mathbf{x}}, \tag{3.22}$$

where $\mathbf{D}_{\text{MSE}}(\mathbf{x})$, and $\mathbf{D}_{\text{KL}}(\mathbf{x})$ are defined similar to $\mathbf{D}(\mathbf{x})$ in (3.5).

3.4 Optimal Mismatched Distortion Measure

In this section, we study the performance of a quantizer which is trained by minimizing the distortion $d_1(\mathbf{x}, \hat{\mathbf{x}})$ but measured by distortion $d_2(\mathbf{x}, \hat{\mathbf{x}})$. The rationale for such analysis is scenarios in which we would like to design a quantizer with distortion measure of relevance being d_2 but for practical and implementation reasons we choose to quantize with distortion measure d_1. The expression for the distortion (measure

by d_2) is as follows

$$E_{d_2}(Q_1) \approx \frac{2^{-2B/n}\kappa_n^{-2/n}}{2(n+2)} \frac{\int_{\mathbf{x}\in D}(|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)}\mathrm{tr}(\mathbf{D_1^{-1}}(\mathbf{x})\mathbf{D_2}(\mathbf{x}))d\mathbf{x}}{(\int_{\mathbf{x}\in D}(|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)})^{-2/n}} \tag{3.23}$$

Interested reader can refer to ([GR95], Section II) for details on derivation of the latter expression. Let us study the derivation of a weighted mean square error (WMSE) distortion measure as the choice for d_1. A WMSE distortion measure can be represented as

$$d_{\mathrm{WMSE}}(\mathbf{x},\hat{\mathbf{x}}) = (\mathbf{x}-\hat{\mathbf{x}})^T \begin{bmatrix} w_1(\mathbf{x}) & & & 0 \\ & w_2(\mathbf{x}) & & \\ & & \ddots & \\ 0 & & & w_n(\mathbf{x}) \end{bmatrix} (\mathbf{x}-\hat{\mathbf{x}}) \tag{3.24}$$

By comparing this expression with the quadratic part of equation (3.3), we conclude that $\mathbf{D}_{\mathrm{WMSE}}(\mathbf{x}) = 2\ \mathrm{diag}(w_1(\mathbf{x}), w_2(\mathbf{x}), \ldots, w_n(\mathbf{x}))$.

If $\mathbf{D_1}$ represents a WMSE distortion measure and therefore has a diagonal matrix form, we will show that optimal $\mathbf{D_1}(\mathbf{x})$ that minimizes (3.23) is

$$\mathbf{D}_{\mathrm{WMSE}}(\mathbf{x}) = \mathrm{diag}((\mathbf{D_2}(\mathbf{x}))_{1,1}, (\mathbf{D_2}(\mathbf{x}))_{2,2}, \ldots, (\mathbf{D_2}(\mathbf{x}))_{n,n}). \tag{3.25}$$

In other words, the optimal WMSE $\mathbf{D}$ matrix consists of the diagonal elements of $\mathbf{D_2}(\mathbf{x})$.

Let us start by observing that in equation (3.23) if elements of $\mathbf{D_1}$ are scaled by

a factor of ℓ, $E_{d_2}(Q_1)$ will not be modified. Details follow: If $\mathbf{D_1'}(\mathbf{x}) = \mathbf{D_1}(\ell\mathbf{x})$

$$\frac{(|\mathbf{D_1'}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)}\mathrm{tr}(\mathbf{D_1'^{-1}}(\mathbf{x})\mathbf{D_2}(\mathbf{x}))}{((|\mathbf{D_1'}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)})^{-2/n}} =$$
$$\frac{(\ell|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)}\mathrm{tr}(\ell^{-1}\mathbf{D_1^{-1}}(\mathbf{x})\mathbf{D_2}(\mathbf{x}))}{((\ell|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)})^{-2/n}} =$$
$$\frac{\ell^{n/(n+2)}\ell^{-1}}{(\ell^{n/(n+2)})^{-2/n}}\frac{(|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)}\mathrm{tr}(\mathbf{D_1^{-1}}(\mathbf{x})\mathbf{D_2}(\mathbf{x}))}{((|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)})^{-2/n}} =$$
$$\frac{\ell^{-2/(n+2)}}{\ell^{-2/(n+2)}}\frac{(|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)}\mathrm{tr}(\mathbf{D_1^{-1}}(\mathbf{x})\mathbf{D_2}(\mathbf{x}))}{((|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)})^{-2/n}} =$$
$$\frac{(|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)}\mathrm{tr}(\mathbf{D_1^{-1}}(\mathbf{x})\mathbf{D_2}(\mathbf{x}))}{((|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)})^{-2/n}}. \quad (3.26)$$

Therefore minimizing equation (3.23) is equivalent to the following constrained minimization problem:

$$\min_{\mathbf{D_1}(\mathbf{x})} \int_{\mathbf{x}\in D} (|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)}\mathrm{tr}(\mathbf{D_1^{-1}}(\mathbf{x})\mathbf{D_2}(\mathbf{x}))d\mathbf{x}$$
$$\text{subject to: } \int_{\mathbf{x}\in D} (|\mathbf{D_1}(\mathbf{x})|^{1/n}p(\mathbf{x}))^{n/(n+2)} = 1. \quad (3.27)$$

Substituting for the elements of $\mathbf{D_1}(\mathbf{x})$ and introducing the Lagrange multiplier λ we conclude the problem to be

$$\min_{\mathbf{D_1}(\mathbf{x})} \int_{\mathbf{x}\in D} ([\prod_{i=1}^{n} w_i(\mathbf{x})]^{1/n}p(\mathbf{x}))^{n/(n+2)} \sum_{i=1}^{n} \frac{(\mathbf{D_2}(\mathbf{x}))_{i,i}}{w_i(\mathbf{x})}d\mathbf{x} +$$
$$\lambda \int_{\mathbf{x}\in D} ([\prod_{i=1}^{n} w_i(\mathbf{x})]^{1/n}p(\mathbf{x}))^{n/(n+2)}d\mathbf{x}. \quad (3.28)$$

Following standard calculus of variations methods, we introduce the alternate solution to be $W_i(\mathbf{x}) = w_i(\mathbf{x}) + \epsilon_i\eta_i(\mathbf{x})$. By substituting the alternate solution into the problem we will have

$$I(\epsilon_1, \epsilon_2, \ldots, \epsilon_n) = \int_{\mathbf{x}\in D} ([\prod_{i=1}^{n} w_i(\mathbf{x}) + \epsilon_i\eta_i(\mathbf{x})]^{1/n}p(\mathbf{x}))^{n/(n+2)} \left[\sum_{i=1}^{n} \frac{(\mathbf{D_2}(\mathbf{x}))_{i,i}}{w_i(\mathbf{x}) + \epsilon_i\eta_i(\mathbf{x})} + \lambda\right] d\mathbf{x}. \quad (3.29)$$

Therefore the optimal $w_i(\mathbf{x})$ is found by solving the following system of equations

$$\left.\frac{\partial I}{\partial \epsilon_j}\right|_{\epsilon_j=0,\forall j} = 0 \tag{3.30}$$

Let us start by calculating the partial derivatives as follow

$$\begin{gathered}\frac{\partial I}{\partial \epsilon_j} I(\epsilon_1, \epsilon_2, \ldots, \epsilon_n) = \\ \frac{1}{n+2}(w_j(\mathbf{x}) + \epsilon_i \eta_j(\mathbf{x}))^{\frac{1}{n+2}-1}\eta_j(\mathbf{x})([\prod_{i\neq j=1}^{n} w_i(\mathbf{x}) + \epsilon_i\eta_i(\mathbf{x})]^{1/n} p(\mathbf{x}))^{n/(n+2)} \\ \left[\sum_{i=1}^{n} \frac{(\mathbf{D_2}(\mathbf{x}))_{i,i}}{w_i(\mathbf{x}) + \epsilon_i\eta_i(\mathbf{x})} + \lambda\right] - \\ ([\prod_{i=1}^{n} w_i(\mathbf{x}) + \epsilon_i\eta_i(\mathbf{x})]^{1/n} p(\mathbf{x}))^{n/(n+2)} \frac{\eta_j(\mathbf{x})(\mathbf{D_2}(\mathbf{x}))_{j,j}}{(w_j(\mathbf{x}) + \epsilon_j\eta_j(\mathbf{x}))^2}\end{gathered} \tag{3.31}$$

Applying the conditions $\epsilon_i = 0, \forall i$ will reduce the equation to

$$\begin{gathered}\left.\frac{\partial I}{\partial \epsilon_j} I(\epsilon_1, \epsilon_2, \ldots, \epsilon_n)\right|_{\epsilon_j=0,\forall j} = \\ \frac{1}{n+2}(w_j(\mathbf{x}))^{\frac{1}{n+2}-1}\eta_j(\mathbf{x})([\prod_{i\neq j=1}^{n} w_i(\mathbf{x})]^{1/n} p(\mathbf{x}))^{n/(n+2)} \\ \left[\sum_{i=1}^{n} \frac{(\mathbf{D_2}(\mathbf{x}))_{i,i}}{w_i(\mathbf{x})} + \lambda\right] - \\ ([\prod_{i=1}^{n} w_i(\mathbf{x})]^{1/n} p(\mathbf{x}))^{n/(n+2)} \frac{\eta_j(\mathbf{x})(\mathbf{D_2}(\mathbf{x}))_{j,j}}{(w_j(\mathbf{x}))^2} = \\ \eta_j(\mathbf{x})([\prod_{i\neq j=1}^{n} w_i(\mathbf{x})]^{1/n} p(\mathbf{x}))^{n/(n+2)} w_j(\mathbf{x}))^{\frac{1}{n+2}-2} \\ \left[\frac{1}{n+2} w_j(\mathbf{x}) \left[\sum_{i=1}^{n} \frac{(\mathbf{D_2}(\mathbf{x}))_{i,i}}{w_i(\mathbf{x})} + \lambda\right] - (\mathbf{D_2}(\mathbf{x}))_{j,j}\right]\end{gathered} \tag{3.32}$$

And finally applying the condition of the latter equation be equal to zero will result in

$$\frac{1}{n+2} w_j(\mathbf{x}) \left[\sum_{i=1}^{n} \frac{(\mathbf{D_2}(\mathbf{x}))_{i,i}}{w_i(\mathbf{x})} + \lambda\right] - (\mathbf{D_2}(\mathbf{x}))_{j,j} = 0 \tag{3.33}$$

It is trivial to verify that $w_i(\mathbf{x}) = \frac{2(\mathbf{D_2}(\mathbf{x}))_{i,i}}{\lambda}, \forall i$ is the solution to this system of equations. As we mentioned before, since the scaling of the elements of $\mathbf{D_1}(\mathbf{x})$ does not matter, the main result is that the optimal diagonal matrix $\mathbf{D_1}(\mathbf{x})$ that approximates $\mathbf{D_2}(\mathbf{x})$ consists of the diagonal elements of $\mathbf{D_2}(\mathbf{x})$, i.e., $(\mathbf{D_2}(\mathbf{x}))_{i,i}, \forall i$.

3.5 Simulations

In this section, we will present a few examples to visualize the effect of different choices of distortion measure on how the optimal point density looks like and behaves for 1-D, and 2-D synthetic data as well as a real data set. Specifically, we are interested in the behavior of the point density function around the boundaries of classification. Furthermore, we will study the effect of the tradeoff parameter, i.e, α, introduced in Section 3.3, on the point density function. In this section we present some examples to demonstrate the efficacy of the vector quantizer designed in the previous sections. Specifically, we show the reader several examples of either synthetic or real world signals which are the subjects of quantization and classification at the decoder. For the choice of synthetic data we choose Gaussian signals with different dimensions. For the choice of real world signal we use the Iris Data Set from the well known database in the pattern recognition literature, UCI's Machine Learning Repository. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant.

3.5.1 Scalar Signals

In this section, we will present a few examples to visualize the effect of different choices of distortion measure on how the optimal point density looks like and

behaves. Specifically, we are interested in the behavior of the point density function around the boundaries of classification. Furthermore, we will study the effect of the tradeoff parameter, i.e, α, introduced in Section 3.3, on the point density function.

