ABSTRACT

XU, PENG. QoS Provisioning and Pricing in Multiservice Networks:
Optimal and Adaptive Control over Measurement-based Scheduling. (Under the direction of Professor Michael Devetsikiotis).

In order to ensure efficient performance under inherently and highly variable traffic in multiservice networks, we propose a generalized adaptive and optimal control framework to handle the resource allocation. Even though this framework addresses rigid Quality of Service concerns for the deterministic delay-bound classes by reserving part of the link capacity and employing appropriate admission control and traffic shaping schemes, our research actually emphasizes the *adaptive* and *optimal* control of the shared resources for the flexible delay-bound classes. Therefore, the resource allocation is delivered by a subsystem of this generalized framework, the measurement-based optimal resource allocation (MBORA) system.

By applying a simple threshold $(\theta 1, \theta 2)$ policy, we first validate the advantages of the adaptivity of our proposed framework through extensive simulation results. Then we introduce a generalized profit-oriented formulation inside decision module of MBORA system, that supplies the network provider with criteria in terms of profit, by leveraging the utility charge revenue and delay-incurred cost. The optimal resource allocation will be affected by the various types of pricing models together with the different levels of service guarantee constraints. As a case study, we investigate this generalized profit-oriented formulation under generalized service models. Combining further with a linear pricing model subject to average queue delay constraints, we propose a fast algorithm for online dynamic and optimal resource allocation under this specific scenario.

Finally, we propose a delay-sensitive nonlinear pricing model for the generalized profit-oriented formulation, that realizes two-tier delay differentiation. By better understanding the fluid queueing model, we propose a generalized solution strategy for linear, nonlinear or mixed pricing models that is free of the dimensionality problem and amenable to online implementation.

QoS Provisioning and Pricing in Multiservice Networks: Optimal and Adaptive Control over Measurement-based Scheduling

by

Peng Xu

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial satisfaction of the
requirements for the Degree of
Doctor of Philosophy

Department of Electrical and Computer Engineering

Raleigh

2005

Approved By:

Dr. Wenye Wang	Dr. Peng Ning
Dr. Ioannis Viniotis	Dr. George Michailidis

Dr. Michael Devetsikiotis
Chair of Advisory Committee

To my parents and the past five years in Raleigh.

致我的父母和在罗利逝去的五年时光。

Biography

Peng Xu was born on 18th of May 1975 in Harbin, a beautiful city in Northeast China. Since the age of 8, he began his long journey of school life, that lasts for almost 22 years. The interesting thing is that 12 years of his study before college was spent on 3 schools all located in a long road named 'FengDou', which means 'struggle' in English. Staying in his hometown, he then received his B.S and M.S degree in Electrical Engineering from Harbin Engineering University in 1998 and 2000, respectively. Shortly after he turned the age of 25, he left his sweet hometown for Ph.D study in North Carolina State University, that proved to be the growing pain for him to be independent mentally and financially. Five years of study and research on the networking technologies really strengthened his conviction to further pursue his career in the fast-growing wireless networking and digitalizing TV communication business, which is believed to have a huge impact on human's life in the future.

Acknowledgements

First and foremost, I am profoundly indebted to my parents whose endless love and care always supported me through ups and downs in the past five years, even though we were thousands miles away from each other. Without their encouragement, Ph.D study in U.S could be a wildest dream for me. I always admire my father for his generosity and optimistic personality and deeply believe that it was the way to win the friendship and every challenge you face down the road of life. Knowing the slackness of her eldest son, my mother asked her calligraphy teacher four years ago to write a banner with her favorite motto, 'God reward the diligent.', that is still hanged on the wall of my apartment in Raleigh.

I would like to give my sincere appreciation to my advisor, Dr. Michael Devetsikiotis who suffered a lot from my last-minute report submission and beard his patience on the development of my research during the period of my Ph.D study. Without his financial support and constant recommendations to the graduate school for GSSP issues, it would be impossible for me to finish this dissertation. I am also deeply thankful to Dr. George Michailidis, the professor from Statistics Department, University of Michigan. He was like the second advisor to me and pulled me out of the deep hole when I was stuck in my research back two years ago.

Besides the guidance from these two professors, I feel very fortunate to meet my friends in NCSU, with whom I went through so many unforgettable moments, no matter what was joyful or frustrated. I enjoyed the beer and cheerful debate with Xiang Zhou, Peng Geng, Cexiong Fu and Weiqing Jin at the weekend. And I am also thankful to the couple, Xinbing Wang and Ping Liu for their generosity and friendship to feed a lazy bachelor's stomach with a rich dinner. Especially, I am grateful to Xiang Zhou, Yong Wang from Operation Research Program for the helpful discussions on the optimization problem, and Xinbing Wang for sharing the frustrations over our courses of the networking research.

Last but not the least, I want to express my gratitude to my girlfriend, Rong Peng who turned the last part of my tedious Ph.D study life into the joyful and fruitful experience. Her tremendous love and support will be cherished in my heart forever.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Network technologies have evolved beyond expectation since the birth of the Internet. Traditionally, the main applications in the Internet are best-effort type of services, such as webpage browsing and email. However, the tendency of current applications is shifting to real-time applications (voice and video), that need guaranteed QoS performance. Along with the growing variety of application types, we can also be bold to predict the transition from the traditional wired networks to the wireless networks in the near future by observing the emergence of 3G systems [17, 27], WLAN [1, 8], WiMax and more. More generally, access networks including cable and DSL systems, as well as wireless systems based on WiFi and WiMax are becoming increasingly ambitious in terms of their multimedia/multiservice capabilities (data, TV, voice, video and games), continuing on a path towards retailing of services on demand. Hence, both in wired and wireless networks, QoS provisioning combined with rationalizing resources and achieving viable finances via proper pricing, remains one of the most critical and challenging issues for multi-service networks in the research and industrial fields.

Another critical issue involved is the corresponding development of pricing schemes [30, 31, 35, 38, 39], now that the differentiated services need to be delivered to the users in such multi-service networks. It is more flexible and efficient to have the pricing

scheme based on the usage and type of the service, instead of the flat pricing scheme that only supports the single-type service networks.

In order to solve both challenging issues, researchers have performed comprehensive studies and investigations on how to design efficient QoS provisioning and optimal pricing schemes in multi-service networks from different perspectives, including admission control schemes for accessing the networks, traffic engineering schemes for balancing the traffic loads through the networks, and the appropriate policing and scheduling schemes for the traffic delivery. Through our studies, it is not difficult to conclude that there always exist two contrasting approaches for each perspective, *static* vs. *dynamic*. The *static* approach has the edge over the *dynamic* one due to its simplicity, while *dynamic* approach has the advantage in terms of better QoS guarantees and optimal control for differentiated services, given the highly varied traffic loads.

Inspired by the dynamic approach, we propose a generalized adaptive and optimal control framework for measurement-based scheduling, in order to achieve QoS requirements and differentiated pricing in multi-service networks.

In this chapter, we first discuss the motivation and objective behind our proposed framework. Then our contributions on this topic are summarized. Finally, we provide an outline of this thesis.

1.1 Motivation and Objective

In multi-service networks, differentiated QoS provisioning and pricing are the key issues for efficient multiplexing of classes, connections or flows on a shared link.

Traditionally, a *static* resource allocation is designed to satisfy QoS requirements by following the contract or agreement between the network provider and the users. However, it also pushes the burden on the user side to characterize their traffic, which is obviously not always convenient or practical. With the advent of bandwidth and delay-sensitive applications such as voice over IP (VoIP), video-conferencing, online gaming, interactive television, etc., a static scheme is far from the flexibility required

given highly adaptive traffic load situations. It also implies that *static* bandwidth reservation protocols accompanied by over-provisioning of network links lead to a significant under-utilization of available resources, while QoS can not be guaranteed otherwise.

Therefore, a *dynamic* resource allocation scheme has been put forward in order to match the inherent high variability of the traffic characteristics. It is achieved by allocating the corresponding resources (bandwidth) based on closely tracking the prevailing traffic characteristics. Under this approach, QoS guarantees depend on the two critical steps: first, traffic monitoring and estimation, and, second, decision policy on the resource allocation. Traffic monitoring and estimation should be accurate enough in order to make sound decisions for the resource allocation.

Dynamic resource allocation can be achieved by schedulers capable of differentiating among traffic classes through our study. In weighed round-robin (WRR) and weighted fair queuing (WFQ), the bandwidth for each class is determined by the weights associated with this class. We can realize *dynamic* resource allocation by simply updating the weights of the schedulers among the different classes over the periodic traffic monitoring time.

Meanwhile, this *dynamic* scheme also needs to be "smart" enough to reach the optimal decision for the resource allocation based on the specific optimization models, which can be formulated from the combined consideration of pricing models and QoS constraints. In this way, the *dynamic* scheme can react to the variation of the traffic rate with optimal decisions for resource allocation. That is also our motivation for putting forward a profit-oriented optimization model for the decision decision policy.

1.2 Contributions of this Dissertation

This dissertation is dedicated to the study on QoS provisioning and pricing schemes for multi-service networks. Our main focus is to design an adaptive and optimal framework and algorithms for online implementation, that can handle the problem of QoS provisioning combined with economics. During the course of our study, we have

made the following contributions:

1. Generalized Adaptive and Optimal Control Framework:
   We introduced a generalized adaptive and optimization framework in order to meet the variety of service requirements from end users. This framework provides an efficient mechanism to differentiate the resources for the deterministic and flexible delay-bound classes. Targeting the shared resources for the flexible delay-bound class, we motivated the need to investigate measurement-based optimal resource allocation (MBORA) system inside this generalized framework.

2. Simple Threshold Policy:
   Starting from the emphasis on the adaptivity of the proposed framework, we analyzed and investigated the performance by applying a simple threshold decision policy, under the assumption that the QoS requirement for each class can be satisfied by allocating the corresponding bandwidth a level no lower than the measurement estimation.

   Dynamic counterparts of WRR and WFQ are evaluated and compared with the static ones by running extensive simulations. Finally, we concluded that DWFQ achieves the best performance. Also from this conclusion and the basic framework, we gained confidence to approach our next step: *Generalized Profit-Oriented Formulation*.

3. Generalized Profit-Oriented Formulation:
   In order to realize the optimal control of our proposed generalized framework, we formulated a generalized *profit center* optimization model for a network node that takes into consideration a pricing model for the various classes of users, as well as service guarantee constraints, such as packet loss rates and queueing delays.

   From the network provider's perspective, the model allows one to balance the trade-off between the revenue of utility charge and delay-incurred cost, in order to achieve the maximum profit from the system. Meanwhile, by varying the

adopted pricing models and optional service guarantee constraints, the network provider achieves more flexibility in the control of the resources.

4. Profit-Oriented Formulation under Liner Pricing Models:
   As a case study, we investigated the profit-oriented formulation under the assumption of generalized service models. Also in this scenario, we adopted a linear pricing model subject to average queue delay constraints. The properties of the optimal solution under these assumptions were analyzed and a fast, low complexity algorithm was proposed for the online dynamic resource allocation. Finally, we validated the proposed scheme through simulation experiments under different traffic scenarios.

5. Profit-Oriented Formulation under Nonlinear Pricing Models:
   We proposed a new delay-sensitive nonlinear pricing model. Compared with the linear counterpart, it realizes a two-tier differentiation pricing scheme, that is the differentiation between the classes and the differentiation inside the class.

   We also attempted to tackle the problem of calculating the optimal solution for more generalized n classes, instead of the limited number of classes in generalized service models. With better understanding on the fluid queueing model, we proposed a generalized solution strategy for linear, nonlinear or mixed pricing models.

1.3 Outline

The rest of the dissertation is organized as follows: In Chapter 2, we will discuss the background and the related work that led to our proposed scheme. Chapter 3 introduces our problem statement along with our proposed generalized adaptive and optimal control framework. Our generalized framework under simple threshold policy is analyzed and evaluated in Chapter 4. Chapter 5 presents our generalized profit-oriented formulation. Subsequently, in Chapter 6, we begin our study of the profit-oriented formulation with linear pricing scheme under generalized service models.

Chapter 7 introduces the nonlinear pricing scheme for the generalized profit-oriented formulation and gives the corresponding generalized solution strategy. And Chapter 8 summarizes the conclusion for this dissertation. Finally, the extended study on performance evaluation for Chapter 4 is discussed in Appendix A; the supermodular definition and the related theorems for Chapter 6 are given in Appendix B; and the related definitions and proofs for Chapter 7 are presented in Appendix C.

Chapter 2

Background and Related Work

In this chapter, we will discuss the background of our research study from four areas. They are traffic modeling, QoS, measurement estimation and scheduling algorithms respectively. At the end, we introduce the related work that motivates us to propose the generalized adaptive and optimal scheduling framework.

2.1 Traffic Modeling

2.1.1 Introduction

Traffic modeling is imperative to network design, management and simulation. In the absence of real-time network traffic, accurate traffic modeling is the key for performance evaluation, which is the most critical issue in providing QoS for different users and applications in the wired network. A good traffic modeling can give the accurate statistical characteristics of real network traffic, which in turn provides the better insight of network to the researchers and engineers, thus the area of network traffic modeling is well-studied in the past 20 years.

Traditionally, the network traffic is characterized by *short ranged dependent* (SRD)

models, which can be characterized by the following properties:

1. The *autocorrelation*(ACF) decays exponentially *i.e.* $r(k) \sim \rho^k, (0 < \rho < 1)$.
2. The ACF is summable *i.e.* $\sum r(k) < \infty$.
3. The *power spectral density*(PSD) is finite at the origin.
4. The *index of dispersion counts*(IDC) converges to a constant value.

SRD models can be classified into the following categories: Renewal models, Markov-based models, and Autoregressive-type models. All the details can be found in [63].

SRD models are widely accepted owing to their analytical tractability until 1991, because in that year Will Leland and Daniel Wilson presented the first analysis of high resolution Ethernet LAN traffic [36], in which the famous Bellcore traces are collected by the high-resolution measurement and the burstiness of Ethernet LAN traffic in many time scales is concluded. Since then, the traditional SRD models are challenged due to their incompetence in describing the burstiness of the network traffic in multiple time scales.

For example, Figure 2.1 shows Poisson traffic in the different time scales, which shows the large variation and burstiness in the fine scale, but smoothes or flattens in the coarse scale. Leland *et al.* [55] introduced the self-similar concept, which is coined by Mandelbrot, to describe the burstiness successfully based on their previous work on Ethernet LAN traffic in 1991. This can be regarded as the milestone paper in the traffic modeling which lead to the new phase of research on *long range dependent* (LRD) models. In 1994, Vern Paxson and Sally Floyd further verified the self-similarity in Wide Area Network(WAN) [46]. Since then, LRD or fractal models became the dominant research issues in the wired network, due to the validation in both LAN and WAN.

Most recently, Reidi *et al.* [49] proposed the shift from the monofractal models to the multifractal models due to the insufficiency of monfracal models in describing the higher order of statistic of data traces.

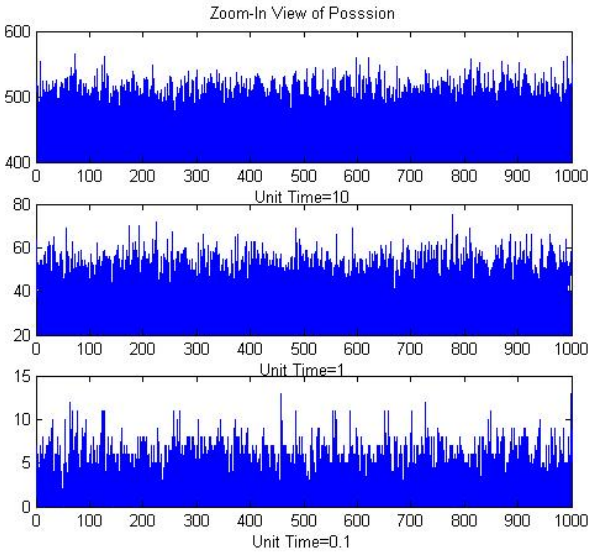


Figure 2.1: Poisson Traffic in Different Time Scales

In the following subsection, we first briefly review LRD models concepts and definitions, then we discuss the LRD models in general, with the emphasis on the one we adopted throughout our study and simulation validation, on-off Fractal Modulated Poisson process (FMPP).

2.1.2 Long Range Dependent Models

Compared with SRD models, the ACF of LRD models decays slower than exponential, which makes them capable to capture the real burstiness nature of network traffic. Let us start with the concepts and definitions.

Concepts and Definitions

Let $X = \{X_t : t = 0, 1...\}$ be a wide-sense stationary(WSS) process with mean μ and variance σ^2. The process X can be said to be *long-range dependence* [60]if:

$$r(k) \sim k^{-D} L_1(k) \quad as \;\; k \to \infty \tag{2.1}$$

L_1 varies slowly at ∞. Or equivalently,

$$f(\lambda) \sim \lambda^{-\alpha} L_2(\lambda) \quad as \;\; \lambda \to 0 \tag{2.2}$$

$0 < \alpha < 1$ and L_2 is a slowly varying function. And $f(\lambda) = \sum_k r(k) e^{jk\lambda}$ denotes the spectral density. Since $f(0) \to \infty$, or alternatively $\sum_k r(k) \to \infty$, the ACF for LRD process is not summarable.

A continuous-time process $Y = \{Y(t), t \geq 0\}$ is *self-similar* with self-similarity parameter H if it satisfies the condition [60]:

$$Y(t) = a^{-H} Y(at), \;\; \forall t \geq 0, \;\; \forall a > 0, \;\; 0 < H < 1 \tag{2.3}$$

where the equality is in the sense of finite-dimensional distributions and a is stretching factor. This definition can be interpreted in this way: the process $a^{-H}Y(at)$ is identical in distribution to the original process $Y(t)$. Typical sample path of such a process appear qualitatively the same, irrespective of the scale of the observation.

If X is a process defined as a series of independent increments of self-similar process $Y(t)$, the ACF of X can be obtained as the following:

$$r(k) = \frac{1}{2}[(k+1)^{2H} - 2k^{2H} + (k-1)^{2H}] \;\; \text{for} \;\; k \geq 0 \tag{2.4}$$

with $r(k) = r(-k)$. For $0.5 < H < 1$, $r(k)$ can be approximated by

$$r(k) \sim H(2H-1)k^{2H-2} \;\; as \;\; k \to \infty \tag{2.5}$$

From (2.5), we can observe that increments of self-similar process with $H > 0.5$ exhibit long-rang dependence property in (2.1). While $H \leq 0.5$, the process is SRD. Hence the value of Hurst parameter(H) is commonly accepted as one of the important parameters to differentiate LRD processes from SRD ones.

Consider a process $X^{(m)} = \{X_k^m : k \geq 1\}$, obtained by averaging the original increment process X over non-overlapping blocks of size m.

$$X_k^{(m)} = \frac{1}{m} \sum_{n=(k-1)m+1}^{km} X_n \tag{2.6}$$

The variance of this sample mean process decays slower than the reciprocal of sample size,

$$Var[X^{(m)}] \sim m^{-\beta} \ \ as \ \ m \to \infty, \ \ 0 < \beta < 1 \tag{2.7}$$

the parameter β is related to H by $H = 1 - \beta/2$. Equation(2.7) indicates the second property of self-similar process: slow decaying variance. It also means that the self-similar process is heavy-tailed according to heavy tail definition as the following equation:

$$Pr\{U \geq u\} \sim m^{-\alpha} \ \ as \ \ u \to \infty, \ \ \alpha > 0 \tag{2.8}$$

Fractal Point Process (FPP) Models

In general, we can classify LRD models into the following types: *fractional brownian motion*, *Fractional ARIMA*, and *fractal point process based models* [63]. However, the former two models require a high computation complexity in generating the synthetic traces. And they do not provide a plausible physical explanation of self-similarity of network traffic either. Therefore, we study and evaluate the network performance based on *fractal point process based models*, first proposed by Ryu in 1996 [9].

A point process is said to *fractal* [9] when the number of relevant statistics exhibit scaling with related scaling exponents, indicating that that represented phenomenon contains clusters of points over all (or relatively large set of time scales.

The ACF of FPP is given by:

$$r(k;T) = \frac{T^\alpha}{T^\alpha + T_0^\alpha} \frac{1}{2}[(k+1)^{\alpha+1} - 2k^{\alpha+1} + (k-1)^{\alpha+1}] \ \ for \ \ k > 0 \tag{2.9}$$

For $T \ll T_0$, $T^\alpha/(T^\alpha + T_0^\alpha)$ approaches zero and thus LRD is negligible at these scales. However, $T \gg T_0$, $T^\alpha/(T^\alpha + T_0^\alpha)$ approaches to 1 and process exhibits the LRD with $H = (1+\alpha)/2$. From equation (2.9), it also indicates that H parameter is not sufficient to describe LRD of FPP. The another parameter T_0 called *fractal onset time* is also needed.

There are two methods defined by Ryu to build FPP models, namely *Renewal Point Process* method and *Doubly Stochastic Poisson Process*(DSPP) method.

- *Renewal Point Process*:

  The interarrival time of renewal processes can follow any general distribution. If the pdf decays as the power law:

$$p(t) = \begin{cases} \gamma A^{-1} e^{-\gamma t/A} & 0 \le t \le A \\ \gamma e^{-\gamma} A^{\gamma} t^{-(\gamma+1)} & t > A \end{cases} \tag{2.10}$$

  where A is a cut-off parameter, γ is a fractal exponent, with $1 < \gamma < 2$. And such process is called *Fractal renewal process*(FRP).

  The sup-FRP process consists of the superposition of M i.i.d FRP, which can obtain more flexible value of H, λ, T_0 due to the addition of M to the traffic modeling equation.

- *Doubly Stochastic Poisson Process* (DSPP):

  DSPP yields a variety of Fractal-Modulated Poisson Point Processes (FMPPs). In this proposal, we will focus on on-off FMPP [4] models, that are implemented for generating synthetic self-similar traces.

On-Off FMPP

The on-off FMPP model is composed of M i.i.d on-off processes, where both on and off periods are i.i.d and at least one of these two states has power law distribution for their average holding time with infinite variance. Through the different combinations of distributions for on-off process, ON-OFF FMPP can be classified into three cases: PowON-PowOFF, ExpON-PowOFF, PowOFF-ExpON. We have implemented all three cases in C codes. To avoid the repetition, only PowON-PowOFF case is discussed as example.

In this model, both on and off processes are applied, with a power law distribution. Assume R as the rate of single source. H, λ, T_0 are given by the following:

$$\begin{aligned} H &= (3-\gamma)/2 \\ \lambda &= M\gamma[1+(\gamma-1)^{-1}e^{-\gamma}]^{-1}A^{-1} \\ T_0^{\alpha} &= 1/2\gamma^{-2}e^{-\gamma}(\gamma-1)^{-1}(2-\gamma)(3-\gamma)[1+(\gamma-1)e^{\gamma}]^2 A^{\alpha} \end{aligned}$$

From the trace files generated from C code, Figure 2.2 is drawn by MATLAB to illustrate the burstiness in the different time scales for PowON-PowOFF model, that gives the pictorial validation against poisson process in Figure 2.1.

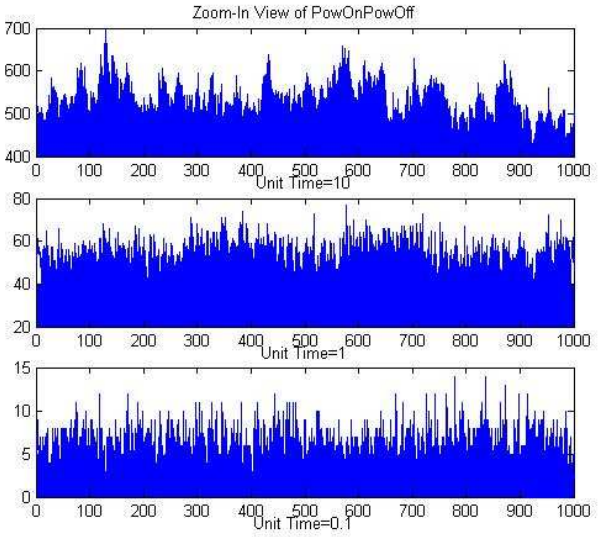


Figure 2.2: Self-Similar Traces generated from PowONPowOFF

Finally, another reason to implement on-off FMPP is the classical conclusion made by Willinger *et al.* on the LRD models [57], stated as the following:

> Many ON/OFF sources whose ON and OFF periods exhibit the *Noah Effect* (heavy-tailed with infinite variance) produce aggregate traffic which features from the *Joseph Effect* (LRD).

2.2 Quality of Service (QoS)

QoS is the abbreviation of Quality of Service. It can be defined as both the performance of a network relative to application needs and the set of technologies that enable a network to make performance assurances [29].

Specifically, QoS can be described by the following metrics [62]:

- Service Availability: The reliability of users' connection to the internet device.
- Delay: The time taken by a packet to travel through the network from one end to another.
- Delay Jitter: The variation in the delay encountered by similar packets following the same route through the network.
- Throughput: The rate at which packets go through the network.
- Packet loss rate: The rate at which packets are dropped, get lost or become corrupted.