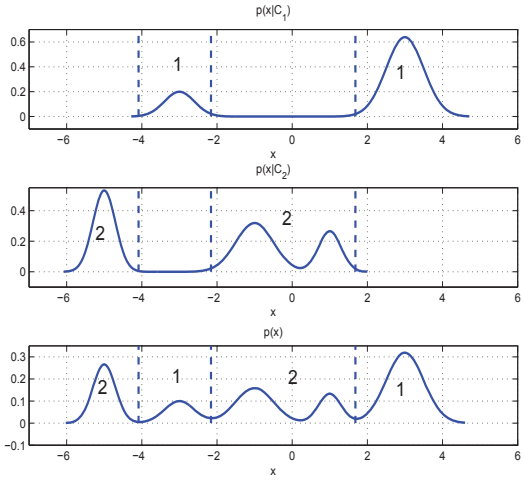
Example 1: As a simple example, let us assume a one-dimensional signal case. The input signal to the encoder is generated from two classes. Each class is defined with the following conditional PDFs

$$p(x|C_1) = \frac{1}{\sqrt{2\pi}} \left(\frac{0.8}{0.5} e^{-\frac{(x-3)^2}{2\times 0.5^2}} + \frac{0.2}{0.4} e^{-\frac{(x+3)^2}{2\times 0.4^2}} \right),$$

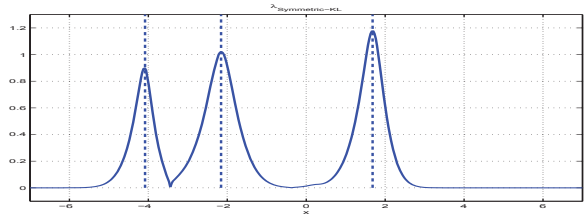
$$p(x|C_2) = \frac{1}{\sqrt{2\pi}} \left(\frac{0.2}{0.3} e^{-\frac{(x-1)^2}{2\times 0.3^2}} + \frac{0.4}{0.5} e^{-\frac{(x+1)^2}{2\times 0.5^2}} + \frac{0.4}{0.3} e^{-\frac{(x+5)^2}{2\times 0.3^2}} \right),$$

and therefore $p(x) = P(C_1)p(x|C_1) + P(C_2)p(x|C_2)$. The plots for these PDFs are presented in Figure 3.1(a) with the assumption $P(C_1) = P(C_2) = 0.5$. Here we assume a naive Bayes classifier and therefore the decision boundaries for classes are based on $P(C_1)p(x|C_1) \gtrless P(C_2)p(x|C_2)$. These decision boundaries are marked as dashed vertical lines in the figures. We will look at λ_{KL} and $\lambda_{\text{Tradeoff}}$. For comparison purposes, we also included the λ_{MSE} (i.e. $\mathbf{D}_{\text{MSE}}(\mathbf{x}) = 2\mathbf{I}$). All of these optimum point densities are plotted in Figure 3.1. It is clear that λ_{KL} allocates more points to the boundaries of each class (dashed vertical lines) and less points as it moves away from the boundaries of classes. On the other hand λ_{MSE} allocates points based on the distribution of the signal regardless of the boundaries for classification. Furthermore, $\lambda_{\text{Tradeoff}}$, depending on the value of α which determines the amount of tradeoff behaves

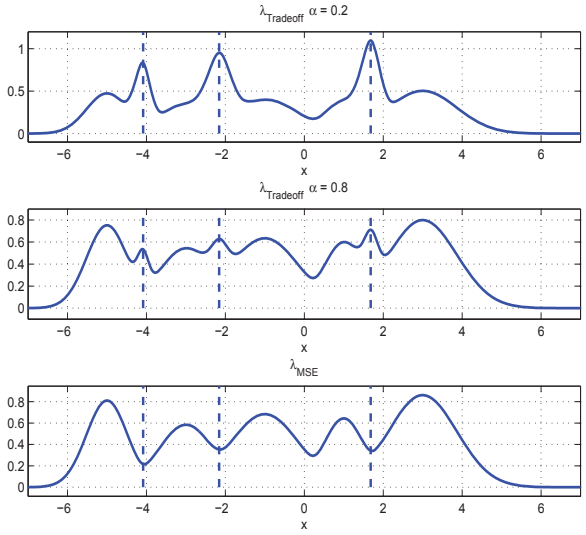
intuitively, i.e., for small α it behaves more like λ_{KL} and less like λ_{MSE}. In other words, it is easy to note that as we move the weight of the tradeoff from KL divergence to MSE, i.e., increasing α, the points of the point density function move from the boundaries to the regions with high probability of signal occurrence.



(a) Top and Middle: $P(x|C_1)$ and $P(x|C_2)$ Conditional PDFs of synthetic data extracted from Class 1 and 2. Bottom: Combined PDF of the source data.



(b) Point Density Function for the Symmetric KL



(c) Point Density Functions for Tradeoff Top: $\alpha = 0.2$, Middle: $\alpha = 0.8$ and Bottom: MSE case

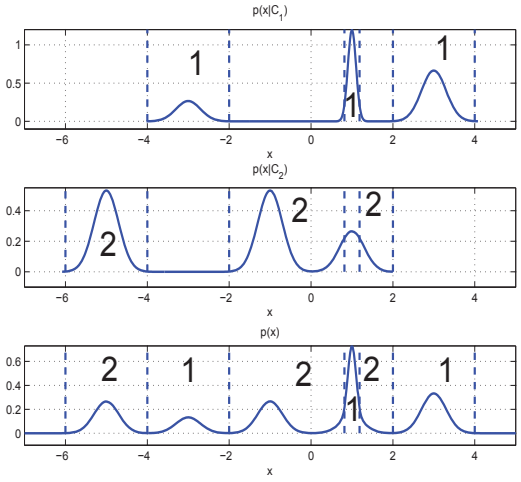
Figure 3.1: Example 1: Synthetic 1-D Signal, PDFs with more separation.

Example 2: In this example, we consider a similar one dimensional case where there exists an overlap of the peaks of the PDFs between the two classes, specifically each class is defined with the following conditional PDFs

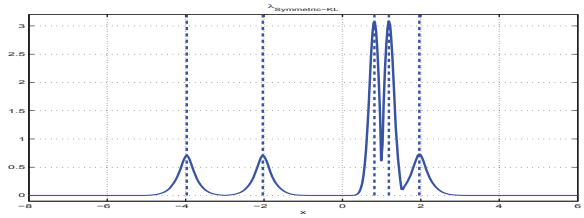
$$p(x|C_1) = \frac{1}{\sqrt{2\pi 0.3^2}}\left(0.5e^{-\frac{(x-3)^2}{2\times 0.3^2}} + 0.2e^{-\frac{(x+3)^2}{2\times 0.3^2}}\right) + 0.3\frac{1}{\sqrt{2\pi 0.1^2}}e^{-\frac{(x-1)^2}{2\times 0.1^2}},$$

$$p(x|C_2) = \frac{1}{\sqrt{2\pi 0.3^2}}\left(0.2e^{-\frac{(x-1)^2}{2\times 0.3^2}} + 0.4e^{-\frac{(x+1)^2}{2\times 0.3^2}} + 0.4e^{-\frac{(x+5)^2}{2\times 0.3^2}}\right),$$

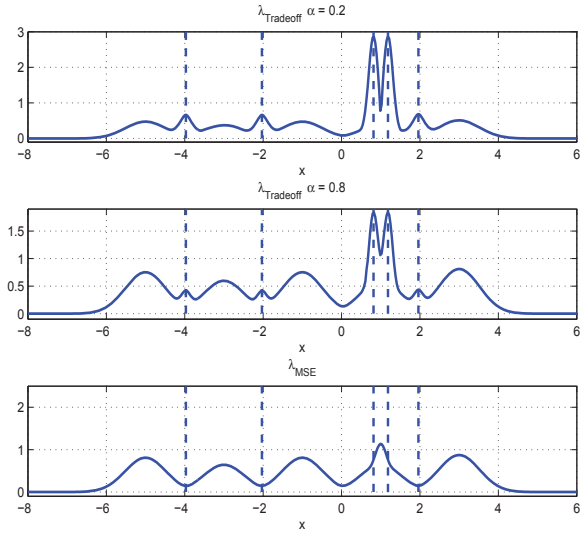
and therefore the overlap occurs at $x = 1$ as shown in Figure 3.2(a). The corresponding point density functions are depicted in Figure 3.2. Similar conclusions as those made regarding Example 1 apply in this example. It is notable that λ_{KL} puts emphasis on the region where the overlap of the peaks of the two classes occurs by forcing more codebook points in order to alleviate the adverse effects of this overlap on classification.



(a) Top and Middle: $P(x|C_1)$ and $P(x|C_2)$ Conditional PDFs of synthetic data extracted from Class 1 and 2. Bottom: Combined PDF of the source data.



(b) Point Density Function for the Symmetric KL



(c) Point Density Functions for Tradeoff Top: $\alpha = 0.2$, Middle: $\alpha = 0.8$ and Bottom: MSE case

Figure 3.2: Example 2: Synthetic 1-D Signal, PDFs with more overlap.

To verify that the newly derived point densities actually perform better in classification accuracy at a decoder we develop the following experiment. We generate synthetic data based on the source PDF, i.e., $p(x)$. Then we design random codebooks based on λ_{KL}, λ_{MSE}, and $\lambda_{\text{Tradeoff}}$ and compare the results for the different cases. Since we assume that the encoders do not perform any classification, the encoding is solely based on the MSE. Since in each experiment we have a random codebook we repeat the experiments 20 times and report the average results to generate a more meaningful and code independent comparison measure. The results for the signal model presented in Example 1 are presented in Table 3.1.

Table 3.1: Example 1: Classification Error(%) and Distortion(dB) results for the signal presented in Example 1 - Please refer to Figure 3.1

	λ_{KL}		$\lambda_{\text{Tradeoff}}$ $\alpha = 0.2$	
Bits	%	dB	%	dB
10	0.0027	-8.7481	0.0053	-41.6524
8	0.0246	-6.8633	0.0662	-29.9626
6	0.0914	-4.1752	0.3603	-18.1660
	$\lambda_{\text{Tradeoff}}$ $\alpha = 0.8$		λ_{MSE}	
Bits	%	dB	%	dB
10	0.0174	-43.3311	0.0482	-43.6344
8	0.1068	-31.7181	0.1476	-31.6691
6	0.4015	-19.6222	0.4507	-20.3667

It is easy to see that the deployment of codebooks based on the symmetric KL divergence introduces a considerable improvement in the classification results. This is

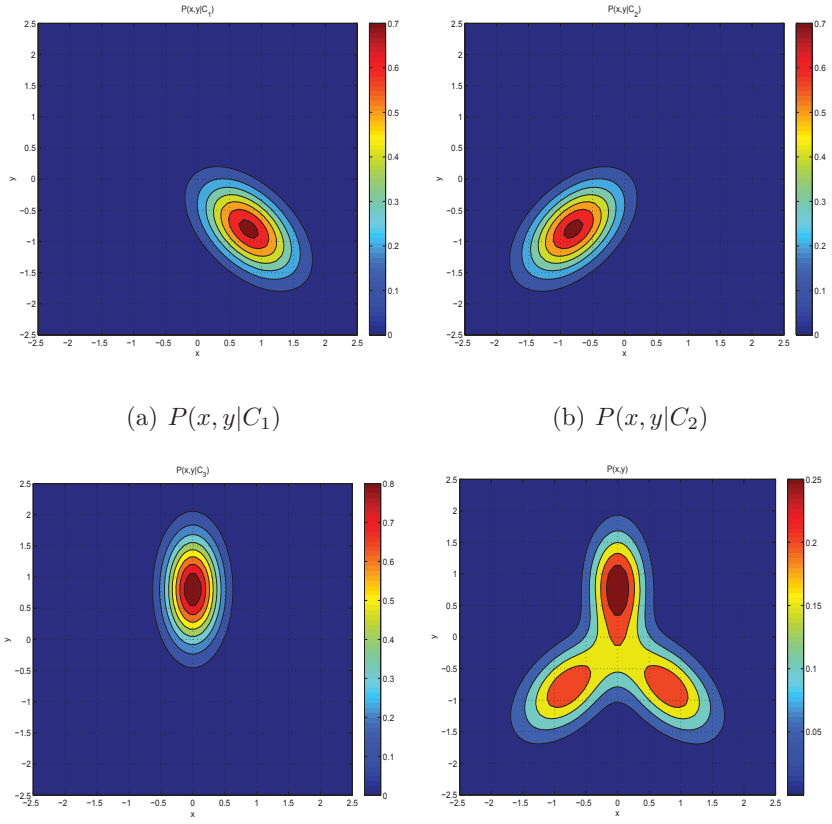
at the expense of introducing more distortion. On the other hand, it is very notable that by introducing the slightest amount of tradeoff, e.g., $\alpha = 0.2$, the distortion result improves considerably while keeping classification results better than those provided by codebooks based on λ_{MSE}. Furthermore, the results for the signal model presented in Example 2 are presented in Table 3.2 with similar conclusions regarding the results. Please note that we used a different number of bits in this example. Because of the complex nature of the signal model, more bits are required to design a codebook that represents the optimal point density function.

Table 3.2: Example 2: Classification Error(%) and Distortion(dB) results for the signal presented in Example 2 - Please refer to Figure 3.2

	λ_{KL}		$\lambda_{\text{Tradeoff}}\ \alpha = 0.2$	
Bits	%	dB	%	dB
12	0.0061	-14.5497	0.0120	-54.1325
10	0.0281	-10.5105	0.0329	-41.9583
8	0.0908	-7.0685	0.1511	-29.8545
	$\lambda_{\text{Tradeoff}}\ \alpha = 0.8$		λ_{MSE}	
Bits	%	dB	%	dB
12	0.0126	-56.8055	0.0295	-57.5876
10	0.0553	-44.7926	0.1302	-45.4882
8	0.4179	-32.5680	0.5571	-33.8399

3.5.2 Two Dimensional Signals

In this section, we look at a two dimensional signal as an example. The signal is drawn from three 2-D gaussian distributions. Each 2-D gaussian represents one possible class of the signal. This is shown in Figure 3.3 .



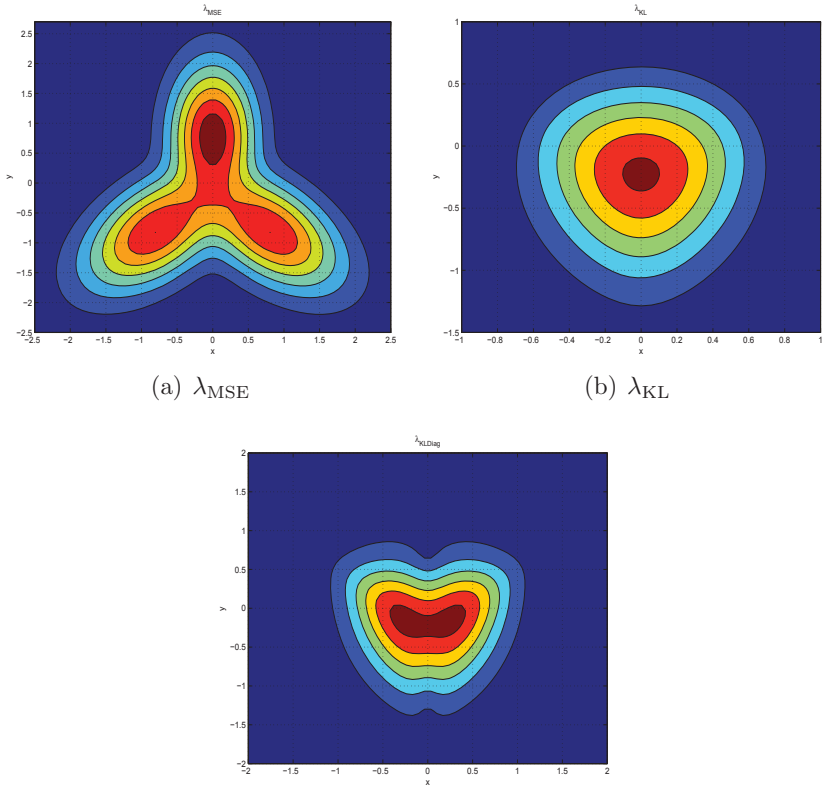
(a) $P(x, y|C_1)$ (b) $P(x, y|C_2)$

(c) $P(x, y|C_3)$ (d) $P(x, y)$

Figure 3.3: (a)-(c) Conditional PDF of 2-D synthetic data extracted from 3 Classes (d) Combined PDF of the source data.

The corresponding point densities for MSE distortion (λ_{MSE}), symmetric KL divergence distortion (λ_{KL} developed in Section 3.2), and the diagonal $\mathbf{D}$ approximation of the symmetric KL divergence distortion (λ_{KLDiag} developed in Section 3.4) are pre-

sented in Figure 3.5. As can be seen from the figure, λ_{MSE} puts emphasis on where the signal is more probable. On the other hand λ_{KL} only puts emphasis on the intersection of the class boundaries. Similar to the scalar example, we generated codebooks



(a) λ_{MSE} (b) λ_{KL}

(c) λ_{KLDiag}

Figure 3.4: Point Density Functions for the MSE, KL, and Diagonal KL for 2-D signal presented in Figure 3.3.

based on different point densities and evaluated the performance of each quantizer. The classification results are presented in Table 3.3. It is clear that λ_{KL} performs better than λ_{MSE} in the classification task. On the other hand the distortion is inferior for λ_{KL} as would be expected. Furthermore, the performance of λ_{KLDiag} is very similar to the performance of λ_{KL}.

Table 3.3: 2-D Example: Classification Error(%) and Distortion(dB) results for the signal presented in Figure 3.3

	λ_{KL}		λ_{KLDiag}	
Bits	%	dB	%	dB
15	0.0012	-21.5656	0.0014	-21.0803
14	0.0017	-19.4565	0.0018	-19.1360
12	0.0033	-15.3555	0.0036	-15.1665
	$\lambda_{\text{KL-Sum-Square}}$		λ_{MSE}	
Bits	%	dB	%	dB
15	0.0015	-20.3405	0.0024	-29.5487
14	0.0020	-18.8674	0.0032	-27.2438
12	0.0037	-15.4987	0.0059	-24.4403

3.5.3 Real Data Set

In this section, we compare the performance of the developed quantization algorithm with the MSE quantization in the task of classification at the decoder. The data set that we use is the iris data set from UCI Machine Learning Repository. The iris data set consists of 4 attributes of three classes of the iris plant. There are 50 instances of each class. The attributes are petal and sepal length and width.

Simulations for 2 attributes of the iris plant To demonstrate the performance of the classification at the decoder, for presentation purposes we first proceed with two out of four attributes. This choice will make the explanation of our method easier to represent with plots and graphs.We divide 50 instances of each class into a partition of 30 instances for training and 20 instances for testing. We do this partitioning randomly and repeat the experiment 10 times. We fit a two dimensional Gaussian to each class. Figure 3.5(a) shows one of these partitionings and the Gaussians that fit three classes. Therefore knowing the probability density function of the signal, and the conditional probability density function of the signals given each class, we calculate

the point density function of the optimal quantizer from Equation 3.13. Furthermore, we generate random codebooks based on the point density function. One example of such random codebooks is presented in Figure 3.5(b). As we observed in the synthetic data examples, the MSE quantizer puts the codebook points at where the signal is and the KL quantizer puts the codebook points at the boundary of classes. We repeat generating random codebooks 100 times in each trial, and average the classification error and the distortion over the 100 instances.

The classification error and distortion results are presented in Table 3.4. As was expected the KL quantizer performs slightly better than MSE quantizer in the task of classification at the decoder.

Simulations for 4 attributes of the iris plant Now we present the classification results for all the four attributes of the iris database. To apply the quantization scheme that we designed in Section 3.2, we need to generate a random codebook based on the analytical expression for point density function (see equation 3.13). This task entails analytically calculating all the components for an $n \times n$ matrix $\mathbf{D}(\mathbf{x})$. Closely looking at the analytical expressions for components of $\mathbf{D}(\mathbf{x})$ in equations 3.19-3.20, we observe that this calculation involves calculating first and second derivatives of a probability density function $p(\mathbf{x})$ and conditional probability density functions $p(\mathbf{x}|C_i)$. For example, for the iris simulations, these probability density functions are the probabilistic model representations of iris dataset attributes and are derived from a 4-dimensional Gaussian Mixture Model fit to the the iris dataset. We were able to follow these aforementioned steps with the help of the Symbolic Math Toolbox of MATLAB for 2-dimensional signals (2 attributes of the iris plant)

but we were not successful in doing these calculations for 4-dimensional signals (all 4 attributes of the iris plant). Therefore, to proceed with our simulation for all four attributes of the iris signal we apply numerical methods for calculating the derivatives of functions. We applied the simple finite difference approximation which is a simple two point approximation for derivative as in

$$u'(x_i) \approx \frac{u(x_i + \Delta x) - u(x_i)}{\Delta x}. \tag{3.34}$$

The classification error and distortion results are presented in Table 3.5. The results are not far from expectations. The higher the number of bits, the better the classification result and the distortion for all three classification schemes. Again, the KL quantizer performs better than the diagonal KL which itself performs better than the MSE quantizer. Furthermore, as we expected the distortion performance is better for the MSE quantizer. By introducing a trade-off factor in our KL quantizer we can improve the distortion while maintaining a good classification.

Table 3.4: Iris Data Set: Classification Error(%) and Distortion(dB) - Two attributes

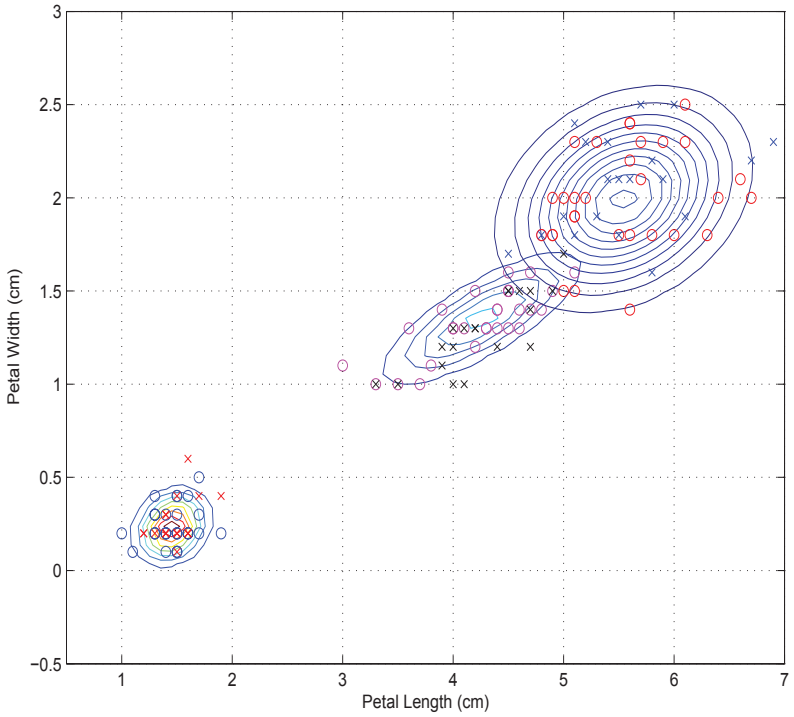
	λ_{KL}		λ_{KLDiag}		λ_{MSE}	
Bits	%	dB	%	dB	%	dB
7	2.2300	-13.2085	3.1950	-16.2386	2.3033	-19.8871
8	1.8583	-15.2203	2.4417	-19.2625	1.9600	-22.8047
9	1.6950	-17.1609	1.9333	-22.2294	1.7417	-25.7076

Table 3.5: Iris Data Set: Classification Error(%) and Distortion(dB) - Four attributes

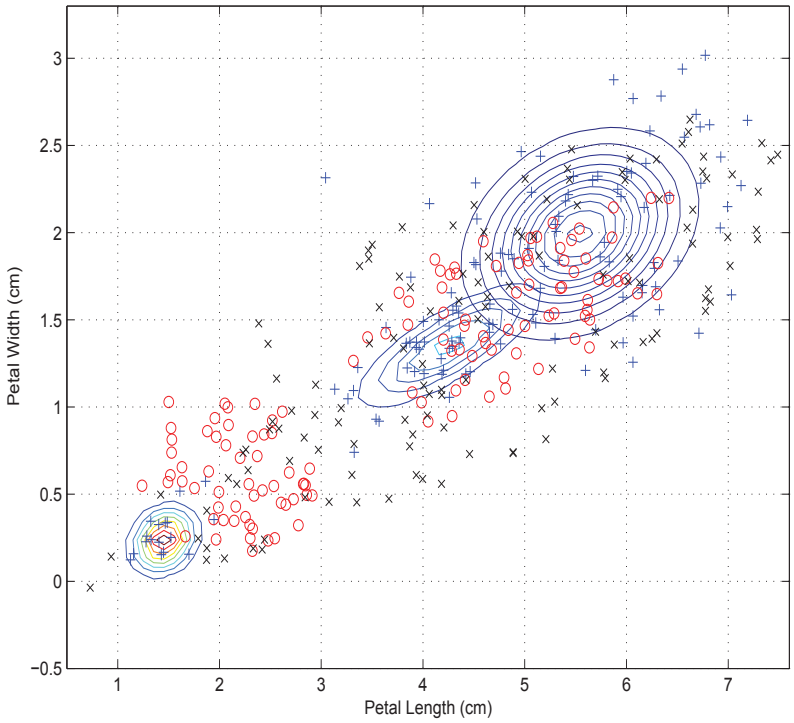
	λ_{KL}		λ_{KLDiag}		λ_{MSE}	
Bits	%	dB	%	dB	%	dB
7	8.1023	0.4721	8.1649	0.4968	12.7641	-6.3498
8	5.7673	-1.1981	5.9978	-1.0239	9.4306	-7.9476
9	5.000	-2.6135	4.9925	-2.6071	7.9290	-9.5392

3.6 Remarks

We presented a solution for quantization of signals for the purpose of obtaining a more accurate classification at the decoder. We employed high-rate theory for quantizer design. Therefore an optimal point density function that determines where the codepoints should lie in the space was derived. We chose the symmetric KL divergence between the conditional probabilities of classes given data before and after quantization as our distortion measure. Furthermore, we studied the tradeoff between classification accuracy and reproduction fidelity and presented a point density function for the case of tradeoff. We analyzed the effects of a mismatched distortion measure and showed that for reduced complexity the original distortion measure can be replaced by a weighted mean square error (WMSE) distortion measure. We examined the performance of these methods on synthetically generated data and real data set and observed that they were superior in the task of classification of signals at the decoder with lower reproduction fidelity as tradeoff.



(a) Three Classes of Iris Plant, training samples are shown in circles and testing samples in crosses



(b) Random Codebooks: MSE (blue pluses), KL (red circles), and KLDiag (black crosses)

Figure 3.5: Iris Data

CHAPTER 4

Cross-Layer Effects of Network Utility Maximization and Quantization

In Chapter 2 of this dissertation, we studied the problem of allocating network resources such as bandwidth and power in a multi-hop wireless network, primarily for the purpose of achieving a good throughput and low average queuing delay. The solution to this problem called for the sources and links to update their transmission rate or power based on either locally measurable messages or feedback sent from other sources and links through a flooding protocol. The solution entailed power control in the physical layer and congestion control in the transport layer. See Figure 4.1.

In Chapter 3 of this dissertation, we designed a quantization method that is optimized for the task of accurate classification of the reconstructed signal at the decoder. The solution entailed moving away from the traditional mean square error measure for distortion and to introduce the symmetric Kullback-Leibler divergence measure. The quantizer is in the presentation layer of a communication network (Figure 4.1).

In this chapter, we would like to demonstrate that solutions to these two important problems can be combined to devise a communication system that not only optimizes

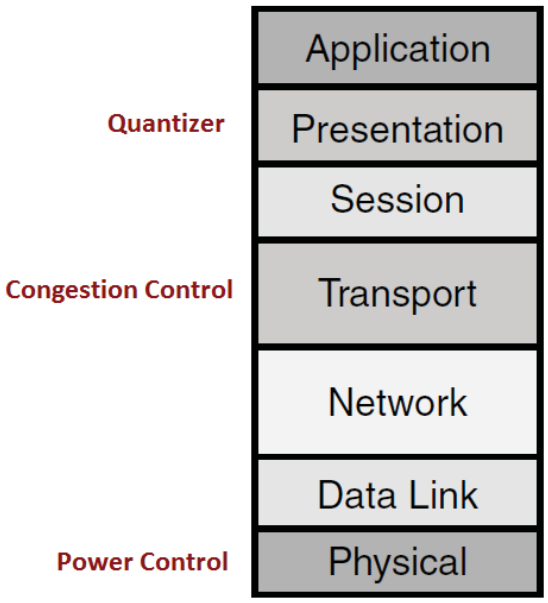


Figure 4.1: Network Layers and the Optimization Algorithms.

the rate of transmission and power of transmitters but also deploys a quantizer that is optimized in the task of classification at the decoder.