For better QoS, the objective is to maximize the service availability and throughput, minimize the delay and eliminate the delay jitter and packet loss rate.

For the rest of the section, we will briefly introduce the control and data plane of QoS. And QoS schemes in IP wired networks are reviewed at the end.

2.2.1 Control and Data Plane of QoS

The QoS in IP network can be correspond to two "planes", namely the control plane and data plane as shown in Figure 2.3.

Control Plane—Global View of QoS

The functionalities of the control plane are([28], [61]):

- Signaling : The information or message exchanged related to the establishment and control of a connection and the management of the network for guaranteed QoS , like RSVP.
- Admission Control : The decision process of whether to accept the new flow of traffic or not according to given network resource and QoS requirement.

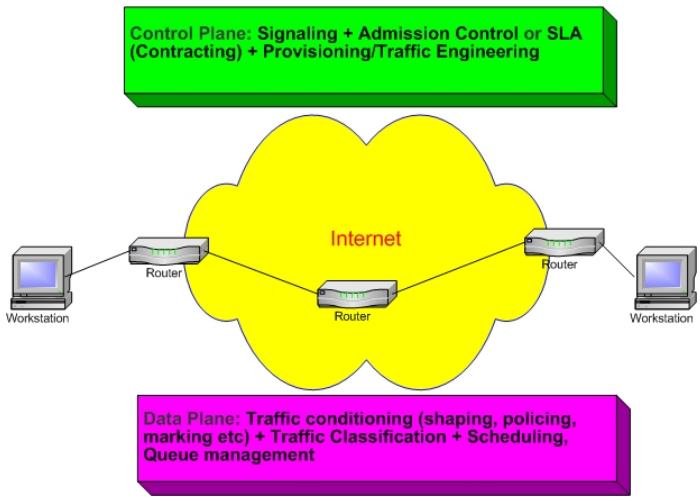


Figure 2.3: QoS Big Picture

- Service Level Agreement(SLA) : A service contract between the customer and the service provider that specifies the forwarding service the customer should receive.
- Traffic Engineering :The process of arranging how traffic flows through the network so that congestion caused by uneven network utilization can be avoided. QoS routing belongs to this category.

Above all, the control plane of QoS works to ensure resource utilization and allocation, to balance the load along the traffic path, and to manage the traffic aggregation and path flexibly.

Data Plane—Local View of QoS

The functionalities of the data plane are ([28], [61]):

- Shaping : Delay the traffic so that it would conform to the predefined rate.
- Policing : Discard some packets when the incoming traffic violates the predefined rule.

- Marking : Set the DS (Differentiated Service) field in the packet, in which the class of differentiated service is encoded.
- Traffic Classification : Sort the packets based on the content of the header according to predefined rules.
- Scheduling : Send the packets in the order of the certain service rules.
- Queue Management : Control the length of queue by dropping the packets based on the defined rules.

In addition, the packet encapsulation and de-encapsulation, and the packet fragmentation and reassembly are also functionalities in the data plane, but without QoS features. Given the incoming traffic, the data plane of QoS is to realize the traffic conditioning (shaping, policing, marking etc.) according to QoS requirements, and to prioritize and differentiate among all incoming different classes of traffic by scheduling and queue management. From the analysis above, it can be concluded that the data plane of QoS focuses on the local scope.

2.2.2 QoS Schemes in IP Wired Networks

In the past decade, three main QoS schemes, MPLS, IntServ and DiffServ, have been proposed by IETF. MPLS is originally designed for IP over ATM network, which is essentially the hybrid of network (Layer 3) and transport (Layer 2) structure. Explicit routing in MPLS, resemblance to end-to-end connection routing, improves the traffic engineering since more predictable performance can be achieved. MPLS focuses more on traffic engineering and backbone architecture than QoS definitions, although QoS features are considered. In this subsection, we will give a short review on IntServ and DiffServ schemes.

IntServ

IntServ is a framework to provide QoS guarantees on the per-flow basis. Prior to the connection establishment, the resource reservation is made between the sender and the receiver by Resource Reservation Protocol (RSVP) and admission control in the control plane so that the end-to-end QoS is guaranteed([68], [59], [33]).

RSVP is the hop-by-hop and two-way signaling protocol riding on the top of the routing protocols, which is intended to be applied in connectionless networks. It is also the receiver-initiated reservation signaling protocol.

However, due to the property of a per-flow basis in IntServ, it is very difficult to maintain the state of connections in the routers as the number of connections increases. Huge storage and processing overhead that involved in accordingly means the bad scalability in the fundamental design of IntServ. In addition, only two classes of service make it inflexible for the more specified and differentiated QoS requirements. In order to solve those difficulties in IntServ, DiffServ is proposed by IETF later.

DiffServ

Motivated by achieving scalability and flexibility, DiffServ adopts a per-class basis for QoS, in which the traffic from different flows can be aggregated into the same class due to the common QoS requirements at the edge of the network ([68], [59], [33]).

In DiffServ, the DS (Differentiated Service) field in the packet header and the corresponding set of Per-Hop-Behaviors(PHBs) are defined. Differentiated services are realized by different treatment on the packets based on the value in DS fields and corresponding PHBs. The different values of 6-digit Differentiated Service Codepoint (DSCP) map the different PHBs.

In general, the whole network can be divided into the different DS (Differentiated Service) domains in DiffServ, which are composed of the DS-capable edge and core routers. The packets from DS-capable host would be classified and conditioned at the edge routers of the DS domain according to the SLA, while core routers only forward the traffic based on PHB and remark PHB possibly. When the packets move

from one DS domain to another, the possible remarking might be involved following the SLA between these two domains. SLA can be static or dynamic. In static SLA scheme, no signaling protocol is needed.

The *data plane* of DiffServ, pushes the complexity to the edge of the network and keeps the simplicity for the core routers of the network [15].

In the edge router, the traffic will go through the more complicated processes, such as traffic conditioning, shaping, etc, while they are only simply forwarded by the core routers based on PHB. PHB defines the different treatment in traffic forwarding according to DS field in the packets. The specific scheduler should be implemented to process the traffic forwarding based on PHBs, that is also one of our motivations to propose an adaptive scheduling framework to guarantee QoS in the multiservice networks, including DiffServ.

Two PHBs have been defined by IETF, equivalent to the concept of service models: Expedited Forwarding(EF) PHB and Assured Forwarding(AF) PHB.

- Expedited Forwarding(EF) [16]: It is also called premium service, which provides a low loss, low latency and low jitter service to the users and guarantees the performance if the incoming traffic is under the peak rate. An example service of EF is the Virtual Leased Line service.

- Assured Forwarding(AF) [14]: It is also defined as assured service, intended for providing the different levels of forwarding assurance for IP packets. There are 4 AF classes. Within each AF class there are three drop priorities which determine the dropping of the packets while the congestion happens. An example service of AF is the so called Olympic Service, in which three service classes, bronze, silver and gold, are defined.

In *control plane* of DiffServ, a Bandwidth Broker(BB) [15] scheme is proposed to handle the admission control. BB functions as a resource manager and determines if the new request accepted, in light of current resource allocation of marked traffic and the policies inside its policy database.

2.3 Measurement Estimation

2.3.1 Overview

Network traffic measurement is very critical area in the field of network research. Hence, it drew a lot attention and research enthusiasm from network researchers in the past decade.

As discussed in Section 2.2, it is not difficult to see how crucial the accuracy of the measurement is to QoS performance. From the data plane, the monitor function block is implemented with measurement algorithms in order to provide the accurate information of the traffic to the rest of blocks in the data plane, such as shaping. From the input of measurement block, the network element (switch or router) can shape and police the traffic according to the traffic profile or QoS requirement. It can also be utilized for better scheduling and queue management.

Upon the control plane of QoS, the combination of measurement with admission control is the most studied topic compared with the other combinations which the measurement is involved with. This combination is often denoted as measurement-based admission control (MBAC). The objective of MBAC is to make the admission decision for the incoming flows, which is based on the estimation result of the specified measurement algorithms, without violating QoS requirements for existing flows as well as newly admitted ones.

Measurement is also frequently affiliated with the term "Self-Sizing" in the research papers. Self-Sizing is an architecture which handles the dynamic resource management for the different classes of traffic. Link partitioning or capacity allocation for a certain class of traffic depends on how good the measurement estimation is.

However, how to choose the appropriate time scale for measurement remains a haunting problem in dynamic resource management. If the time scale is too small, the measurement estimation is too pessimistic, which results in an over-allocation of resources; while if the time scale is too large, the measurement result will be underestimated. For very adaptive source traffic, an inappropriate time scale will lead to

the dramatic oscillation of the resource allocation, which is not good for the stability of the system. Fonseca *et al.* proposed a dynamic time scale to tame this effect [20].

The measurement estimation scheme can be divided into two parts in terms of functionality: statistics collection and mathematic algorithms. In statistics collection, the parameters or data needed for the input of mathematic algorithms are collected. The most basic data collection unit is denoted as "measurement slot", namely, τ. For statistics consideration, like mean and variance, multiple measurement slots are often grouped into a measurement window, equal to $W = T \cdot \tau$. With the parameters needed, the estimation of traffic rate is given after the calculation of certain mathematic algorithms.

The results of measurement estimation have been interpreted in two famous terms in the area of the measurement research: *equivalent capacity* and *effective bandwidth*. *Equivalent capacity* is defined as the value $C(\epsilon)$ such that the arrival rate for the traffic exceeds C with probability at most ϵ [19]. Upon *effective bandwidth*, there is no unified and formal definitions before the one Kelly proposed in [21]. However, whatever it is denoted by *effective bandwidth* or *equivalent capacity*, it is in general an estimation result with the stochastic consideration in order to meet the certain QoS requirements, e.g., delay and loss probability.

Over the past decades, a variety of measurement algorithms have been proposed and studied. And they can be classified into different categories based on the different predefined criteria as the following:

- **Granularity of Flows:** They can be categorized into two groups: *per-flow* based and *aggregate* based. Since a higher computation complexity is involved in the per-flow measurement, while the measurement for aggregate traffic is a good approach to differentiate among the different classes of traffic in order to provide the differentiated QoS.

- **Buffer Size:** The measurement algorithms can be categorized as measurement based on *bufferless* assumption and measurement based on *large buffer* assumption. In respect with the *bufferless* assumption, Gaussian approximation and Hoeffding bound methods are two representative ones [19]. And for *large buffer*

assumption, it is often associated with *large deviation theory* [41]. Thus, it can be also defined as a measurement based on large deviation theory. The representative algorithms in this category are *effective bandwidth* proposed by Kelly [21] and Courcoubetis algorithm [13]. All the details can be found in [63].

- **Traffic Characteristics:** The measurement algorithms can be regarded as the ones good for the SRD and LRD traffic [32, 44]. For the latter, the typical measurement algorithm is the one based on fBm traffic, proposed by Norros [44]. This algorithm considers the *hurst parameter*, H, to make it more friendly for LRD traffic, however, it also shows inflexibility for the online version because of the indispensable H estimation involved [3] and the related high computation complexity.

In the next subsection, we will describe the measurement algorithm based on traffic envelope, the one we normally implement in our simulation study.

2.3.2 Measurement based on Traffic Envelope (M-TE)

Measurement based on the *traffic envelope* concept was proposed by Qiu and Knightly [47]. Traffic envelope is maximum traffic rate over the specified time scale for the aggregate traffic. Whatever the aggregate traffic is LRD or SRD, this algorithm performs with robustness due to its second order statistics nature. It characterizes the traffic in different time scales: the burstiness in the small time scale and fluctuation of aggregate in large time scale.

The most basic measurement unit is the measurement slot, τ. The measurement window is adaptive, and composed of different number of measurement slots, varied from 1 to T, $W_k = k\tau(k = 1, 2, \cdots, T)$.

In certain measurement window W_k, assume $A[t, t + W_k]$ as the arrivals in the interval$[t, t + W_k]$, then $A[t, t + W_k]/W_k$ is the rate over that interval. The maximal rate for W_k over this time can be defined as $R_k = \max_t A[t, t + W_k]/W_k$.

Suppose $A_t = A[t\tau, (t + 1)\tau]$ as the arrivals in the time slot starting from t. In this way, the maximal rate over the certain measurement window with the size of $k\tau$,

for the past $T\tau$ from the current time t can be obtained by

$$R_k^1 = \frac{1}{k\tau} \max_{t-T+k \le s \le t} \sum_{u=s-k+1}^{s} A_u \quad \text{for} \quad k = 1, 2, \cdots, T. \tag{2.11}$$

This equation is introduced for the consideration of the burstiness in the small time scale.