Consider scenarios where signals gathered from sensors are being transmitted in a multi-hop wireless network. These signals require timely delivery of their content to the receivers. The capacity of the links puts a cap on the maximum allowed transmission rate, therefore the signals need to be quantized efficiently. Furthermore, at the receivers we desire not only the reconstruction of the signals but also the preservation of the properties of the signal which ensure accurate classification. In this scenario, the quantizer in the presentation layer should be designed based on the optimization criteria developed in Chapter 3. The message passing and convergence of the sub-gradient method follow the theory developed in Chapter 2 which involved power control in the physical layer and congestion control in the transport later.

Furthermore, the achievable data rates on wireless network links will constrain the bits/vector rate that is used in the quantization. Therefore, feedback from the network that would determine the data rate of a source would also affect the bits/vector rate for quantization purposes.

In the following, we demonstrate how we combine the algorithms developed in the previous chapters. We use some of the examples from the previous chapters for the choice of network topology and the same parameters for the NUM optimization problem. Furthermore we use the same signals (four attributes of iris plant) as the signal of interest to transmit through the network. We build on top of the simulations we presented in Chapter 2 and Chapter 3 and present simulation results for combination of the algorithms developed in the previous chapters. We demonstrate that by taking advantage of the combination of the algorithms we are able to achieve the desired average queuing delay requirements, maximum power available, and better performance of the classification results at the decoder.

4.1 Combined Network Utility Maximization and Quantization

Let us assume a network optimization convergence behavior similar to the convergence behavior we observed in Figure 2.3 of Chapter 2. We repeat this figure here for the ease of reading, see Figure 4.2. This figure shows the convergence of data rates of 4 sessions, marked 1 through 4, in the network shown in Figure 2.1 repeated here in Figure 4.3. This pattern of change in the data rates of four sessions is converted into bits/vector rates of Figure 4.4. The network algorithm enforces the sources to send data with variable bits/vector depending on the data rate that was achieved from the

solution of the network algorithm. The effect of delay requirements on the bits/vector for two of the sessions (Sessions 1 and 4) is also shown with dashed lines in Figure 4.4. For the other two sessions, the effect of delay requirements on the bits/vector rates is negligible.

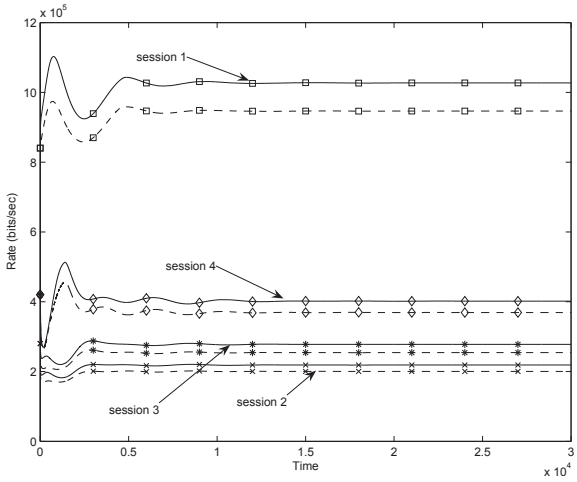


Figure 4.2: Convergence for the rates of 4 sessions. Solid curves are for the case of no delay constraint and dashed curves are for the case with delay constraints.

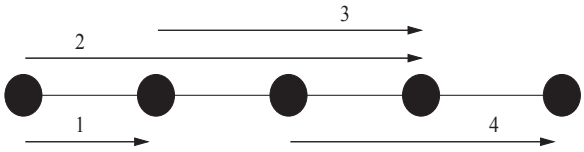


Figure 4.3: Network and flow topology I.

Let us assume that the data being sent is the four attributes of the iris plant presented in Section 3.6.3. In the aforementioned section we observed the classification error and distortion results presented in Table 3.5 for three choices of bits/vector. We expand this table to many more choices of bits/vector for the purpose of simulations in this section. The results are presented in Table 4.1. We use these results in the following simulations.

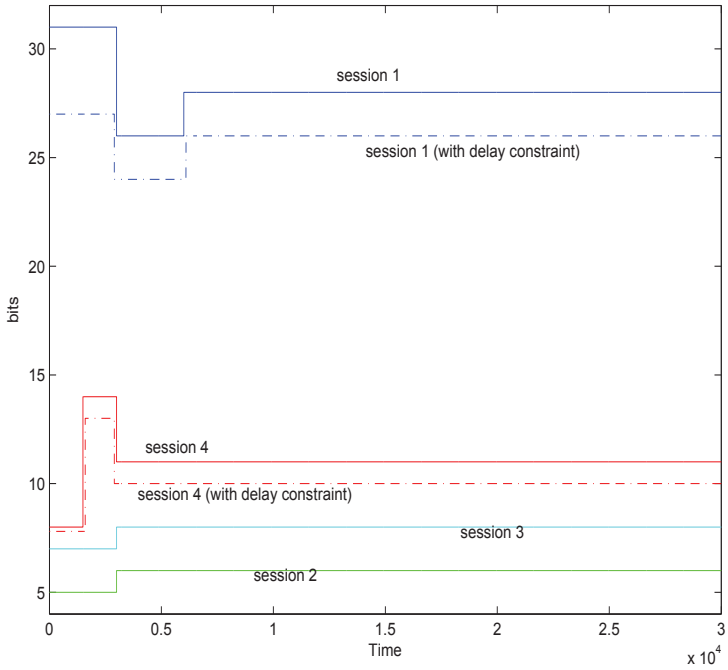


Figure 4.4: Bit Rate Constraint based on Network Optimization Convergence (4 sessions)

4.1.1 Average Behavior Of Network Optimization Convergence

The different values of bits/vector rate at each time interval of Figure 4.4 will result in a different classification result and also different distortion of the signals at each time interval until the algorithm converges. Once the algorithm converges the results of classification and distortion will follow the corresponding results for the specific bits/vector presented in Table 4.1. But before the convergence (time 0.6×10^4 in Figure 4.4), we would like to look at the average result of classification and distortion measure. If we average the classification and distortion performance and by taking the results from Table 4.1 into account, we can derive the results in

Table 4.1: Iris Data Set (Four attributes): Classification Error(%) and Distortion(dB) results for multiple choices of bits/vector.

	λ_{KL}		λ_{KLDiag}		λ_{MSE}	
Bits/vector	%	dB	%	dB	%	dB
4	23.8526	7.4631	23.8416	7.1900	36.1042	0.7404
5	14.4563	4.5628	15.3820	4.5431	22.8974	-2.0927
6	10.1825	2.3444	11.0941	2.3931	16.1980	-4.3232
7	8.1023	0.4721	8.1649	0.4968	12.7641	-6.3498
8	5.7673	-1.1981	5.9978	-1.0239	9.4306	-7.9476
9	5.000	-2.6135	4.9925	-2.6071	7.9290	-9.5392
10	4.0321	-4.0253	3.8744	-4.0123	6.6716	-10.4657
11	3.1715	-5.1890	3.1731	-5.1380	5.2822	-11.9055
12	2.8445	-5.9271	2.8656	-6.1425	4.5944	-12.7056
13	2.2905	-6.8768	2.3135	-6.8714	3.9699	-14.5315
14	1.9860	-7.7229	2.0376	-7.7784	3.3384	-15.6657
19	1.1599	-11.2211	1.1366	-11.2765	1.9091	-18.6926
21	0.9263	-12.9310	0.9037	-13.5464	1.4933	-20.6199
22	0.8746	-13.8142	0.8597	-13.8624	1.4840	-20.4051
23	0.8100	-14.4640	0.7756	-14.2114	1.2661	-21.4629
24	0.7339	-14.7171	0.6896	-14.3811	1.1771	-22.6660
26	0.6384	-15.0421	0.6277	-15.1192	1.0658	-23.1751
27	0.5642	-16.3304	0.5442	-16.3040	0.9621	-23.0438
28	0.5529	-15.9460	0.5169	-17.3628	0.9230	-23.4110
31	0.4645	-17.1634	0.4207	-17.2230	0.7329	-25.0613

Table 4.2 which shows the average distortion and classification error results (during convergence of data rates) for the four sessions with different quantization schemes. It is apparent from the results in Table 4.2 that both λ_{KL} and λ_{KLDiag} quantization schemes perform better in the task of classification.

Table 4.2: Average Classification Error(%) and Distortion(dB) results for transmitting Four Attributes of the Iris Data Set through a network with data rate (bits/ convergence behaviour shown in Figure 4.4:

	λ_{KL}		$\lambda_{\mathrm{KLDiag}}$		λ_{MSE}	
Session	%	dB	%	dB	%	dB
1	0.5526	-15.9774	0.5183	-17.1245	0.9183	-23.5524
1 (with delay constraint)	0.6406	-15.1384	0.6255	-15.1639	1.0666	-23.1110
2	10.6099	2.5663	11.5228	2.6081	16.868	-4.1001
3	6.0008	-1.0311	6.2145	-0.8718	9.7639	-7.7878
4	3.2420	-5.1162	3.2576	-5.0643	5.3924	-11.8956
4 (with delay constraint)	4.0318	-4.0265	3.9025	-4.0058	6.6745	-10.5431

4.1.2 Short-Term Behavior Of Network Optimization Convergence

Now let us look at the short term effects of the convergence of the network algorithm on the classification and distortion results. In order to observe this behavior we choose a moving average filter of length 1500. These results are presented in Figure 4.5 for classification error and in Figure 4.6 for distortion. The pre-convergence trend of classification error and distortion is what is expected based on the variations in the bits/vector rate, i.e., short-term upward spikes in the bits/vector will result in better classification and distortion temporarily. The reverse argument also holds, i.e., short-term dips of bits/vector adversely affects short-term classification and distortion.

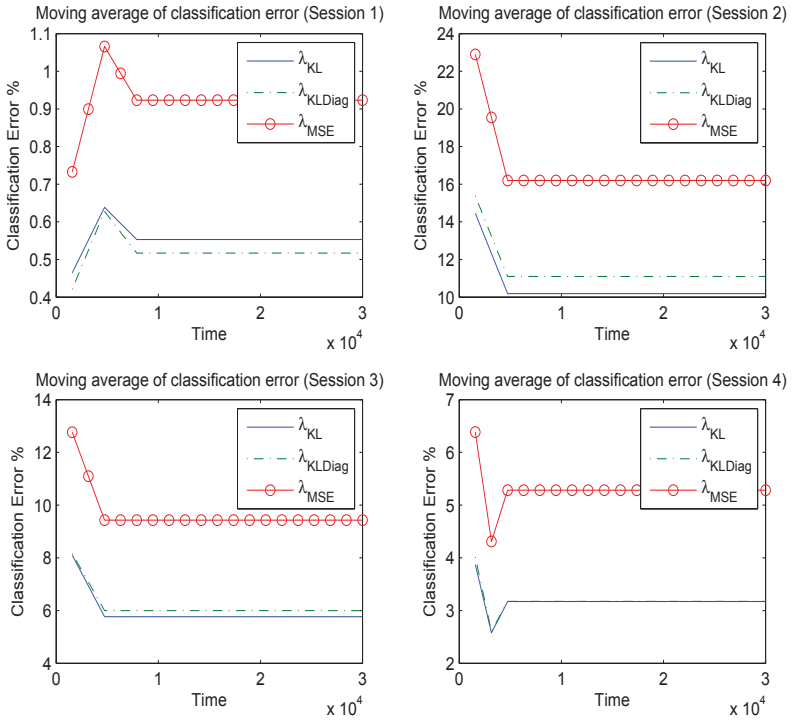


Figure 4.5: Moving Average of Classification Error (4 sessions)

4.1.3 Changing the Delay Requirement

Let us now study the effect of imposing delay requirements on two of the sessions in the previous example, specifically Sessions 1 and 4. Previously in Figure 4.4, we saw the effect of the delay requirements on bits/vector rate of Sessions 1 and 4. Now in the Figures 4.7-4.10 we see the effect of these bits/vector rates on the classification and distortion measures. Imposing delay requirements will result in lower bits/vector rates as can be seen in Figure 4.4. The lower bits/vector rate will result in lower classification and distortion performance as is depicted in Figures 4.7-4.10.

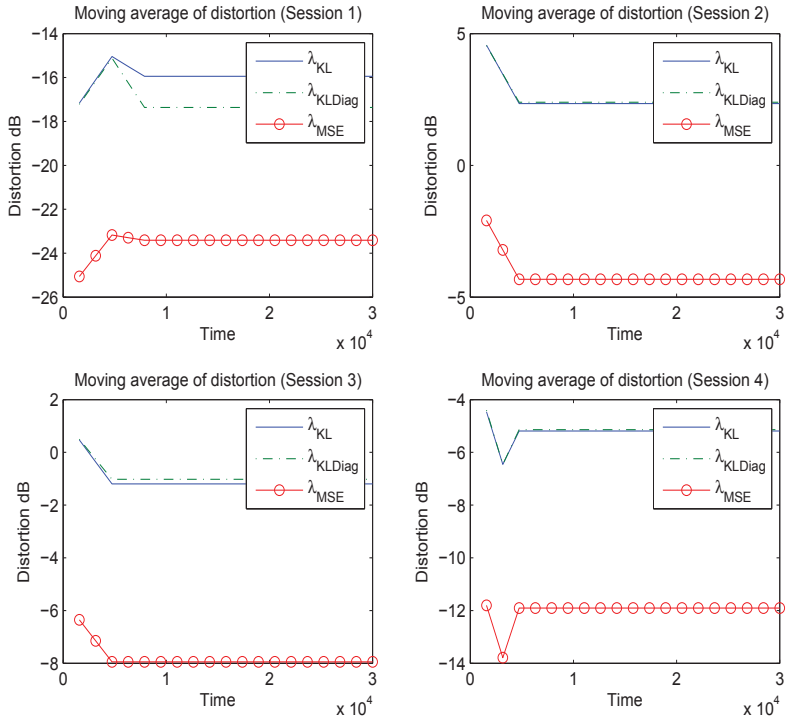


Figure 4.6: Moving Average of distortion (4 sessions)

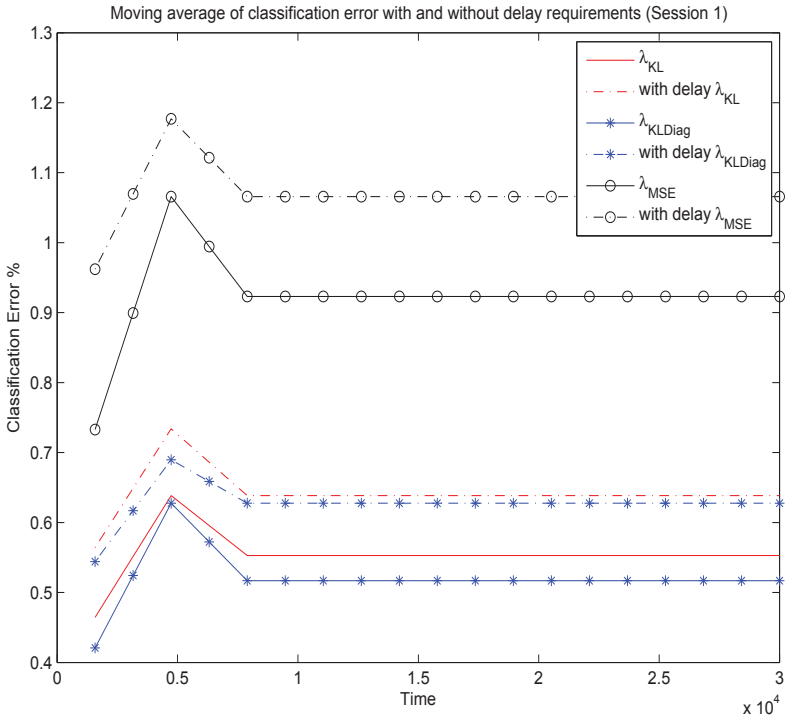


Figure 4.7: Moving Average of Classification Error with and without delay requirements(Session 1)

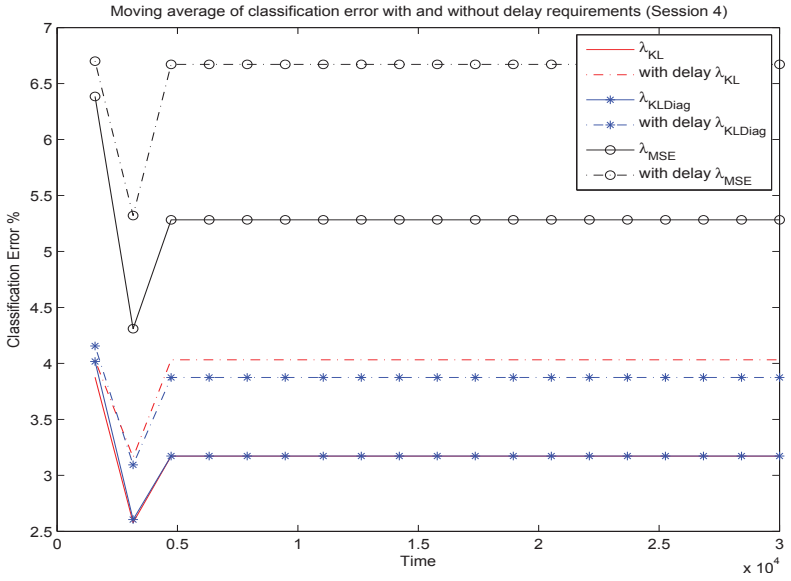


Figure 4.8: Moving Average of Classification Error with and without delay requirements(Session 4)

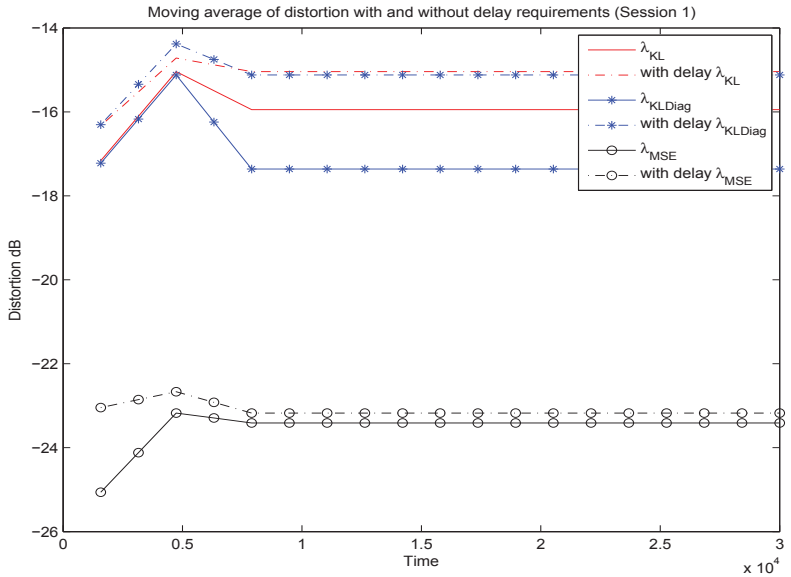


Figure 4.9: Moving Average of Distortion with and without delay requirements (Session 1)

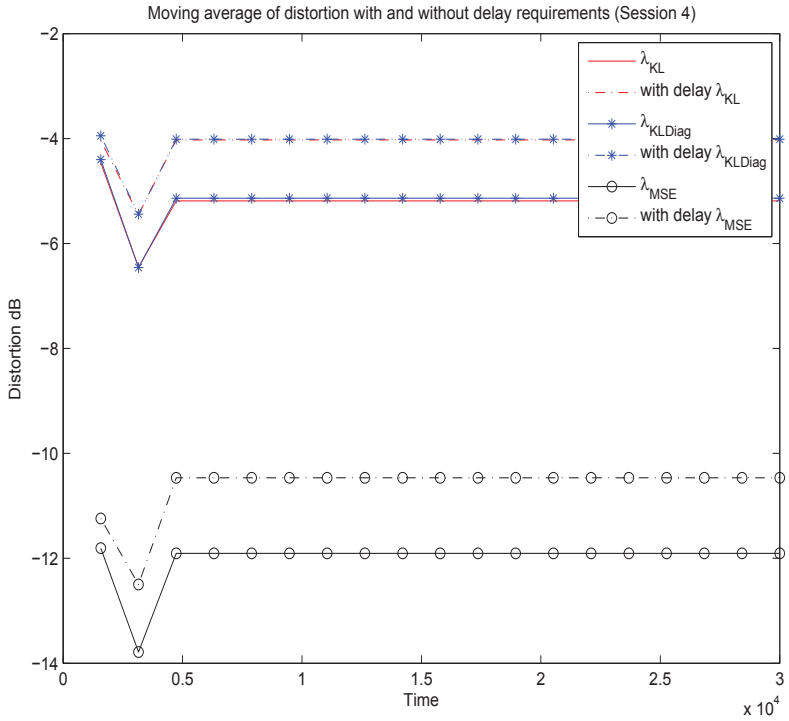


Figure 4.10: Moving Average of Distortion with and without delay requirements (Session 4)

In the following we study some of the network topology, and parameter settings that we studied in Section 2.4. As a reminder from Chapter 2, in the simulations, the utility function for all sessions s is chosen as $U_s(x_s) = \alpha_s \log x_s$, which is an increasing, strictly concave function of the session rate x_s. It is also the utility function of TCP Vegas. In congestion control with TCP Vegas, the utility function's parameter α_s is typically set to the same constant for all sessions in order to achieve proportional fairness of network rates. However, proportional fairness no longer remains true in the NUM problem (2.13) formulation because of the inclusion of the sessions' average queueing delay as constraints. Therefore, in the simulations, we compare cases of all

sessions possessing the same value of α_s with cases in which some sessions change their value of α_s in order to reflect their preference for high data rate. In particular, we examine cases of greedy sources desiring both high rate and low delay, sources possessing a preference for low delay, and sources possessing a preference for high rate. In these comparisons, we consider two different simple wireless network configurations with 5 nodes and 4 sessions. See Figs. 4.11 and 4.12.

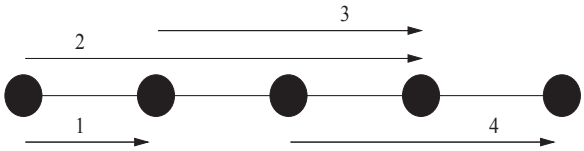


Figure 4.11: Network and flow topology I.

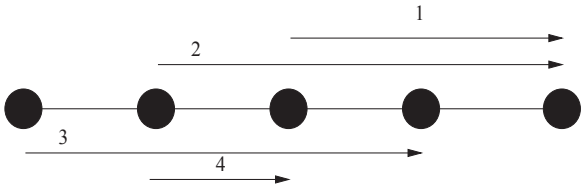


Figure 4.12: Network and flow topology II.

Specifically, in Set A of the simulations, we study the case of sessions with no delay requirements and another case with different delay requirements. The results are then compared and the impact of the delay requirements is studied. In Set B of the simulations, we simulate how different α_s values of different sessions affect the rate-power allocation.

In Set C of the simulations, the relationship between the rate-power allocations and flow topology is studied. In the captions of the tables, $\boldsymbol{\alpha_s}$ represents the vector of α_s pertaining to different sessions, i.e., $\boldsymbol{\alpha_s} = [\alpha_1, \alpha_2, \alpha_3, \alpha_4]$.

We present a description of simulations in different scenarios in Table 4.3 . We

suggest that the reader refer to this table in order to follow the detailed description of the scenarios in the following sections with ease.

Table 4.3: Summary of the simulation scenarios.

Set A	Scenarios 1-4	Session 2 tightens its delay requirements from 10 ms down to 2 ms while other sessions maintain their delay requirements at 10 ms
Set B	Scenarios 5-8	Session 2 tighten its delay requirements just like Scenarios 1-4 but also requires higher data rate by setting $\alpha_s = 2$ versus all other sessions having $\alpha_s = 1$
Set A (to compare with Scenario 9)	Scenario 5	All sessions have 10 ms delay requirements. Session 2 requires higher data rate by setting $\alpha_s = 2$ versus all other sessions having $\alpha_s = 1$
Set B (to compare with Scenario 5)	Scenario 9	Compared to Scenario 5, Sessions 3 and 4 require lower delay. Therefore, they set their delay requirement at 5 ms.
Set B	Scenario 10	Sessions 3 and 4 require heigher rate by setting $\alpha_s = 2$. Session 2 requires lower delay of 5 ms versus 10 ms for all other sessions.
Set C	Scenario 11-14	Similar to Scenarios 1-4 with a different network topology.

4.1.4 Set A: Changing the Delay Requirement

We start by Scenarios 1-4, presented in Tables 4.4-4.5, where we study how different delay requirements affect the data rate and power settings of the sources and links. We observe that as Session 2 tightens its delay requirements (10ms $\rightarrow$ 8ms $\rightarrow$ 5ms $\rightarrow$ 2ms) in Scenarios 1-4, the transmission power on link 2 increases. Furthermore the data rate and therefore bits/vector rate decreases for Session 2. A decrease in bits/vector rate on Session 2 will result in lower classification accuracy regardless

of the quantization method. However, the quantization methods developed in Chapter 3 will fare better as far as classification results are concerned. Notably here, an application that requires low delay usually can tolerate higher classification error, for example a voice over IP application. On the other hand, all the other sessions were forced to reduce their data rates and therefore bits/vector rates (in order to reduce the average queuing delay, refer to Equation 2.5) which results in higher classification errors.

Table 4.4: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 1, 1, 1]$

	Session #			
	1	2	3	4
Scenario 1				
Delay Requirement (ms)	10.0	10.0	10.0	10.0
Delay Achieved (ms)	2.7	10.0	7.3	10.0
Data Rate (Kb/s)	946.5	200.4	254.3	369.2
bits/vector	26	6	7	10
λ_{KL} %	0.6384	10.1825	8.1023	4.0321
λ_{KL} dB	-15.0421	2.3444	0.4721	-4.0253
λ_{KLDiag} %	0.6277	11.0941	8.1649	3.8744
λ_{KLDiag} dB	-15.1192	2.3931	0.4968	-4.0123
λ_{MSE} %	1.0658	16.1980	12.7641	6.6716
λ_{MSE} dB	-23.1751	-4.3232	-6.3498	-10.4657
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.111	0.120	0.060
Scenario 2				
Delay Requirement (ms)	10.0	8.0	10.0	10.0
Delay Achieved (ms)	2.1	8.0	5.9	10.0
Data Rate (Kb/s)	921.9	195.7	248.4	362.5
bits/vector	26	5	7	10
λ_{KL} %	0.6384	14.4563	8.1023	4.0321
λ_{KL} dB	-15.0421	4.5628	0.4721	-4.0253
λ_{KLDiag} %	0.6277	15.3820	8.1649	3.8744
λ_{KLDiag} dB	-15.1192	4.5431	0.4968	-4.0123
λ_{MSE} %	1.0658	22.8974	12.7641	6.6716
λ_{MSE} dB	-23.1751	-2.0927	-6.3498	-10.4657
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.112	0.120	0.059

Table 4.5: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 1, 1, 1]$

	Session #			
	1	2	3	4
Scenario 3				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	1.3	5.0	3.7	10.0
Data Rate (Kb/s)	843.5	179.9	228.7	338.8
bits/vector	23	5	6	9
λ_{KL} %	0.8100	14.4563	10.1825	5.0000
λ_{KL} dB	-14.4640	4.5628	2.3444	-2.6135
λ_{KLDiag} %	0.7756	15.3820	11.0941	4.9925
λ_{KLDiag} dB	-14.2114	4.5431	2.3931	-2.6071
λ_{MSE} %	1.2661	22.8974	16.1980	7.9290
λ_{MSE} dB	-21.4629	-2.0927	-4.3232	-9.5392
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.115	0.120	0.058
Scenario 4				
Delay Requirement (ms)	10.0	2.0	10.0	10.0
Delay Achieved (ms)	0.5	2.0	1.5	10.0
Data Rate (Kb/s)	413.4	83.9	105.3	199.5
bits/vector	11	2	3	6
λ_{KL} %	3.1715	.	.	10.1825
λ_{KL} dB	-5.1890	.	.	2.3444
λ_{KLDiag} %	3.1731	.	.	11.0941
λ_{KLDiag} dB	-5.1380	.	.	2.3931
λ_{MSE} %	5.2822	.	.	16.1980
λ_{MSE} dB	-11.9055	.	.	-4.3232
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.116	0.120	0.120	0.051

4.1.5 Set B: Changing α_s Together with the Delay Requirement

Furthermore, we consider a situation in which Session 2 emphasizes a preference for higher rate than the rest of the sessions. This preference is expressed by choosing α_s higher than other sessions. As Session 2 tightens its delay requirement in the Scenarios 5-8 depicted in Tables 4.6-4.7, it turns itself into a greedy source which requires both high rate and low delay. Comparing the bits/vector rate and classification results of Scenarios 5-8 with the bits/vector rate and classification results from Scenarios 1-4, we see that Session 2 achieves a higher bits/vector rate at the expense of lower bits/vector rates for other sessions. This in turn will result in better classification results for Session 2 and worse classification results for all other sessions. Note that in congestion control with TCP Vegas, the utility functions parameter α_s is typically set to the same constant for all sessions in order to achieve proportional fairness of network rates. However, proportional fairness no longer remains true in the NUM problem 2.13 formulation because of the inclusion of the sessions average queueing delay as constraints.