The current envelope R_k^1 is measured and updated over every $T \cdot \tau$ measurement window, $R_k^n \leftarrow R_k^{(n-1)}$ for $k = 1, 2, \cdots, T$ and $n = 2, 3, \cdots, N$. The variance between envelopes over the past N windows can be computed by the following equation:

$$\sigma_k^2 = \frac{1}{N-1} \sum_{n=1}^{N} (R_k^n - \bar{R}_k)^2 \tag{2.12}$$

where $\bar{R}_k = \frac{1}{N} \sum_{n=1}^{N} R_k^n$ is the mean of the past N envelopes.

The effective bandwidth in the traffic envelope can be calculated at the small and large time scales according to [40].

For the large time scale, the effective bandwidth is obtained by

$$EB_{large} = \bar{R}_T + \alpha_{large} \sigma_T \tag{2.13}$$

where $\bar{R}_T$ and σ_T are the mean and deviation for past N envelopes with the measurement window size of $T \cdot \tau$. And $\alpha_l arge$ is used to specify the confidence interval. It cold be computed by the inverse of the complementary CDF of an N(0,1) Gaussian distribution, $\alpha_{large} = Q^{-1}(\frac{\epsilon \bar{R}_T}{\sigma_T})$.

Upon the small time scale, the effective is computed by

$$EB_{small} = \max_{k=1,2,\cdots,T} \frac{(\bar{R}_k + \alpha_{small}\sigma_k)k\tau}{k\tau - B/C} \tag{2.14}$$

where B and C are buffer size and capacity respectively. The mean $\bar{R}_k$ and deviation σ_k is for measurement window $k \cdot \tau$.And $\alpha_{small} = Q^{-1}(\frac{\epsilon \bar{R}_k}{\sigma_k})$ is computed by using the same approach as α_{large}.

The algorithm can give the worst case effective bandwidth by choosing the maximum between the small-scale effective bandwidth and the large-scale one.

$$EB = \max\{EB_{large}, EB_{small}\} \tag{2.15}$$

2.4 Scheduling Algorithms

2.4.1 Introduction

From the discussion in Section 2.2, one of the challenges to provide the guaranteed Qos is how to handle the interaction among the packets from the different flows which aggregate into the same switch. Such interaction can affect QoS of each other's adversely if no proper scheduling control scheme. For example, one misbehaving flow is arriving in higher rate than defined, it will undermine the throughput and delay bound requirements for the other well-behaving flows due to the malicious use of resources (capacity) by the misbehaving flow.

Thus, the scheduling scheme is the one to handle all incoming packets based on the predefined service order or priority, and to forward them later. A good scheduling scheme should exhibit the following properties: *efficiency*, *scalability*, *flexibility*, *protection* and *fairness*.

Regarding the fairness property, two important fairness metrics are widely accepted. They are *service fairness index*(SFI) proposed by Golestani [23] and *worst-case fairness index* by Parkeh [45].

- Service Fairness Index (SFI): Fairness is represented by the maximum difference between the normalized services received by two connections over the interval $(t_1, t_2]$, in which both sessions are continuously backlogged. The term "backlogged" means that the queue for that session never empty during that specified interval. The equation is given by: $SFI = |W_i(t_1, t_2)/r_i - W_j(t_1, t_2)/r_j|$, where $W_i(t_1, t_2)$ and $W_j(t_1, t_2)$ denote the service received by session $iandj$ during $(t_1, t_2]$, and r_i, r_j are allocated rates for session i, j respectively. The smaller the SFI, the better the fairness.

- Worst-Case Fairness Index (WFI): WFI measures the discrepancy between the service received under the service discipline and the fluid Generalized Processor Sharing(GPS) discipline. It is defined as: $(t_2 - t_1)\rho_i - W_i(t_1, t_2) \leq \varsigma_i$, where ρ_i is service rate of corresponding fluid model and ς_i is a constant. The minimum

value of ς_i is called WFI.

Among the various scheduling algorithms, First-Come-First-Serve (FCFS) is the simplest. However, this simple algorithm is only suitable for best effort traffic. Because it can not differentiate between the different connections, any burst in the low priority connections can result in the large delay for the high priority connections.

Priority scheduling is proposed to differentiate the various flows based on priority levels. Under this service discipline, the packets in the higher priority level will be always served before the ones in the lower priority level. But this simple differentiation results in the starvation of the lower priority level. Hence, neither FCFS nor priority scheduling can handle guaranteed and differentiated service in the network. That is also the reason why more complicated and flexible scheduling algorithms are needed.

In general, the scheduling algorithms can be classified into two big categories: working-conserving and non-work-conserving. A work-conserving scheduler is never idle when there is the packet in the queue, while a non-work-conserving scheduler can be idle even if there are packets in the queue waiting for the service. In the next subsection, we will give the general discussion on the work conserving algorithms, while emphasizing on *weighted round robin* and *weighted fair queueing*.

2.4.2 Work-Conserving Scheduling Algorithms

The work conserving algorithms can be divided into two types: *round robin* or *framed-based* and *service deadline* or *sorted priority*-based.

- **Round Robin:** The service discipline serves the queue in round robin fashion using frames or cycles. In each cycle, the every queue is given the chance to transmit. Different throughput guarantee is achieved by different weights associated with each queue. One of representative scheduling algorithms with *round robin* style is *weighted round robin* (WRR). Also, *deficit round robin* (DRR) was proposed in order to provide the better treatment for variable sizes of packets [52].

- **Service Deadline:** The service discipline calculates the service deadline, F_i^j, for each packet of connection j at the instant i, and serves them in the increasing order of the deadline. If there is a tie, the packets will be re-ordered randomly. The general calculation function for service deadline is as the following:

$$F_i^j = \max\{F_{i-1}^j, v_i^j\} + 1/r_j$$

where r_j is the minimum service rate of connection j , and v_i^j is the virtual time for connection j at the instant i. *Virtual clock* [70], *weighted fair queueing* , worst-case weighted fair queueing [5], *self-clocked fair queueing* [23], *delay earliest due date* [18], are all in this category. The difference between them is the way to calculate the service deadline.

Next we discuss *weighted round robin* and *weighted fair queueing*, two algorithms chosen for our proposed adaptive and optimal scheduling framework.

Weighted Round Robin (WRR)

The integer value, called weight, is allocated to each queue or connection at the initialization. In each service cycle, each queue is given the proportional slots based on their weights and served in turn if it is nonempty. There are weight counters for each queue, which will decrease by 1 after that queue is served. If the weight counter or queue length in all queues are 0, that means one service cycle is done and all the weight counters are reset to their weight values.

Under the WRR scheme, the ith queue can obtain its share of bandwidth equivalent to, $w_i / \sum_{j \in B(t)} w_j$, where $B(t)$ is the set of the backlogged queues at time t. When the weights of all queues are equal, that corresponds to the well known *Round Robin* scheduling, in which each queue receives an equal share of the bandwidth if they are all backlogged. Since it is round robin style scheduling, the computation complexity is just $O(1)$.

However, there are two inherent drawbacks in this scheduling algorithm. First, although it performs very well in dealing with a small number of connections or

queues, it will result in the larger service cycle as the the number of queues increases, which in turn causes a longer delay in each queue.

Second, it is difficult to handle the packets with variable size. One proposed solution is the weight normalization through dividing the original weight of the queue by its average packet size. However, the average packet size is often unpredictable at advance. For the better solution, that gives the rise to *deficit round robin* (DRR).

Weighted Fair Queueing(WFQ)

Fluid fair queueing(FFQ), or equivalent generalized processor sharing (GPS),is ideal scheduling policy. First, since N connections shares the same link, FFQ assumes that N connections can be served simultaneously. And the service rate for ith nonempty queue can be obtained from the equation: $\frac{\phi_i}{\sum_{j \in B(\tau)}} C$, where ϕ_i is the positive real number assigned to ith queue and $B(\tau)$ is the set of backlogged queues at time τ. However, it is not feasible in a real implementation. Second, FFQ is based on infinitesimal assumption of traffic, which is not realistic either.

Based on FFQ and GPS, separate but parallel research was performed to approximate these two ideal scheduling disciplines. The research result turns out to be the same, or equivalent, scheduling discipline. The successor of FFQ developed by Demers, Keshav, and Shenker [2] is called weighted fair queueing(WFQ), while the successor of GPS proposed by Parekh and Gallager [45] is named packet-based generalized packet sharing (PGPS).

The virtual time of WFQ is equivalent to the round number calculated from bit-by-bit round robin scheduler. If the packet arrives in an inactive queue, the virtual finish time is the sum of the recomputed virtual time (round number) and service time for this packet. If the packet arrives in an active queue, the virtual finish time is the sum of last packet finish time and service time for this packet. It can be written in the following equation:

$$F_i^k \leftarrow \max\{V(a_i^k), F_i^{k-1}\} + L_i^k/\phi_i$$

where $V(a_i^k)$ is the virtual time of kth packet of connection i, L_i^k is length of kth

packet of connection i and ϕ_i is the service rate allocated to connection i.

A connection is defined as active in WFQ if the last packet served from it, or in its queue, there is a packet with the finish number greater than the current round number.

Since WFQ needs to keep track of active connections to calculate the virtual time, the computation complexity of this algorithm is up to $O(N)$. It is also proved that the service provided by WFQ is no later than FFQ by at most one packet size.

2.5 Related Work

In this section, we will briefly review the related work, that inspires us to propose the adaptive and optimal scheduling framework in multi-service networks.

Significant contributions have been made already in related areas such as measurement-based admission control (MBAC), self-sizing network frameworks and QoS adaptive routing.

Traffic measurement and estimation has been widely studied in the past decade due to the importance of its accuracy and efficiency on QoS performance, as we discussed in Section 2.3. Again, *effective bandwidth* (EB) [21] is a well-known concept that aims to allocate an efficient amount of bandwidth in order to satisfy QoS requirements of the incoming traffic.

Approaches to QoS can be differentiated by the combination of traffic measurement with different components under control and data plane of QoS [61]. MBAC [32, 34, 47, 50] is one of the first approaches to utilize the traffic measurement and estimation in order to make admission decisions based on QoS requirements.

In regard to traffic engineering, a QoS routing mechanism based on global control was proposed in [10, 71]. Alternatively, self-sizing frameworks in which online measurements are utilized for optimization and QoS routing (under global control) have been proposed in [25,69]. Recently, due to the large overhead involved in signaling for global control, Z. Zhang *et al.* proposed QoS routing based on local control [43]. And the counterpart of self-sizing framework based on local control is studied by Nalatwad

and Devetsikiotis in [42].

In QoS scheduling, Shin *et al.* have proposed the adaptive allocation of weights according to the average queue length of the premium service, in which only QoS constraints of premium service are considered [58]. Most recently, Chandra *et al* [12] describe a dynamic resource allocation technique that uses on-line measurements. However, other than [12], there have been limited advances in formally defined, control-theoretic closed-loop methodologies. Thus, we are motivated to propose the more generalized, flexible and optimal-control oriented scheduling framework, with more complete QoS and pricing considerations.

Chapter 3

MBORA System: Problem Formulation

In Chapter 1, we clarified our motivation and objective, that is to design a generalized adaptive scheduling framework in multi-service networks that meets QoS requirements while making optimal decisions through dynamic resource allocation.

Before we start the discussion on our proposed generalized framework, we want to further clarify the following two issues.

First, our proposed generalized framework is assumed to run on single basic network elements, such as a router, switch, etc. Unlike the global optimization and adaptation that traffic engineering characterizes, it originates from the the local point of view, emphasizing the traffic delivery on a single node, for example an access point or concentrator (e.g., DSLAM in cable systems).

Second, our optimization model for the decision policy is formulated from the perspective of the network provider, therefore it is the revenue or profit, instead of the incurred cost for the user side, that is emphasized in our proposed framework.

Next, we will introduce our proposed generalized adaptation and optimization framework over measurement-based scheduling.

3.1 Measurement-Based Optimal Resource Allocation (MBORA) System

In this section, we will introduce our proposed generalized framework for a network node that can accommodate multiple classes (see Figure 6.1). Suppose m deterministic delay-bounded classes and n flexible delay-bounded classes share the link capacity, C. Due to the rigidity of the deterministic delay-bound classes, such service level requirements can only be guaranteed by the appropriate traffic shaping and admission control schemes ([22, 34, 45]), together with a certain amount of reserved bandwidth, C_r. Hence, these classes are excluded from our proposed Measurement Based Optimal Resource Allocation (MBORA) system.

A MBORA system is responsible of optimally allocating the excess bandwidth, $C' = C - C_r$, shared by the n flexible delay-bound classes. The proper allocation between the reserved bandwidth and the excess bandwidth can be achieved by a generic scheduler, such as Hierarchical Packet Fair Queueing mechanism proposed by [6].