In Scenario 9, we consider a case of two sessions Sessions 3-4 requiring lower delay than they require in Scenario 5. But since they set their α_s values to 1, they are not greedy as far as expecting high rate as well. Comparing the results in Scenario 9 with those in Scenario 5, we see that Sessions 3-4, that require lower delay, achieve it at the expense of a lower bits/vector rate, and therefore an increase in their classification error rate. In Scenario 10 in Table 4.8, we consider the case where two sessions Sessions 3-4 require a higher rate but not very low delay and another session Session 2 requires lower delay but not a higher rate. Therefore Sessions 3-4 set their α_s

value at 2 while the rest of the sessions keep it at 1. Comparing the results of Scenario 10 with those of Scenario 3, we infer that the higher rate requirements of Sessions 3-4 are achieved at the expense of lower bits/vector rate rates for other sessions. Therefore, the lower classification performance for other sessions and better classification performance for Sessions 3-4 are achieved.

For some low data rates, the achieved bits/vector rate presents a very high quantization distortion and non-acceptable classification results which were not presented in the tables.

Table 4.6: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 2, 1, 1]$

	Session #			
	1	2	3	4
Scenario 5				
Delay Requirement (ms)	10.0	10.0	10.0	10.0
Delay Achieved (ms)	2.6	10.0	7.4	10.0
Data Rate (Kb/s)	786.0	317.6	199.0	295.1
bits/vector	22	9	6	8
λ_{KL} %	0.8746	5.000	10.1825	5.7673
λ_{KL} dB	-13.8142	-2.6135	2.3444	-1.1981
λ_{KLDiag} %	0.8597	4.9925	11.0941	5.9978
λ_{KLDiag} dB	-13.8624	-2.6071	2.3931	-1.0239
λ_{MSE} %	1.4840	7.9290	16.1980	9.4306
λ_{MSE} dB	-20.4051	-9.5392	-4.3232	-7.9476
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.118	0.120	0.056
Scenario 6				
Delay Requirement (ms)	10.0	8.0	10.0	10.0
Delay Achieved (ms)	2.1	8.0	5.9	10.0
Data Rate (Kb/s)	764.7	310.2	194.6	289.9
bits/vector	21	9	5	8
λ_{KL} %	0.9263	5.000	14.4563	5.7673
λ_{KL} dB	-12.9310	-2.6135	4.5628	-1.1981
λ_{KLDiag} %	0.9037	4.9925	15.3820	5.9978
λ_{KLDiag} dB	-13.5464	-2.6071	4.5431	-1.0239
λ_{MSE} %	1.4933	7.9290	22.8974	9.4306
λ_{MSE} dB	-20.6199	-9.5392	-2.0927	-7.9476
	Link 1	Link 2	Link 3	Link 4
Power	0.120	0.118	0.120	0.056
Scenario 7				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	1.3	5.0	3.7	10.0
Data Rate (Kb/s)	700.0	285.1	179.0	272.6
bits/vector	19	8	5	8
λ_{KL} %	1.1599	5.7673	14.4563	5.7673
λ_{KL} dB	-11.2211	-1.1981	4.5628	-1.1981
λ_{KLDiag} %	1.1366	5.9978	15.3820	5.9978
λ_{KLDiag} dB	-11.5464	-1.0239	4.5431	-1.0239
λ_{MSE} %	1.9091	9.4306	22.8974	9.4306
λ_{MSE} dB	-18.6926	-7.9476	-2.0927	-7.9476
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.120	0.120	0.055

Table 4.7: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 2, 1, 1]$

	Session #			
	1	2	3	4
Scenario 8				
Delay Requirement (ms)	10.0	2.0	10.0	10.0
Delay Achieved (ms)	0.5	2.0	1.5	10.0
Data Rate (Kb/s)	326.1	133.4	83.9	168.3
bits/vector	9	4	2	5
λ_{KL} %	5.000	23.8526	.	14.4563
λ_{KL} dB	-2.6135	7.4631	.	4.5628
λ_{KLDiag} %	4.9925	23.8416	.	15.3820
λ_{KLDiag} dB	-2.6071	7.1900	.	4.5431
λ_{MSE} %	7.9290	36.1042	.	22.8974
λ_{MSE} dB	-9.5392	0.7404	.	-2.0927
	Link 1	Link 2	Link 3	Link 4
Power	0.112	0.120	0.120	0.050
Scenario 9				
Delay Requirement (ms)	10.0	10.0	5.0	5.0
Delay Achieved (ms)	5.0	10.0	5.0	5.0
Data Rate (Kb/s)	835.4	305.6	187.0	268.8
bits/vector	23	8	5	7
λ_{KL} %	0.8100	5.7673	14.4563	8.1023
λ_{KL} dB	-14.4640	-1.1981	4.5628	0.4721
λ_{KLDiag} %	0.7756	5.9978	15.3820	8.1649
λ_{KLDiag} dB	-14.2114	-1.0239	4.5431	0.4968
λ_{MSE} %	1.2661	9.4306	22.8974	12.7641
λ_{MSE} dB	-21.6926	-7.9476	-2.0927	-6.3498
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.120	0.118	0.120	0.058

Table 4.8: Simulation results for network and flow topology I with $\boldsymbol{\alpha_s} = [1, 1, 2, 2]$

	Session #			
	1	2	3	4
Scenario 10				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	1.2	5.0	3.8	10.0
Data Rate (Kb/s)	773.1	119.9	283.8	381.2
bits/vector	21	3	8	11
λ_{KL} %	0.9263	.	5.7673	3.1715
λ_{KL} dB	-12.9310	.	-1.1981	-5.1890
λ_{KLDiag} %	0.9037	.	5.9978	3.1731
λ_{KLDiag} dB	-13.5464	.	-1.0239	-5.1380
λ_{MSE} %	1.4933	.	9.4306	5.2822
λ_{MSE} dB	-20.6199	.	-7.9476	-11.9055
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.105	0.108	0.120	0.060

4.1.6 Set C: A Different Flow Topology

Simulations similar to Tables 4.4-4.5 are generated for the network and flow topology II in Fig. 4.12. The corresponding results are shown in Tables 4.9-4.10. The

Table 4.9: Simulation results for network and flow topology II with $\boldsymbol{\alpha_s} = [1, 1, 1, 1]$

	Session #			
	1	2	3	4
Scenario 11				
Delay Requirement (ms)	10.0	10.0	10.0	10.0
Delay Achieved (ms)	6.5	10.0	10.0	3.5
Data Rate (Kb/s)	347.3	192.9	197.3	433.9
bits/vector	10	5	5	12
λ_{KL} %	4.0321	14.4563	14.4563	2.8445
λ_{KL} dB	-4.0253	4.5628	4.5628	-5.9271
λ_{KLDiag} %	3.8744	15.3820	15.3820	2.8656
λ_{KLDiag} dB	-4.0123	4.5431	4.5431	-6.1425
λ_{MSE} %	6.6716	22.8974	22.8974	4.5944
λ_{MSE} dB	-10.4657	-2.0927	-2.0927	-12.7056
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.054	0.120	0.120	0.073
Scenario 12				
Delay Requirement (ms)	10.0	8.0	10.0	10.0
Delay Achieved (ms)	5.0	8.0	10.0	3.0
Data Rate (Kb/s)	334.2	188.8	194.8	434.1
bits/vector	9	5	5	12
λ_{KL} %	5.000	14.4563	14.4563	2.8445
λ_{KL} dB	-2.6135	4.5628	4.5628	-5.9271
λ_{KLDiag} %	4.9925	15.3820	15.3820	2.8656
λ_{KLDiag} dB	-2.6071	4.5431	4.5431	-6.1425
λ_{MSE} %	7.9290	22.8974	22.8974	4.5944
λ_{MSE} dB	-9.5392	-2.0927	-2.0927	-12.7056
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.052	0.120	0.120	0.074

same observations that were made in the previous scenarios can also be made here. We emphasize another point by the network and flow topology II: the result of rate-power allocation for any particular session is highly interdependent with the topology

of the other flows. For instance, Session 1 in flow topology II resembles Session 4 in flow topology I. But the results of the power, bits/vector rate allocations for Session 4 in the flow topology I (see Tables 4.4-4.5) is different from the results for Session 1 in the flow topology II when comparing the achieved delay (see Tables 4.9-4.10).

Table 4.10: Simulation results for network and flow topology II with $\boldsymbol{\alpha_s} = [1, 1, 1, 1]$

	Session #			
	1	2	3	4
Scenario 13				
Delay Requirement (ms)	10.0	5.0	10.0	10.0
Delay Achieved (ms)	3.2	5.0	10.0	1.8
Data Rate (Kb/s)	297.9	173.5	181.7	415.7
bits/vector	8	5	5	12
λ_{KL} %	5.7673	14.4563	14.4563	2.8445
λ_{KL} dB	-1.1981	4.5628	4.5628	-5.9271
λ_{KLDiag} %	5.9978	15.3820	15.3820	2.8656
λ_{KLDiag} dB	-1.0239	4.5431	4.5431	-6.1425
λ_{MSE} %	9.4306	22.8974	22.8974	4.5944
λ_{MSE} dB	-7.9476	-2.0927	-2.0927	-12.7056
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.051	0.120	0.120	0.077
Scenario 14				
Delay Requirement (ms)	10.0	2.0	10.0	10.0
Delay Achieved (ms)	1.3	2.0	10.0	0.7
Data Rate (Kb/s)	125.7	83.3	92.9	247.1
bits/vector	3	2	3	7
λ_{KL} %	.	.	.	8.1023
λ_{KL} dB	.	.	.	0.4721
λ_{KLDiag} %	.	.	.	8.1649
λ_{KLDiag} dB	.	.	.	0.4968
λ_{MSE} %	.	.	.	12.7641
λ_{MSE} dB	.	.	.	-6.3498
	Link 1	Link 2	Link 3	Link 4
Power (W)	0.046	0.120	0.120	0.092

4.2 Remarks

In summary, the results in these simulations demonstrated that by combining the network utility maximization problem developed in Chapter 2 and quantization method developed in Chapter 3, we can satisfy the minimum rate requirements of the data transfer, limits on power of transmitters, and delay requirements of sources. We furthermore were able to perform better in the classification of signals at the decoder by using the quantization schemes developed in Chapter 4.

CHAPTER 5

Conclusion and Future Work

5.1 Conclusion

In this dissertation, we studied two important problems that occur in quantization and transmission over multi-hop wireless networks. The first problem addresses resource allocation in a resource-limited network. We presented the problem of allocating resources of bandwidth and power in a multi-hop wireless network in a Network Utility Maximization (NUM) framework, specifically incorporating an average queuing delay requirement of the sessions into the NUM problem. We transformed the non-convex problem to a convex problem by a change of variable and assuming a high SIR scenario. We presented a distributed iterative algorithm solving the NUM problem. In particular, in our solution algorithm, both sources and links exchange information allowing for the NUM problem to solve for the session rate, power of transmitters, and delay share of each link in an iterative distributed manner. The simulations showed the performance of the solution and comparisons with the previously developed algorithms and demonstrated that average queuing delay requirement of the sessions were achieved.

The second problem we studied arises in scenarios where vector signals are quan-

tized and transferred to a receiver (decoder) over a communication network. At the receiver, not only the reconstruction of the signals is intended but also classifications of signals may be desirable. In the second part of this dissertation we presented a solution for quantization of signals for the purpose of obtaining a more accurate classification at the decoder. We employed high-rate theory for quantizer design. Therefore an optimal point density function that determines where the codepoints should lie in the space was derived. We chose the symmetric KL divergence between the conditional probabilities of classes given data before and after quantization as our distortion measure. The performance of this method on synthetically generated data was examined and observed to be superior in the task of classification of signals at the decoder. The tradeoff between the reproduction fidelity and classification accuracy was studied as well. Furthermore, we examined the effectiveness of our proposed method on real data sets with low dimensions. It was verified by the simulation results that the proposed method shows improvement in the classification results of real data sets as well.

We combined the two algorithms developed in this dissertation. We used the examples from the simulation results of each algorithm and built on top of them to present simulation results for combination of the algorithms developed in previous chapters. We demonstrated that taking advantage of the combination of the algorithm we were able to achieve the desired average queuing delay requirements, maximum power available, and better performance of the classification results at the decoder.