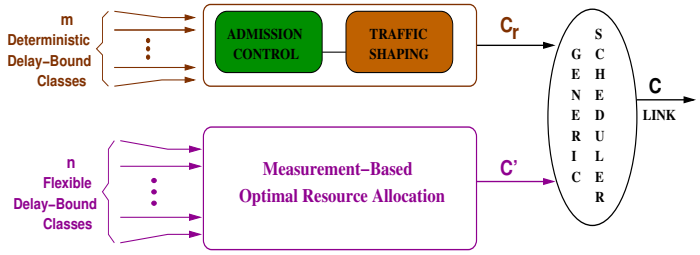


Figure 3.1: General proposed framework to deliver service guarantees for flexible multi-class networks.

Next, we focus on the main components and related coordinations of a MBORA system shown in Figure 3.2. Its main components are: a *traffic measurement module* that provides an accurate estimation of the future traffic load of the different classes under consideration over a pre-specified time interval (window), a *decision module* that determines how bandwidth is distributed among the various classes of traffic based on the information acquired from *traffic measurement module* and a *scheduling*

module that deals with the packet forwarding mechanism.

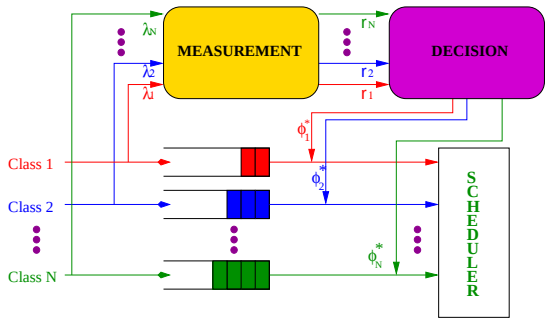


Figure 3.2: Illustration of the adaptive framework under consideration and its components.

The coordination of these three components is described next: when a job/customer of class i arrives at the scheduler, it is assigned the queue of the corresponding queue, waiting to receive service from the scheduling module. At the same time, the measurement module updates the arrival rate statistics of the corresponding traffic class, and provides an estimate of the arrival rate. It should be noted that the measurement module performs the above operation over a pre-specified time interval (window). Next, the decision module allocates the service rate (bandwidth) to the queues using information about the estimated arrival rate and the queue length processes. For the decision module, we focus on the problem that module solves at every decision time instant (at the end of a W-duration window). Regarding the scheduling module, it updates the weights for the different classes of the generalized scheduler (WRR or WFQ), in order to accommodate the service rates determined by the decision module.

In general, we can describe our proposed framework from two respects: the *adaptiveness* and *optimum*. The *adaptiveness* respect is more focused on the functionality and coordination of the components in the proposed framework, while the optimum respect is more associated with what the optimal decision modelings are formulated and how to reach the optimal solution in the decision module, given the various pricing models and QoS considerations.

Chapter 4

MBORA System under Simple Threshold Policy

4.1 Introduction

Regardless of the specific QoS delivery mechanism, present or future, there are great efficiency and robustness advantages to be gained from enhanced, measurement-based algorithms for adaptation of scheduler settings. Our adaptive technique for generalized schedulers and its analysis presented here, apply equally well to any of these QoS mechanisms. In this chapter, we introduce our first investigation on the adaptiveness of the proposed framework, where the simple threshold decision policy is applied without considering its optimum respect.

In this first and simple approach, we examine the performance extensively via simulation in order to validate the advantage of the adaptiveness of the proposed framework. Another objective is to evaluate the capability of the proposed framework to maximize the fairness of resource allocation among all classes of traffic, while satisfying their own QoS requirements.

In this chapter, we will discuss the preliminaries for our proposed framework first. Next, we introduces the kernel of this framework, the threshold policy purely based on the measurement estimation. Part of performance evaluation study of the proposed approach will be presented in Section 4.4 and the rest can be found in Appendix A. Finally, some concluding remarks are drawn.

4.2 Preliminaries

Without loss of generality, we assume that our proposed scheme operates under the DiffServ mechanism, where the various traffic flows can be classified as: Expedited Forwarding (EF) service, Assured Forwarding (AF) service and Best Effort (BE) service. At an appropriate level of abstraction, these three service classes can be thought of as delay-sensitive, loss-sensitive and best effort, respectively, a characterization that we adopt for the remainder of this chapter.

We focus on WFQ and WRR-like scheduling algorithms, which are well discussed in Section 2.4. Moreover, the definition of WRR shows that the class weights correspond to the number of slots to be served in a single scheduling cycle. Hence, for both WRR and WFQ, the bandwidth allocated to class i at decision time τ is given by $\phi_i = w_i C / \sum_{j \in N(\tau)} w_j$, where w_i is the weight allocated to the i-th class and $N(\tau)$ is the number of backlogged queues at time instant τ. With respect to the measurement algorithm, we adopt the traffic envelope algorithm proposed by Qiu and Knightly in Section 2.3.2 [47].

4.3 Simple Threshold Policy (θ_1, θ_2)

Notice that transient phenomena, such as traffic burstiness, coupled with imbalances in the traffic load over classes, would result in performance degradations, unless the class capacity allocations are accordingly adjusted. A simple and intuitive rule that satisfies the QoS requirements is that at every point in time we have $\phi_i C' \geq EB_i$,

where ϕ_i is the share of the bandwidth of traffic class i and EB_i the estimated effective bandwidth for the i-th class. In practice, we require that the above relation holds for a time interval of length W, appropriately chosen (for more on the choice of W see Appendix A).

Nevertheless, in many instances, the sum of the estimated arrival rates could be greater than C', which corresponds to the under-provisioning case in this dissertation. In such instances, the QoS requirements of the delay-sensitive class should be satisfied at the expense of the remaining two classes, provided that it does not exceed a prespecified threshold θ_1. Furthermore, if the under-provisioning is due to the temporal traffic pattern of the best effort class, then the QoS requirement of the loss sensitive class should also be satisfied, provided that it does not exceed a different prespecified threshold θ_2. Summarizing the above requirements, the queue weights are determined as follows:

1. Obtain the effective bandwidth, EB_i for the i-th class from the measurement module.

2. If the sum of the three effective bandwidths is less than the link capacity (i.e., $\sum EB_i < C'$), then each class' share of bandwidth is given by $\phi_i C' \geq EB_i$; otherwise, go to step 3.

3. If $\sum EB_i > C'$, then provide bandwidth to the delay-sensitive class up to $\phi_1 C' \leq \theta_1 C'$, to the loss-sensitive class up to $\phi_2 \leq \theta_2 (EB_2/C')$ and the remainder to the best effort class.

4. If the scheduling module is based on WRR, then normalize the share of bandwidth for each class in order to get the integer weights allocated to them. If it is based on WFQ, then the share of bandwidth for each class corresponds to the weights.

Finally, notice that the proposed approach also decreases the computational complexity of the resulting DWFQ scheme, since it is determined by the number of classes, as opposed to the number of connections.

4.4 Performance Evaluation

4.4.1 Simulation Scenarios

The model used in our experimental investigations is based on the architecture shown in Figure 4.1. The network element is a processor sharing system with three input queues that map to the delay-sensitive, the loss-sensitive and best effort class, respectively. Each queue is associated with a certain weight whose value is controlled

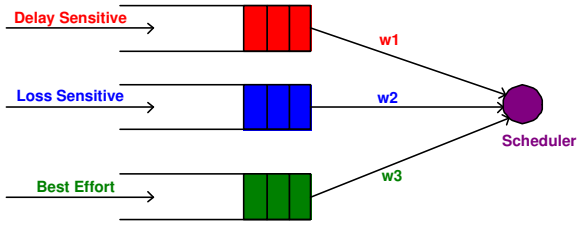


Figure 4.1: The architecture of the simulation model.

by the scheduling module. Our proposed scheduling scheme changes the weights *dynamically* over the adaptive time window, after their values are initially set at $w_1 = 2, w_2 = 2$, and $w_3 = 1$, reflecting the importance of the respective classes, for all scheduling algorithms in our simulation study. The excess bandwidth is 64kbps and the buffer sizes for the three classes are set to 500 (delay-sensitive), 5000 (loss-sensitive), and 10000 (best effort) bytes, respectively.

The effective bandwidth is recalculated for every adaptive window $N \cdot T \cdot \tau$, with $\tau = 0.01$, $T = 10$, and $N = 10$. The queue management mechanism used corresponds to Drop-Tail.

The traffic generation model follows the on-off Fractal Modulated Poisson process (FMPP) proposed in [4]. The FMPP produces long-range dependent traffic, by using a power-law distribution for the "on" or "off" periods, while the packet distribution Over the "on" is Poisson. The traffic rate varies over time due to the changing number of flows in the "on" state. Furthermore, in our setting additional variability is introduced through changes in the composition of traffic over the three classes. For

the sake of simplicity the packet size was fixed to 50 bytes and the Hurst parameter of the traffic process to 0.7.

Given the excess bandwidth, the system's available capacity (stability) constraint is at 6.25 msec. That is, if the mean packet inter-arrival time is larger than this value, the system is over-provisioned and relatively few packet losses are expected to occur. On the other hand, for mean packet inter-arrival times smaller than 6.25 msec, the system is unstable and highly backlogged queues together with frequent packet losses are expected. The case where the mean time is exactly 6.25 corresponds to the critical regime.

Finally, the policies to be compared in the simulation study are the Static Weighted Round Robin (SWRR), the Static Weighted Fair Queue (SWFQ) and their dynamic counterparts DWRR and DWFQ. Their main difference is that for the static policies the set of weights do not change over time. The performance metrics used in our study are packet loss probabilities and average queue delays.

4.4.2 Results and Analysis

We start by providing a picture of the traffic patterns used in our study. Figure 4.2 shows how the total traffic rate and that of the three classes changes over time. Every 100 seconds, the traffic pattern of each class changes, which induces a transient bursty behavior for the traffic of each class. And Figures 4.3–4.5 demonstrate the traffic rate and allocated capacity by different scheduling policy schemes in under-provisioned case. We assume that the queues are always backlogged over each adaptive window.

The plots indicate that, in our simulation scenarios, the delay and loss sensitive classes should be adequately provisioned under the proposed QoS scheme. We turn our attention next to evaluating the performance of our scheme. In order to be able to examine a large range of performance the mean inter-arrival packet time is varied from 5 ms to 9.1 ms, thus covering both the under and the over-provisioned cases. In the following figures the class loss probabilities and average queue delays are plotted as functions of the input rate for the two dynamic and the two static policies.

Figures 4.6 (a) and 4.6 (b) show the substantial decrease in loss and delay obtained

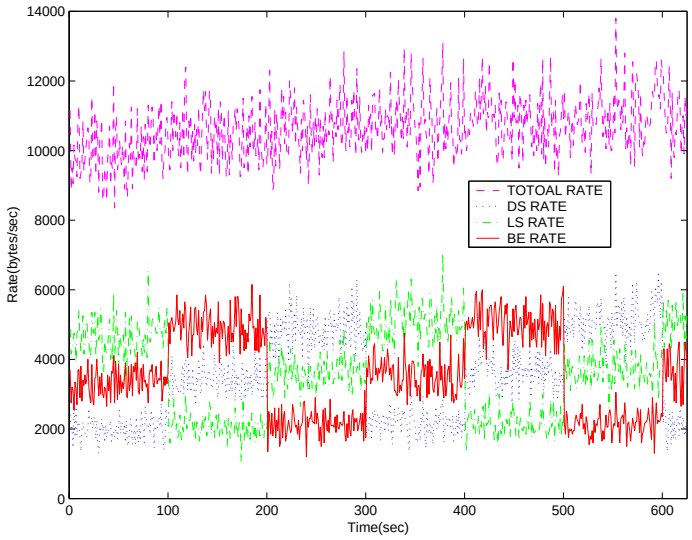


Figure 4.2: Pictorial view of total traffic rate vs. traffic rate for each class in the under-provisioned case.

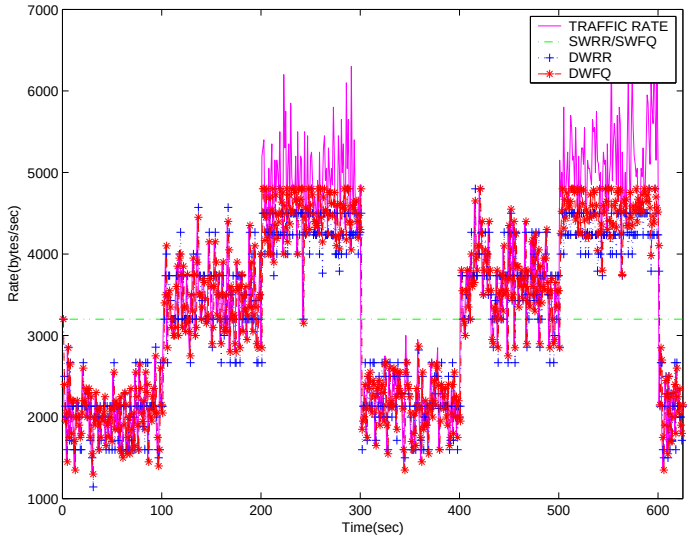


Figure 4.3: Pictorial view of traffic rate vs. resource allocated by different scheduling algorithms for delay-sensitive class of traffic in the under-provisioned case.

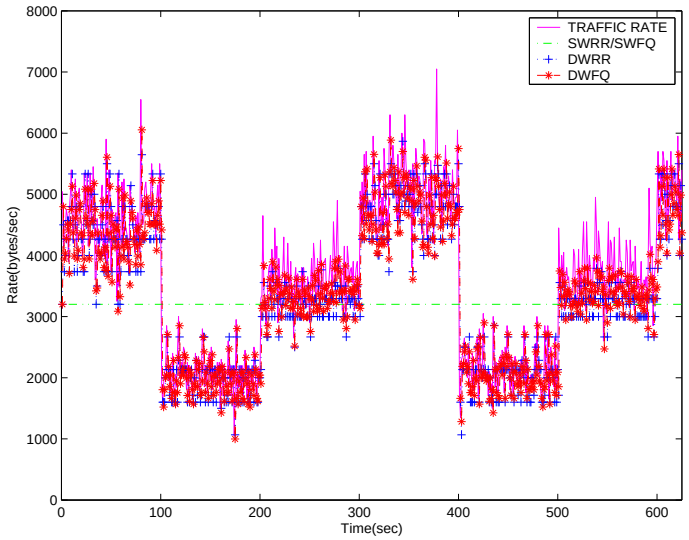


Figure 4.4: Pictorial view of the traffic rate vs. resource allocated by different scheduling algorithms for the loss-sensitive class of traffic in the under-provisioned case.

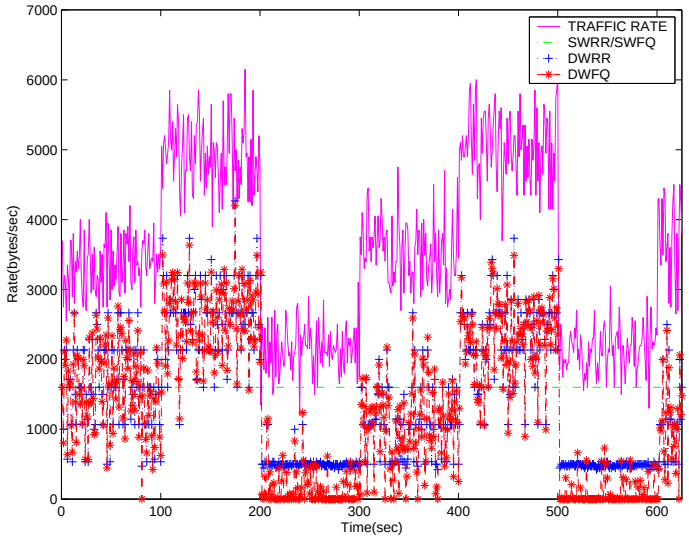


Figure 4.5: Pictorial view of the traffic rate vs. resource allocated by different scheduling algorithms for best effort class of traffic in the under-provisioned case.

from the DWFQ policy for the delay sensitive class in the under-provisioned scenario. This policy also achieves the best performance in the well provisioned case, but the gains are smaller in the over-provisioned case because the bandwidth will be automatically redistributed among the rest of the nonempty queues whenever any queue becomes empty due to the work-conserving nature of the static scheduling policy. It is also worth noting that the DWRR policy outperforms both static policies under the loss performance metric for the same class, although this comes at the expense of larger delays. The latter is a consequence of the integer weight normalization for DWRR.

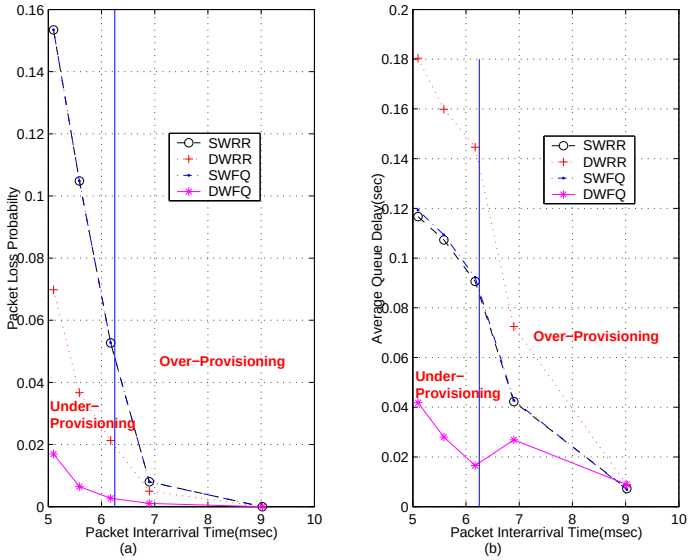


Figure 4.6: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for delay-sensitive class.

Analogous conclusions can be reached for the loss-sensitive class by examining Figures 4.7 (a) and 4.7 (b). However, since queueing delays are not that important for this class, it can be concluded that DWRR outperforms the static policies, as well.

Finally, the dynamic policies prove competitive with respect to the best effort class, as Figures 4.8 (a) and 4.8 (b) illustrate for the over-provisioned scenario, while they exhibit an up to 50% degradation in performance for the under-provisioned case. This result is expected, given the fact that this class receives the lowest priority under the proposed scheme.

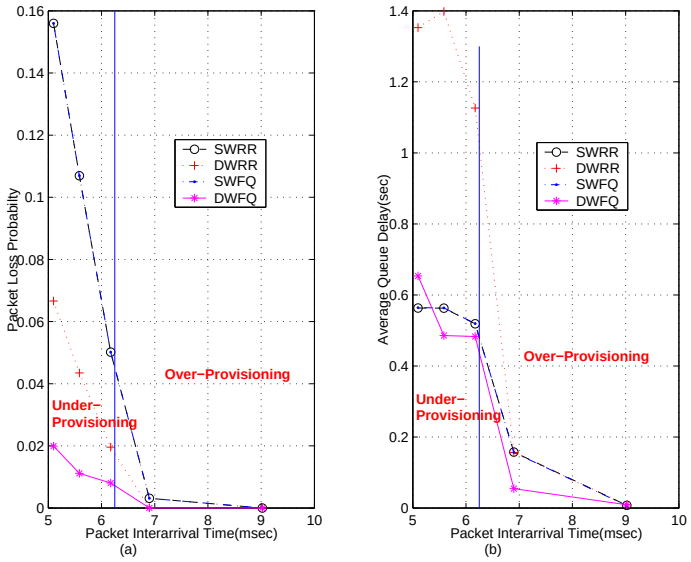


Figure 4.7: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for loss-sensitive class.

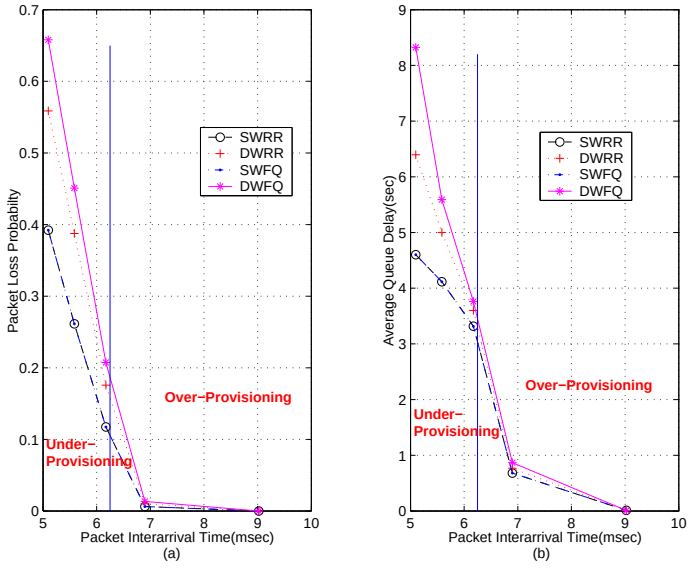


Figure 4.8: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for best effort class.

The main conclusions are: (i) that the dynamic scheduling algorithms perform as well as their static counterparts for all classes in the over-provisioned scenario and (ii) they significantly *outperform* the static policies for the delay and loss sensitive classes under the remaining scenarios.

We also investigate the performance of our proposed scheme by tuning the following parameters: the threshold $(\theta 1, \theta 2)$, pattern change interval (PCI), the unbalanced factor (σ), and the adaptive window (W). The details are shown in Appendix A, by which our conclusion is further validated.

4.5 Conclusions

In this chapter our generalized framework based on the threshold policy, is introduced to provides QoS guarantees for different traffic classes under the DiffServ mechanism. The proposed scheme, due to its *dynamic and adaptive* nature exhibits a superior performance to static algorithms. Two dynamic algorithms DWFQ and DWRR have been investigated and the former outperforms the latter in almost all scenarios examined, due to its inherent flexibility on determining the weights. Our proposed scheme proves robust to the underlying load distribution of the traffic classes and to the changing nature of traffic characteristics over time.

The proposed scheme relies on two, externally set parameters (e.g. by the ISPs) (θ_1, θ_2), that determine the relative importance of the traffic classes. It also utilizes an adaptive sampling window, whose optimal determination is a topic of current work.

Chapter 5

MBORA System as a Profit Maximization Center

After the validation of the advantage of the generalized proposed framework in the adaptiveness respect, we further explore our framework from the optimality point of view in this chapter.

The objective is to formally investigate optimal settings based on profit-oriented formulation that also take into account all the relevant QoS considerations, such as loss probability and delay.

5.1 Generalized Profit-Oriented Formulation

We generally describe our optimal scheduler as a *profit center*, that is the core of the proposed framework for bandwidth allocation [56]. The provider's long-term profit consists of a revenue based on usage charge, $\sum R_i(\phi_i)$, and a cost component related to the delay congestion, $\sum \mathcal{C}'_i(\phi_i)$, for all the classes, where ϕ_i is the proportion of the excess bandwidth allocated to class i. The objective is to maximize the usage

charge while minimizing the delay penalty cost incurred by the congestion. Thus, our generalized profit-oriented model can be formulated as the following optimization problem:

$$\max_{\phi} f(\phi) = \sum_{i} U_i(\phi_i) = \sum_{i} R_i(\phi_i) - \sum_{i} \mathcal{C}'_i(\phi_i) \tag{5.1}$$

s.t.

$$\sum \phi_i \leq 1$$

Based on this formulation, the system can achieve the optimal allocation of resources in terms of financial and QoS concerns. Furthermore, the network provider also achieves greater flexibility in controlling the shared resources by means of varying the pricing models and additional service guarantees constraints in this generalized formulation.

Next, we briefly discuss how our generalized formulation is affected by the characteristics of the pricing models adopted and by the nature of service guarantee constraints.

- Pricing Models:

  The pricing model can have a linear or a non-linear functional form, which affects the sensitivity of the bandwidth allocated with respect to price changes. In the linear case, the cost component is given by $\mathcal{C}'_i(\phi_i) = b_i D_i$, whereas an example of a nonlinear pricing function is $\mathcal{C}'_i(\phi_i) = b_i e^{(D_i - d_i)} D_i$. In the latter case, the penalty cost increases exponentially when the desired delay cannot be met, compared with the linear pricing model. The details about pricing model will be discussed later in Chapter 7.

- Service Guarantees Constraints:

  In the case that the network provider wants to take an aggressive approach in satisfying QoS requirements, it can be achieved by adding more service guarantees constraints in the generalized profit-oriented formulation, such as the

following:

$$f_1(\phi_i) \leq P^i_{loss}$$
$$D_i = f_2(\phi_i) \leq d_i$$

where P^i_{loss}, D_i and d_i are the loss probability, expected queue delay and desired queue delay of class i, respectively. Furthermore, $f_1(\phi_i)$ and $f_2(\phi_i)$ represent the functions that provide the desired loss probability and delay, respectively.

Similarly, in the profit center model, the type of service level requirements used has a significant impact on the amount of allocated bandwidth. Notice that the constraint functions $f_1(\phi_i)$ and $f_2(\phi_i)$ can have either a deterministic form – e.g., average queue delay, $\bar{D}_i < d_i$ – or a stochastic one – e.g., $P\{D_i > d_i\} \leq \epsilon_i$ [53,72]. It can also be seen that the constraints we allow in our formulation can correspond to any statistic of interest of the delay (or loss) distribution, such as the mean, the median or any other quantile of interest.