5.2 Proposed Future Research Directions

We believe that there is a lot of potential in the development of the second part of this research work beyond this dissertation. The emergence of many recent classification algorithms and the proof of their effectiveness and superiority can lead one to make an effort to design quantizers that jointly perform well when paired with the classifiers. One possible choice could be a joint Support Vector Machine Classifier Vector Quantizer. Furthermore, the developments in the area of dimensionality reduction and finding patterns in high-dimensional signals by finding a lower dimension manifold, can also be combined with vector quantization.

Furthermore, the combination of the algorithms developed in Chapter 2 and 3 can be devised through a joint cross-layer NUM and quantization method. This can be performed through an even larger augmentation of the basic NUM problem, an augmentation beyond what was presented in the Chapter 2 of this dissertation. This new larger augmentation should include the distortion measure of the quantizer in order to jointly solve the NUM-Quantization problem. It is expected that this optimization problem will be highly non-convex and would require a change of variables and other methods beyond what we presented in Chapter 2 to convert the problem into a convex optimization problem.

Practical implementation concerns of a quantizer is one important research extension to this dissertation. For high bit rates a full search quantizer is impractical, in terms of training, complexity and storage. Exploring different methods for simplifying higher rate quantizers and therefore avoiding a full search method would be a matter of further research. One of the methods that has shown superiority in this

regard is the implementation of the transform coder in [DR07].

In the following, we present another possible way to think of a combination of the algorithms developed in Chapter 2 and 3 of this dissertation that some readers may be interested to explore more.

5.2.1 NUM framework with Quantized messages

Consider the problem we studied in Chapter 2 of this dissertation, i.e., incorporating delay requirements of the sources into the congestion control algorithm. But let us assume that because of further network resource limitations we would like to avoid unnecessary overhead in the network resource usage. This overhead is introduced by message passing that was required for our iterative algorithm to achieve optimal resource allocation. Let us look at the following section of `Algorithm I` we developed in Chapter 2.

`(2) Each transmitter calculates a message`

$m_j(t)$ `based on values that can be determined at each node and passes it to all other`

`transmitters by a flooding protocol:`

$$m_j(t) = \frac{\lambda_j(t)\mathtt{SIR}_j(t)}{P_j(t)G_{jj}}. \tag{5.1}$$

`(3) Each transmitter updates its power based on`

$$P_\ell(t+1) = P_\ell(t) + \kappa_p \left(\frac{\lambda_\ell(t)}{P_\ell(t)} - \sum_{j \neq \ell} G_{j\ell} m_j(t) \right). \tag{5.2}$$

Realistically, because of network limitations the messages, $m_j(t)$, are not transmitted unquantized. Coarse quantization of the messages, $m_j(t)$, will affect the convergence of the sub-gradient method. The effect of quantization of messages in various network settings has been studied in a number of research works. The authors in [NOOT08] provided bounds on the convergence rate of the subgradient method under a quantized messages constraint. Furthermore, they showed that for their developed algorithm sources were able to achieve the optimal objective value of their utility within a certain error. In another research paper, the authors in [RN05] studied the effect of quantization of optimization variables in a wireless sensor network among other contributions. First they examined the convergence of their algorithm under constant step size for subgradient methods. They claimed that in many wireless sensor applications achieving a coarse estimate of the parameters may be an acceptable tradeoff if the amount of energy and bandwidth used by network is less than what is required to achieve a more accurate estimate. Furthermore, they showed that there will be some error introduced in the achieved optimal values if quantization is introduced. If the quantization is precise the amount of this error is very minimal. But for a coarser quantization, which is required for a resource constrained network, the rate of convergence is not affected considerably.

Quantizing the messages that are required for the iterative algorithms to converge to an optimal solution brings up the interesting problem of how to quantize the messages to assure the convergence and speed of convergence of the iterative algorithms. Design of a quantizer that minimizes the effect of quantization error

on the convergence performance of a specific iterative algorithm can be a matter of research interest. The authors in [CL10, DG13] studied the convergence behavior of distributed iterative algorithms and multi-agent systems with quantized message passing. They proposed two time-invariant quantization methods to minimize the effect of quantization error on the convergence. They also studied the effect of the design of quantizer on the convergence performance.

In another paper, [NB10], authors studied the effect of deterministic noise in sub-gradient methods. They discussed the convergence properties using different stepsize rules and they proved convergence to the optimal value within some tolerance.

All of these different scenarios suggest that there have been some good studies considering the effect of noise or quantization on the convergence properties of decentralized sub-gradient algorithms which was the heart of Chapter 2 of this dissertation. Furthermore, the idea of quantizing for a different purpose other than only reproduction fidelity, which was the core of Chapter 3 of this dissertation, has good merit in the study of how to quantize messages to achieve better convergence properties. We believe pursuing some of these suggested scenarios is a matter of further research and expansion to this dissertation. These expansions can be thought of as either the combination of the individual contribution of each part of this dissertation, as we mentioned in the first scenario we described above. Or they can be a more complex matter of studying the effect of different quantization methods on the convergence properties of the sub-gradient algorithm. Or even a whole new interesting topic of designing a quantizer that jointly optimizes reproduction fidelity, classification accuracy at the decoder and speed of convergence of the subgradient method.

APPENDIX A

Derivation of vector $\mathbf{d}(\hat{\mathbf{x}})$

Vector $\mathbf{d}(\hat{\mathbf{x}})$ is an n-element row vector whose elements are defined by

$$d_i(\hat{\mathbf{x}}) = \left.\frac{\partial d(\mathbf{x}, \hat{\mathbf{x}})}{\partial x_i}\right|_{\mathbf{x}=\hat{\mathbf{x}}}. \tag{A.1}$$

If $d(\mathbf{x}, \hat{\mathbf{x}})$ is defined as in (3.2) which is repeated here for ease of reading

$$d(\mathbf{x}, \hat{\mathbf{x}}) = D\Big(P(C|\mathbf{x})||P(C|\hat{\mathbf{x}})\Big) + D\Big(P(C|\hat{\mathbf{x}})||P(C|\mathbf{x})\Big) = \sum_{j=1}^{C}\left[P(C_j|\mathbf{x})\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})} + P(C_j|\hat{\mathbf{x}})\log\frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\right], \tag{A.2}$$

then

$$\begin{aligned} d_i(\hat{\mathbf{x}}) = \left.\frac{\partial d(\mathbf{x}, \hat{\mathbf{x}})}{\partial x_i}\right|_{\mathbf{x}=\hat{\mathbf{x}}} &= \frac{\partial}{\partial x_i}\sum_{j=1}^{C}\left[P(C_j|\mathbf{x})\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})} + P(C_j|\hat{\mathbf{x}})\log\frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}} \\ &= \sum_{j=1}^{C}\left[\frac{\partial}{\partial x_i}\left(P(C_j|\mathbf{x})\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}\right) + \frac{\partial}{\partial x_i}\left(P(C_j|\hat{\mathbf{x}})\log\frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\right)\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}} = \sum_{j=1}^{C} \\ &\left[\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}\frac{\partial}{\partial x_i}P(C_j|\mathbf{x}) + P(C_j|\mathbf{x})\frac{\partial}{\partial x_i}\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})} + P(C_j|\hat{\mathbf{x}})\frac{\partial}{\partial x_i}\log\frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}} \\ &= \sum_{j=1}^{C}\left[\log\frac{P(C_j|\mathbf{x})}{P(C_j|\hat{\mathbf{x}})}\frac{\partial}{\partial x_i}P(C_j|\mathbf{x}) + \frac{\partial}{\partial x_i}P(C_j|\mathbf{x}) - \frac{P(C_j|\hat{\mathbf{x}})}{P(C_j|\mathbf{x})}\frac{\partial}{\partial x_i}P(C_j|\mathbf{x})\right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}} = 0. \end{aligned} \tag{A.3}$$

APPENDIX B

$|\mathbf{D}|$ for 2-D-2-Class Scenarios

Let us assume that the distortion measure is defined as the following

$$d(\mathbf{x}, \hat{\mathbf{x}}) = \sum_{j=1}^{C} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})), \tag{B.1}$$

where f is any function of its arguments. Here, we will show that the determinant of the sensitivity matrix associated with $d(\mathbf{x}, \hat{\mathbf{x}})$ is zero if all the following conditions are satisfied

1. $\frac{\partial f(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{y}} = 0$.

2. Signals are two dimensional.

3. Number of classes is two, i.e., $C = 2$.

It is notable that the first condition holds for most distortion measures of choice.

$$D_{i,k}(\hat{\mathbf{x}}) = \frac{\partial}{\partial x_i}\frac{\partial}{\partial x_k} d(\mathbf{x},\hat{\mathbf{x}}) = \frac{\partial}{\partial x_i}\frac{\partial}{\partial x_k}\sum_{j=1}^{C} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}}))\Big|_{\mathbf{x}=\hat{\mathbf{x}}} =$$

$$\frac{\partial}{\partial x_i}\sum_{j=1}^{C}\frac{\partial}{\partial P(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial P(C_j|\mathbf{x})}{\partial x_k}\Big|_{\mathbf{x}=\hat{\mathbf{x}}} =$$

$$\sum_{j=1}^{C}\frac{\partial}{\partial P(C_j|\mathbf{x})}\frac{\partial}{\partial P(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial P(C_j|\mathbf{x})}{\partial x_i}.\frac{\partial P(C_j|\mathbf{x})}{\partial x_k} +$$

$$\frac{\partial}{\partial P(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial}{\partial x_i}\frac{\partial P(C_j|\mathbf{x})}{\partial x_k}\Big|_{\mathbf{x}=\hat{\mathbf{x}}} =$$

$$\sum_{j=1}^{C}\frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial P(C_j|\mathbf{x})}{\partial x_i}.\frac{\partial P(C_j|\mathbf{x})}{\partial x_k} +$$

$$\frac{\partial}{\partial P(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial^2}{\partial x_i \partial x_k} P(C_j|\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad \text{(B.2)}$$

For a two-dimensional signal D is a 2×2 matrix whose elements can be represented as

$$D_{1,1}(\hat{\mathbf{x}}) = \sum_{j=1}^{C}\frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).[\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}]^2 +$$

$$\frac{\partial}{\partial P(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial^2}{\partial x_1^2} P(C_j|\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}}, \quad \text{(B.3)}$$

$$D_{1,2}(\hat{\mathbf{x}}) = \sum_{j=1}^{C}\frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}.\frac{\partial P(C_j|\mathbf{x})}{\partial x_2} +$$

$$\frac{\partial}{\partial P(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial^2}{\partial x_1 \partial x_2} P(C_j|\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}} = D_{2,1}(\hat{\mathbf{x}}), \quad \text{(B.4)}$$

and

$$D_{2,2}(\hat{\mathbf{x}}) = \sum_{j=1}^{C}\frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).[\frac{\partial P(C_j|\mathbf{x})}{\partial x_2}]^2 +$$

$$\frac{\partial}{\partial P(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial^2}{\partial x_2^2} P(C_j|\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad \text{(B.5)}$$

Under the condition that $\frac{\partial f(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{y}} = 0$, the aforementioned elements of matrix D reduce to the following

$$D_{1,1}(\hat{\mathbf{x}}) = \sum_{j=1}^{C} \frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).[\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}]^2\Big|_{\mathbf{x}=\hat{\mathbf{x}}}, \quad \text{(B.6)}$$

$$D_{1,2}(\hat{\mathbf{x}}) = \sum_{j=1}^{C} \frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}.\frac{\partial P(C_j|\mathbf{x})}{\partial x_2}\Big|_{\mathbf{x}=\hat{\mathbf{x}}} = D_{2,1}(\hat{\mathbf{x}}), \quad \text{(B.7)}$$

and

$$D_{2,2}(\hat{\mathbf{x}}) = \sum_{j=1}^{C} \frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).[\frac{\partial P(C_j|\mathbf{x})}{\partial x_2}]^2\Big|_{\mathbf{x}=\hat{\mathbf{x}}}. \quad \text{(B.8)}$$

And therefore

$$\begin{aligned}
|D| = D_{1,1}(\hat{\mathbf{x}})D_{2,2}(\hat{\mathbf{x}}) - D_{1,2}(\hat{\mathbf{x}})D_{2,1}(\hat{\mathbf{x}}) = \\
\cancel{\sum_{j=1}^{C} \left[\frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}}))\right]^2 \left[\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}\frac{\partial P(C_j|\mathbf{x})}{\partial x_2}\right]^2} + \\
\sum_{j,\ell=1, j\neq\ell}^{C} \frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).[\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}]^2 \\
\frac{\partial^2}{\partial P^2(C_\ell|\mathbf{x})} f(P(C_\ell|\mathbf{x}), P(C_\ell|\hat{\mathbf{x}})).[\frac{\partial P(C_\ell|\mathbf{x})}{\partial x_2}]^2 - \\
\cancel{\sum_{j=1}^{C} \left[\frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}}))\right]^2 \left[\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}\frac{\partial P(C_j|\mathbf{x})}{\partial x_2}\right]^2} - \\
\sum_{j,\ell=1, j\neq\ell}^{C} \frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})).\frac{\partial P(C_j|\mathbf{x})}{\partial x_1}.\frac{\partial P(C_j|\mathbf{x})}{\partial x_2}. \\
\frac{\partial^2}{\partial P^2(C_\ell|\mathbf{x})} f(P(C_\ell|\mathbf{x}), P(C_\ell|\hat{\mathbf{x}})).\frac{\partial P(C_\ell|\mathbf{x})}{\partial x_1}.\frac{\partial P(C_\ell|\mathbf{x})}{\partial x_2}. \quad \text{(B.9)}
\end{aligned}$$