Chapter 6

MBORA System under Linear Pricing Models

Given the generalized profit-oriented formulation, our first study will be focused on sub-case of special interest in this chapter, under the assumption of the generalized service models for the multi-service networks [51].

This chapter is organized as follows: We first introduce the generalized service models in Section 6.1. In Section 6.2 the optimization problem under linear pricing model and average queue delay is formulated. In Section 6.3.1 we discuss in detail the form of the solution in different cases depending on the binding constraints for the stable case, while our proposed relaxation and threshold scheme for the unstable case are presented in Section 6.3.2. Section 6.4 further derives the calculation of the optimal values for the stable case and presents a complete solution algorithm for both the stable and the unstable cases. Section 6.5 contains a sensitivity analysis of the optimization problem with respect to the prices charged to the various traffic classes, while in Section 6.6 a similar analysis is presented for choosing the adaptive window over which the scheduling decision is implemented, W. Simulation results are presented in Section 6.7. We also summarize and conclude with open issues and

future research directions in Section 6.8. Finally, for the sake of completeness, we give the definition of supermodularity [54] which is used in this dissertation and some related proofs in Appendix B.

6.1 Generalized Service Models

First, we start by introducing a generalized service model for the various applications in multi-service networks, that can also similarly apply to other QoS schemes, such as IntServ and DiffServ.

Following Shenker [51], service models can be differentiated by the related utility function, that reflects the mapping relationship between the service delivered and the performance of the application. Pure data applications are defined as an *elastic* class that can tolerate packet drops and delays and is characterized by a strictly concave and differentiable utility function. On the other hand, real-time applications can be further differentiated into three classes; *hard real-time, delay-adaptive, and rate-adaptive*. The hard real-time class needs to adhere to strict deterministic delay bounds, whereas the delay-adaptive and the rate-adaptive classes are more tolerant and require delay guarantees on the average, with the former being significantly more demanding than the latter. Mapping these classes accordingly into Figure 3.1, we study the special scenario with $m = 1$ and $n = 3$. Accordingly, the proposed framework and MBORA system under the generalized service models can be shown in Figure 6.1 and 6.2.

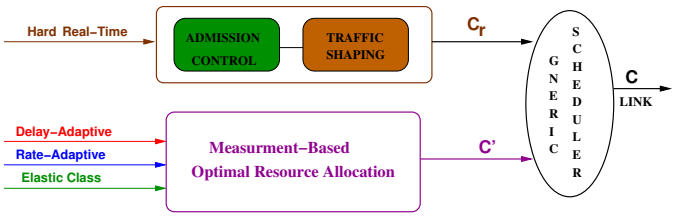


Figure 6.1: General proposed framework to deliver QoS under the generalized service models.

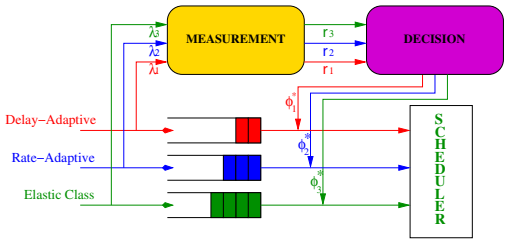


Figure 6.2: MBORA system under the generalized service models.

6.2 Problem Formulation

Applied with the linear pricing model and average queue delay consideration, the profit-oriented problem in this scenario can be expressed as

$$\max_{\phi}\{\sum R_i - \sum {C'}_i\} = \max_{\phi}\{\sum_{i=1}^{3} p_i\phi_i C' - \sum_{i=1}^{2} \frac{b_i\bar{q}_i}{\phi_i C'}\}, \tag{6.1}$$

subject to

$$\sum_{i=1}^{3}\phi_i \le 1, \quad \phi_i \le 1$$
$$\phi_i \ge \max\{\tfrac{r_i}{C'}, \tfrac{\bar{q}_i}{C'd_i}\},\ i = 1,2 \quad , \quad \phi_3 \ge \tfrac{r_3}{C'}$$

where $\bar{q}_i$ is the average queue length, p_i is the price per unit of the utilization of the system's resources for class i and b_i is the cost per unit of time incurred by class i, and with $\phi = (\phi_1, \phi_2, \phi_3)$. Since the higher priority class requests a better QoS than the lower priority classes, it should be charged with a higher price for per unit of resource utilization and the cost per unit of time due to the premium service expected to receive, over the remaining classes; therefore, $p_i > p_j$ and $b_i > b_j$ for $i < j$. Note that the delay cost for the elastic class is not considered.

In equation 6.1, r_i is a estimated generalized traffic rate for class i. It could be selected as the mean rate, peak rate, or *effective bandwidth* (of one of many possible definitions), depending on the different QoS requirements for different classes and how conservatively the capacity is allocated among them for the shared-link system. Effective bandwidth is defined by Kelly as the bandwidth required to satisfy QoS

constraints via large deviation theory [21], that is a value between the mean rate and the peak rate. It is an efficient measure to allocate bandwidth in order to make their loss probability under a given bound [24]. For those classes without strict QoS constraints, like our "elastic" class, the mean rate is adequate for bandwidth allocation purposes, whereas the effective bandwidth rate might be too conservative.

The average queue length plays an important role in the above formulation and can be derived from the fluid model given in [37]. Notice that the instantaneous queue length process at time t for class i can be obtained through the formula

$$q_i(t) = \max[q_i^0 + (r_i - \phi_i C')t, 0] \tag{6.2}$$

where q_i^0 is the initial queue length of the i-th class and t denotes the length of the time interval. The max operator prevents the process from taking negative values.

In our proposed scheme, the share of system resources (bandwidth) allocated to the various classes would be dynamically assigned over an adaptive window W. Thus, the average queue length of class i during an adaptive window W is given by

$$\begin{aligned} \bar{q}_i &= \frac{1}{W}\int_0^{\tau_i} q_i(t)dt \\ &= \frac{\tau_i}{W}[q_i^0 + \frac{\tau_i}{2}(r_i - \phi_i C')] \end{aligned} \tag{6.3}$$

where $[0, \tau_i]$ is the time interval during which the queue length process remains positive during an adaptive window W and τ_i is determined by $\tau_i = \min\{t_i^0, W\}$ with t_i^0 being the time it takes to empty the queue. In turn, t_i^0 can be obtained by $t_i^0 = q_i^0/(\phi_i C' - r_i)$, given the initial queue length q_i^0. It can be seen that the average queue length depends on τ_i and therefore we distinguish the following cases:

Condition 1: if $t_i^0 < W$, we obtain the following relationships from the QoS constraints.

$$\phi_i > \frac{q_i^0}{WC'} + \frac{r_i}{C'} = \varphi_i^1 \tag{6.4}$$

On the other hand, the average queue length $\bar{q}_i$ can be written as:

$$\bar{q}_i = \frac{[q_i^0]^2}{2W} \times \frac{1}{\phi_i C' - r_i}$$

Then, the delay component of the QoS can be rewritten as follows:

$$\phi_i \geq \frac{r_i}{2C'} + \sqrt{(\frac{r_i}{2C'})^2 + (\frac{q_i^0}{C'})^2 \times \frac{1}{2d_iW}} = \varphi_i^2 \tag{6.5}$$

Hence, equations 6.4 and equation 6.5, imply that $\phi_i \geq \max[\varphi_i^1, \varphi_i^2]$, while the constraint $\phi_i \leq 1$ has to be satisfied.

We next examine the relationship between the variables φ_i^2 and φ_i^1.

In Figure 6.3, we plot the relationship between between φ_i^2 and φ_i^1 as a function of the initial queue length of class i, q_i^0. It can be seen that

$$\max[\varphi_i^1, \varphi_i^2] = \begin{cases} \varphi_i^1 & \text{if } q_i^0 \leq \frac{r_iW}{\frac{W}{2d_i}-1} \\ \varphi_i^2, & \text{otherwise} \end{cases}$$

There are only two intersection points for the curves of φ_i^2 and φ_i^1; one at $q_i^0 = 0$ and the other one at $q_i^0 = \frac{r_iW}{\frac{W}{2d_i}-1}$. Intuitively, when q_i^0 is less than the threshold $\frac{r_iW}{\frac{W}{2d_i}-1}$, the system's first priority is to empty the queue, while the desired queue delay is met. However, when q_i^0 is larger than the above threshold value, the system needs to allocate a larger share of its bandwidth to meet the desired delay requirement rather than trying to empty the queue. As the duration of the scheduling window W increases, the value of the threshold will decrease. When $W \gg d_i$, this threshold can be approximately written as $2r_id_i$.

Condition 2: if $t_i^0 \geq W$, we analogously obtain

$$\phi_i \leq \frac{q_i^0}{WC'} + \frac{r_i}{C'} = \varphi_i^1 \tag{6.6}$$

In addition,

$$\bar{q_i} = q_i^0 + \frac{W}{2}(r_i - \phi_iC')$$

which in turn gives that

$$\phi_i \geq \frac{\frac{q_i^0}{WC'} + \frac{r_i}{2C'}}{\frac{1}{2} + \frac{d_i}{W}} = \varphi_i^d \tag{6.7}$$

However, in some cases φ_i^d may be greater than φ_i^1, which creates a conflict between the constraints. In such a case we must at least have $\phi_i \geq \frac{r_i}{C'}$, whereas if φ_i^d is no

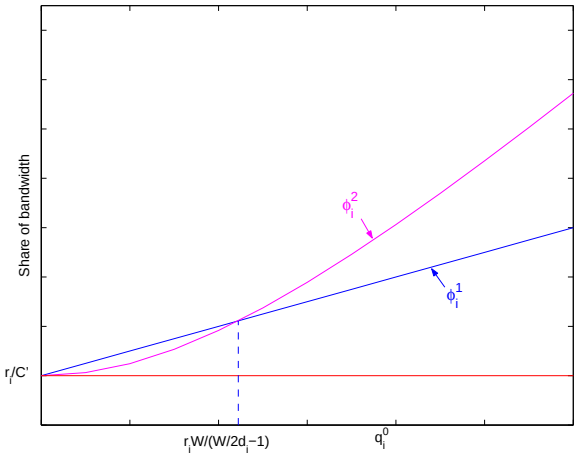


Figure 6.3: Relationship between φ_i^1 and φ_i^2.

greater than φ_i^1, then the lower bound of ϕ_i should be given by the maximum of $\frac{r_i}{C'}$ and φ_i^d. In summary we have:

$$\varphi_i^3 = \begin{cases} \frac{r_i}{C'} & \text{if } \varphi_i^d > \varphi_i^1 \\ \max[\frac{r_i}{C'}, \varphi_i^d] & \text{if } \varphi_i^d \leq \varphi_i^1 \end{cases}$$

Hence, under condition 2 we get the following constraint $\varphi_i^3 \leq \phi_i \leq \varphi_i^1$.

We explore next the relationship between φ_i^d and φ_i^1.

In Figure 6.4, we plot the relationship between between φ_i^d and φ_i^1 as a function of q_i^0. The threshold value of q_i^0 that makes $\varphi_i^d = \varphi_i^1$ is exactly the same as the one we discussed in Figure 6.3. Moreover, there is also only intersection point between the two curves. From Figure 6.4, when $\varphi_i^d > \varphi_i^1$, the delay constraint can not be met due to the conflict with the assumption that $\phi_i \leq \varphi_i^1$ for Condition 2. Thus, we must at least have $\phi_i \geq \frac{r_i}{C'}$. For $\max[\frac{r_i}{C'}, \varphi_i^d]$, we finally obtain the following equation from Figure 6.4:

$$\max[\frac{r_i}{C'}, \varphi_i^d] = \begin{cases} \frac{r_i}{C'} & \text{if } q_i^0 \leq \frac{r_i d_i}{WC'} \\ \varphi_i^d, & \text{otherwise} \end{cases}$$

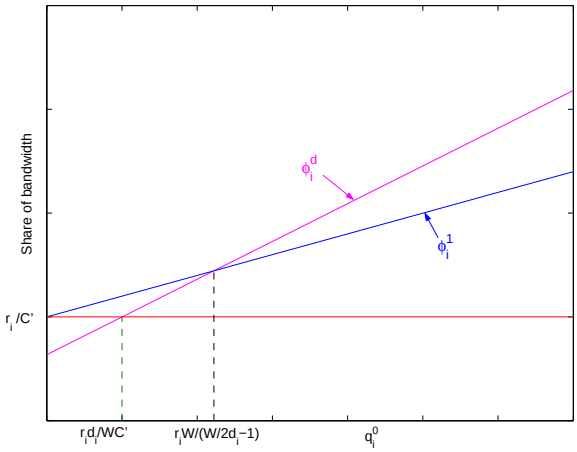


Figure 6.4: Relationship between φ_i^1 and φ_i^d.