Furthermore $|D|$ simplifies to

$$|D| = \sum_{j,\ell=1,j\neq\ell}^{C} \frac{\partial^2}{\partial P^2(C_j|\mathbf{x})} f(P(C_j|\mathbf{x}), P(C_j|\hat{\mathbf{x}})). \frac{\partial^2}{\partial P^2(C_\ell|\mathbf{x})} f(P(C_\ell|\mathbf{x}), P(C_\ell|\hat{\mathbf{x}}))$$
$$\left[\left[\frac{\partial P(C_j|\mathbf{x})}{\partial x_1} \frac{\partial P(C_\ell|\mathbf{x})}{\partial x_2} \right]^2 - \frac{\partial P(C_j|\mathbf{x})}{\partial x_1} . \frac{\partial P(C_j|\mathbf{x})}{\partial x_2} \frac{\partial P(C_\ell|\mathbf{x})}{\partial x_1} . \frac{\partial P(C_\ell|\mathbf{x})}{\partial x_2} \right] \Bigg|_{\mathbf{x}=\hat{\mathbf{x}}} . \quad \text{(B.10)}$$

Applying the last of the conditions stated at the beginning of this appendix, i.e., $C = 2$, we conclude that

$$\begin{aligned} |D| = {} & \frac{\partial^2}{\partial P^2(C_1|\mathbf{x})} f(P(C_1|\mathbf{x}), P(C_1|\hat{\mathbf{x}})). \frac{\partial^2}{\partial P^2(C_2|\mathbf{x})} f(P(C_2|\mathbf{x}), P(C_2|\hat{\mathbf{x}})) \\ & \left[\left[\frac{\partial P(C_1|\mathbf{x})}{\partial x_1} \frac{\partial P(C_2|\mathbf{x})}{\partial x_2} \right]^2 + \left[\frac{\partial P(C_2|\mathbf{x})}{\partial x_1} \frac{\partial P(C_1|\mathbf{x})}{\partial x_2} \right]^2 - \right. \\ & \left. 2 \frac{\partial P(C_1|\mathbf{x})}{\partial x_1} . \frac{\partial P(C_1|\mathbf{x})}{\partial x_2} \frac{\partial P(C_2|\mathbf{x})}{\partial x_1} . \frac{\partial P(C_2|\mathbf{x})}{\partial x_2} \right] \Bigg|_{\mathbf{x}=\hat{\mathbf{x}}} = \\ & \frac{\partial^2}{\partial P^2(C_1|\mathbf{x})} f(P(C_1|\mathbf{x}), P(C_1|\hat{\mathbf{x}})). \frac{\partial^2}{\partial P^2(C_2|\mathbf{x})} f(P(C_2|\mathbf{x}), P(C_2|\hat{\mathbf{x}})) \\ & \left[\frac{\partial P(C_1|\mathbf{x})}{\partial x_1} \frac{\partial P(C_2|\mathbf{x})}{\partial x_2} - \frac{\partial P(C_2|\mathbf{x})}{\partial x_1} \frac{\partial P(C_1|\mathbf{x})}{\partial x_2} \right]^2 \Bigg|_{\mathbf{x}=\hat{\mathbf{x}}} \end{aligned} \quad \text{(B.11)}$$

As the last step we will show that the last square bracket term is zero if $C = 2$ and therefore that will prove that $|D| = 0$ under the aforementioned conditions. When we only have two classes, the following holds

$$P(C_1|\mathbf{x}) + P(C_2|\mathbf{x}) = 1, \quad \text{(B.12)}$$

and therefore

$$P(C_2|\mathbf{x}) = 1 - P(C_1|\mathbf{x}). \quad \text{(B.13)}$$

Replacing this in the last bracket term in (B.11) yields

$$\frac{\partial P(C_1|\mathbf{x})}{\partial x_1}\frac{\partial P(C_2|\mathbf{x})}{\partial x_2} - \frac{\partial P(C_2|\mathbf{x})}{\partial x_1}\frac{\partial P(C_1|\mathbf{x})}{\partial x_2} =$$
$$\frac{\partial P(C_1|\mathbf{x})}{\partial x_1}\frac{\partial [1 - P(C_1|\mathbf{x})]}{\partial x_2} - \frac{\partial [1 - P(C_1|\mathbf{x})]}{\partial x_1}\frac{\partial P(C_1|\mathbf{x})}{\partial x_2} =$$
$$- \frac{\partial P(C_1|\mathbf{x})}{\partial x_1}\frac{\partial P(C_1|\mathbf{x})}{\partial x_2} + \frac{\partial P(C_1|\mathbf{x})}{\partial x_1}\frac{\partial P(C_1|\mathbf{x})}{\partial x_2} = 0. \quad \text{(B.14)}$$

Bibliography

[Bam98] N. Bambos, *Toward power-sensitive network architectures in wireless communications: concepts, issues, and design aspects*, Personal Communications, IEEE **5** (1998), no. 3, 50 –59.

[BB99] Dimitri P. Bertsekas and Dimitri P. Bertsekas, *Nonlinear programming*, 2nd ed., Athena Scientific, September 1999.

[BG92] Dimitri P. Bertsekas and Gallager, *Data networks (2nd edition)*, 2 ed., Prentice Hall, January 1992.

[BV04] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge University Press, March 2004.

[CB04] Mung Chiang and J. Bell, *Balancing supply and demand of bandwidth in wireless cellular networks: utility maximization over powers and rates*, INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, vol. 4, 7-11 2004, pp. 2800 – 2811 vol.4.

[Chi05] Mung Chiang, *Balancing transport and physical layers in wireless multihop networks: jointly optimal congestion control and power control*, Selected Areas in Communications, IEEE Journal on **23** (2005), no. 1, 104 – 116.

[CL10] Ying Cui and V.K.N. Lau, *Convergence-optimal quantizer design of distributed contraction-based iterative algorithms with quantized message passing*, Signal Processing, IEEE Transactions on **58** (2010), no. 10, 5196–5205.

[DFMP07] Behzad Mohammadi Dogahe, Xingzhe Fan, Manohar N. Murthi, and Kamal Premaratne, *Balancing power and rate to achieve bounded average delay in wireless networks*, Military Communications Conference, 2007. MILCOM 2007. IEEE, 29-31 2007, pp. 1 –7.

[DG13] Runsha Dong and Zhiyong Geng, *Design and analysis of quantizer for multi-agent systems with a limited rate of communication data*, Communications in Nonlinear Science and Numerical Simulation **18** (2013), no. 2, 282 – 290.

[DM11] B.M. Dogahe and M.N. Murthi, *Quantization for classification accuracy in high-rate quantizers*, Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop (DSP/SPE), 2011 IEEE, Jan 2011, pp. 277–282.

[DMFP10] BehzadMohammadi Dogahe, ManoharN. Murthi, Xingzhe Fan, and Kamal Premaratne, *A distributed congestion and power control algorithm to achieve bounded average queuing delay in wireless networks*, Telecommunication Systems **44** (2010), no. 3-4, 307–320 (English).

[DR07] E.R. Duni and B.D. Rao, *A high-rate optimal transform coder with gaussian mixture companders*, Audio, Speech, and Language Processing, IEEE Transactions on **15** (2007), no. 3, 770–783.

[Ger79] A. Gersho, *Asymptotically optimal block quantization*, Information Theory, IEEE Transactions on **25** (1979), no. 4, 373–380.

[GG92] A. Gersho and R. M. Gray, *Vector quantization and signal compression.*

[GH03] R. Gupta and III Hero, A.O., *High-rate vector quantization for detection*, Information Theory, IEEE Transactions on **49** (2003), no. 8, 1951 – 1969.

[GK77] M. Gerla and L. Kleinrock, *On the topological design of distributed computer networks*, Communications, IEEE Transactions on **25** (1977), no. 1, 48 – 60.

[GN98] R.M. Gray and D.L. Neuhoff, *Quantization*, Information Theory, IEEE Transactions on **44** (1998), no. 6, 2325 –2383.

[Gol05] Andrea Goldsmith, *Wireless communications*, Cambridge University Press, New York, NY, USA, 2005.

[GR95] W.R. Gardner and B.D. Rao, *Theoretical analysis of the high-rate vector quantization of lpc parameters*, Speech and Audio Processing, IEEE Transactions on **3** (1995), no. 5, 367 –381.

[Gra89] R. M. Gray, *Source coding theory.*

[JCOB02] D. Julian, Mung Chiang, D. O'Neill, and S. Boyd, *Qos and fairness constrained convex optimization of resource allocation for wireless cellular and ad hoc networks*, INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 2, 23-27 2002, pp. 477 – 486.

[JXB03] M. Johansson, Lin Xiao, and S. Boyd, *Simultaneous routing and power allocation in cdma wireless data networks*, Communications, 2003. ICC '03. IEEE International Conference on, vol. 1, 11-15 2003, pp. 51 – 55 vol.1.

[Kas77] S. Kassam, *Optimum quantization for signal detection*, Communications, IEEE Transactions on **25** (1977), no. 5, 479 – 484.

[KMT98] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, *Rate control for communication networks: Shadow prices, proportional fairness and stability*, The Journal of the Operational Research Society **49** (1998), no. 3, 237–252.

[LCC06a] J.-W. Lee, Mung Chiang, and A.R. Calderbank, *Price-based distributed algorithms for rate-reliability tradeoff in network utility maximization*, Selected Areas in Communications, IEEE Journal on **24** (2006), no. 5, 962 – 976.

[LCC06b] Jang-Won Lee, M. Chiang, and R.A. Calderbank, *Jointly optimal congestion and contention control based on network utility maximization*, Communications Letters, IEEE **10** (2006), no. 3, 216 – 218.

[LCCD09] Ying Li, Mung Chiang, A.R. Calderbank, and S.N. Diggavi, *Optimal rate-reliability-delay tradeoff in networks with composite links*, Communications, IEEE Transactions on **57** (2009), no. 5, 1390 –1401.

[LJ07] M.A. Lexa and D.H. Johnson, *Joint optimization of distributed broadcast quantization systems for classification*, Data Compression Conference, 2007. DCC '07, mar. 2007, pp. 363 –374.

[LL99] S.H. Low and D.E. Lapsley, *Optimization flow control. i. basic algorithm and convergence*, Networking, IEEE/ACM Transactions on **7** (1999), no. 6, 861 –874.

[LMS04] Jang-Won Lee, R.R. Mazumdar, and N.B. Shroff, *Opportunistic power scheduling for multi-server wireless systems with minimum performance constraints*, INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, vol. 2, 7-11 2004, pp. 1067 – 1077 vol.2.

[Low03] S.H. Low, *A duality model of tcp and queue management algorithms*, Networking, IEEE/ACM Transactions on **11** (2003), no. 4, 525 – 536.

[LPCC11] Ying Li, Antonis Papachristodoulou, Mung Chiang, and A. Robert Calderbank, *Congestion control and its stability in networks with delay sensitive traffic*, Computer Networks **55** (2011), no. 1, 20 – 32.

[LPW01] Steven Low, Larry Peterson, and Limin Wang, *Understanding tcp vegas: A duality model*, In Proceedings of ACM Sigmetrics, 2001, pp. 226–235.

[MB02] P. Marbach and R. Berry, *Downlink resource allocation and pricing for wireless networks*, INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 3, 2002, pp. 1470 – 1479 vol.3.

[MCLG06] R. Madan, S. Cui, S. Lal, and A. Goldsmith, *Cross-layer design for lifetime maximization in interference-limited wireless sensor networks*, Wireless Communications, IEEE Transactions on **5** (2006), no. 11, 3142 –3152.

[NB10] Angelia Nedi and DimitriP. Bertsekas, *The effect of deterministic noise in subgradient methods*, Mathematical Programming **125** (2010), no. 1, 75–99 (English).

[NOOT08] A Nedic, A Olshevsky, A Ozdaglar, and J.N. Tsitsiklis, *Distributed subgradient methods and quantization effects*, Decision and Control, 2008. CDC 2008. 47th IEEE Conference on, Dec 2008, pp. 4177–4184.

[PC06] D.P. Palomar and Mung Chiang, *A tutorial on decomposition methods for network utility maximization*, Selected Areas in Communications, IEEE Journal on **24** (2006), no. 8, 1439 –1451.

[PDE08] J. Papandriopoulos, S. Dey, and J. Evans, *Optimal and distributed protocols for cross-layer design of physical and transport layers in manets*, Networking, IEEE/ACM Transactions on **16** (2008), no. 6, 1392 –1405.

[PPG+96] K.O. Perlmutter, S.M. Perlmutter, R.M. Gray, R.A. Olshen, and K.L. Oehler, *Bayes risk weighted vector quantization with posterior estimation for image compression and classification*, Image Processing, IEEE Transactions on **5** (1996), no. 2, 347 –360.

[QBX14] Fan Qiu, Jia Bai, and Yuan Xue, *Optimal rate allocation in wireless networks with delay constraints*, Ad Hoc Networks **13, Part B** (2014), 282 – 295.

[RLM02] J. Razavilar, K.J.R. Liu, and S.I. Marcus, *Jointly optimized bit-rate/delay control policy for wireless packet networks with fading channels*, Communications, IEEE Transactions on **50** (2002), no. 3, 484 –494.

[RN05] M.G. Rabbat and R.D. Nowak, *Quantized incremental algorithms for distributed optimization*, Selected Areas in Communications, IEEE Journal on **23** (2005), no. 4, 798–808.

[SLGY07] M. Saad, A. Leon-Garcia, and Wei Yu, *Optimal network rate allocation under end-to-end quality-of-service requirements*, Network and Service Management, IEEE Transactions on **4** (2007), no. 3, 40 –49.

[SS77] F. Sha and L.K. Saul, *Applications of ali-silvey distance measures in the design generalized quantizers for binary decision systems*, Communications, IEEE Transactions on **25** (1977), no. 9, 893 – 900.

[VB10] Joffrey Villard and Pascal Bianchi, *High-rate vector quantization for the neyman-pearson detection of some stationary mixing processes*, Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on, 13-18 2010, pp. 1608 –1612.

[VV08] K.R. Varshney and L.R. Varshney, *Quantization of prior probabilities for hypothesis testing*, Signal Processing, IEEE Transactions on **56** (2008), no. 10, 4553 –4562.