6.3 Optimal Allocation of Resources

6.3.1 The Over-Provisioned Case

Equation 6.1 shows that the problem under consideration is a nonlinear optimization one with inequality constraints. In the ensuing discussion it is assumed that the system is *over-provisioned*, in the sense that the sum of the expected input rates does not exceed the excess bandwidth of the router; i.e. $\sum_{i=1}^{3} r_i < C'$.

In this subsection we outline how the optimal solution is obtained. It is important to note that the constraints change the nature of the objective function over different regions of the parameter space. We start by considering the nature of the constraints.

Notice that at the optimum we must have $\sum_{i=1}^{3} \phi_i = 1$; otherwise, resources would be wasted. Furthermore, since the best effort class (3rd class) pays the lowest price and has no constraints on its delay, we get that at the optimum $\phi_3 = \frac{r_3}{C'}$. These two facts show that the optimal solution satisfies

$$\phi_1 + \phi_2 = 1 - \frac{r_3}{C'}.$$

Furthermore, the problem has been reduced to one involving only two decision variables, namely ϕ_1 and ϕ_2.

The relationship between the duration of the measurement window W and the duration of emptying the queue τ_i imposes QoS constraints, that in turn translate to the following four cases for our optimization problem:

1. Case 1: Suppose that both decision variables (i.e. ϕ_1 and ϕ_2) satisfy condition 1. The optimization problem can then be written as:

$$\begin{aligned}\max_{\phi_1,\phi_2} f(\phi_1,\phi_2) =& p_1\phi_1 C' + p_2\phi_2 C' \\ &- \frac{b_1 \times [q_1^0]^2}{2W(\phi_1 C' - r_1)\phi_1 C'} \\ &- \frac{b_2 \times [q_2^0]^2}{2W(\phi_2 C' - r_2)\phi_2 C'}\end{aligned}$$

   subject to the constraints

$$\begin{aligned}&\textstyle\sum_{i=1}^{2} \phi_i = 1 - \frac{r_3}{C'} \\ &\phi_1 \geq \max[\varphi_1^1, \varphi_1^2] \\ &\phi_2 \geq \max[\varphi_2^1, \varphi_2^2]\end{aligned}$$

2. Case 2: Suppose that ϕ_1 satisfies condition 1, whereas ϕ_2 satisfies condition 2. In this case the optimization problem can be written as:

$$\begin{aligned}\max_{\phi_1,\phi_2} f(\phi_1,\phi_2) =& p_1\phi_1 C' + p_2\phi_2 C' \\ &- \frac{b_1 \times [q_1^0]^2}{2W(\phi_1 C' - r_1)\phi_1 C'} \\ &- b_2 \times (\frac{q_2^0 + \frac{W}{2} r_2}{\phi_2 C'} - \frac{W}{2})\end{aligned}$$

   subject to the constraints

$$\begin{aligned}&\textstyle\sum_{i=1}^{2} \phi_i = 1 - \frac{r_3}{C'} \\ &\phi_1 \geq \max[\varphi_1^1, \varphi_1^2] \\ &\varphi_2^3 \leq \phi_2 \leq \varphi_2^1\end{aligned}$$

3. Case 3: Suppose that ϕ_1 satisfies condition 2, whereas ϕ_2 satisfies condition 1. Thus, the optimization problem can be written as:

$$\begin{aligned}\max_{\phi_1,\phi_2} f(\phi_1,\phi_2) =& p_1\phi_1 C' + p_2\phi_2 C' \\ & - b_1 \times (\frac{q_1^0 + \frac{W}{2} r_1}{\phi_1 C'} - \frac{W}{2}) \\ & - \frac{b_2 \times [q_2^0]^2}{2W(\phi_2 C' - r_2)\phi_2 C'}\end{aligned}$$

subject to the constraints

$$\begin{aligned}\textstyle\sum_{i=1}^{2} \phi_i &= 1 - \frac{r_3}{C'} \\ \varphi_1^3 \leq \phi_1 &\leq \varphi_1^1 \\ \phi_2 &\geq \max[\varphi_2^1, \varphi_2^2]\end{aligned}$$

4. Case 4: Suppose that both decision variables satisfy condition 2. The optimization problem becomes:

$$\begin{aligned}\max_{\phi_1,\phi_2} f(\phi_1,\phi_2) =& p_1\phi_1 C' + p_2\phi_2 C' \\ & - b_1 \times (\frac{q_1^0 + \frac{W}{2} r_1}{\phi_1 C'} - \frac{W}{2}) \\ & - b_2 \times (\frac{q_2^0 + \frac{W}{2} r_2}{\phi_2 C'} - \frac{W}{2})\end{aligned}$$

subject to the following constraints

$$\begin{aligned}\textstyle\sum_{i=1}^{2} \phi_i &= 1 - \frac{r_3}{C'} \\ \varphi_1^3 \leq \phi_1 &\leq \varphi_1^1 \\ \varphi_2^3 \leq \phi_2 &\leq \varphi_2^1\end{aligned}$$

Insight about the nature of the problem under consideration is obtained by examining the following plots. Suppose that the intersection of the constraints from case 1 occurs inside the region determined by the inequality $\phi_1 + \phi_2 \leq 1 - \frac{r_3}{C'}$ (see Figure

6.5), we can then conclude case 1 is feasible. It is also then easy to see that we do not have to consider the optimization problem given in case 4. To summarize briefly, the sufficient condition for case 1 is $\max[\varphi_1^1, \varphi_1^2] + \max[\varphi_2^1, \varphi_2^2] \leq 1 - \frac{r_3}{C'}$.

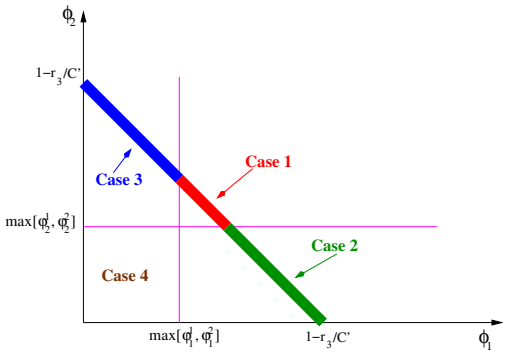


Figure 6.5: Structure of the overall optimization problem when case 1 is feasible.

If, on the other hand, the intersection of the constraints from case 1 occurs outside the region determined by $\phi_1 + \phi_2 \leq 1 - \frac{r_3}{C'}$ (see Figure 6.6), then case 1 becomes infeasible and we have only to consider the optimal solutions for the remaining 3 cases. In the latter case, we need to further examine the constraints used in case 2, 3

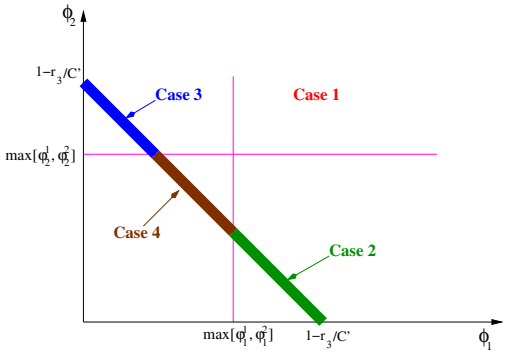


Figure 6.6: Structure of the overall optimization problem when case 1 is not feasible.

and 4. Given the magnitude of the lower bounds (φ_1^3 and φ_2^3) and the upper bounds (φ_1^1 and φ_2^1), the detailed sufficient conditions for the feasibility regions of different cases are the following:

1. If $\max[\varphi_1^1, \varphi_1^2] + \varphi_2^3 \leq 1 - \frac{r_3}{C'}$, case 2 is feasible.

2. If $\max[\varphi_2^1, \varphi_2^2] + \varphi_1^3 \leq 1 - \frac{r_3}{C'}$, case 3 is feasible.

3. If $\varphi_1^3 + \varphi_2^3 \leq 1 - \frac{r_3}{C'}$, case 4 is feasible.

In Figure 6.7, we give an example for the constraints that renders cases 2 , 3, and 4 feasible, while a different set of constraints, shown in Figure 6.8 makes case 4 the only feasible one.

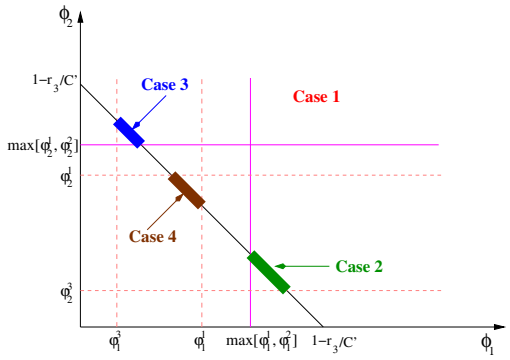


Figure 6.7: A more refined view of the feasible cases, when case 1 is not feasible.

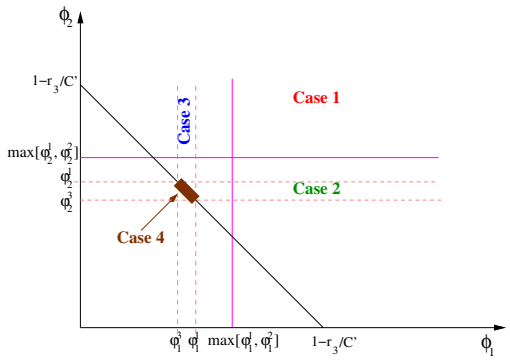


Figure 6.8: Another more refined view of the feasible cases, when case 1 is not feasible.

6.3.2 The Under-Provisioned Case

In many instances, the sum of the estimated arrival rates can be greater than C', which leads to an under-provisioned system. In such a case, it often leads to

an exponential increase in the queue length process and correspondingly in the delay process. In this case, the system is not capable (even by ignoring the QoS constraints) to accommodate all three traffic classes. It becomes then necessary to sacrifice some of QoS requirements of the lower priority classes. However, we also need to ensure that every class is going to obtain some percentage of the available bandwidth, in order to avoid starvation phenomena.

In our proposed scheme we allocate all the necessary bandwidth to the higher priority classes, in order to satisfy their QoS requirements first, and the remaining bandwidth is given to the best effort class. Hence, we have for the unstable $\phi_3 = 1 - \phi_1^{QoS} - \phi_2^{QoS}$.

For the unstable case, the previously presented maximization problem is infeasible due to the conflict between the constraints. According to the above discussed rule for the best effort class we have that if $\phi_3 C' \geq r_3$ is relaxed, then the maximization problem is still solvable. In this case, we have $0 < \phi_3 < \frac{r_3}{C'}$; or equivalently, $1 - \frac{r_3}{C'} < \phi_1 + \phi_2 < 1$. The maximization problem is fairly complicated to solve, although the constraints can be obtained if ϕ_3 is given. To simplify the problem, only the bounds of the feasible regions for each case are considered while the interior optimal solution is ignored.

However, for the profit maximization problem the relaxation of the constraints is critical in allocating the available bandwidth. This relaxation takes place in two stages: relaxation between the classes (large scale) and relaxation within a class (small scale).

For the relaxation within a class, the relationships between those constraints for i-th class can be written as follows:

$$\max[\varphi_i^1, \varphi_i^2] \geq \min[\varphi_i^1, \varphi_i^2] \geq \varphi_i^3 \geq \frac{r_i}{C'}.$$

Hence, our proposed scheme should relax the constraints recursively in the above descending order. Regarding the relaxation of the constraints between the classes, our rules imply that the constraints of the higher class are not relaxed until all of the constraints of the lower classes in the above order are relaxed. In this way, the higher classes of traffic can be guaranteed with the best service that the system can provide.

Finally, in order to avoid starvation of a particular class (that would happen for the best effort class, if for example, $\sum_{i=1}^{2} r_i \geq C'$), the network provider can implement prespecified thresholds to discipline the allocation for each class, as studied in our previous work [64]. In such instances, the QoS requirements of the delay-adaptive class should be satisfied at the expense of the remaining two classes, provided that it does not exceed a prespecified threshold θ_1. Furthermore, the QoS requirement of the loss sensitive class should also be satisfied, provided that it does not exceed a different prespecified threshold θ_2.

6.4 Calculating the Optimal Solution

In this section we continue our investigation into the solution of the optimization problems given in cases 1-4 under the stable case. In principle, the problem can be solved by nonlinear optimization methods. For the objective function derived in cases 1-4 it can be shown that the Hessian matrix of second partial derivatives is negative definite. For example, for the objective function in case 4 the Hessian is given by

$$\mathbf{H} = \begin{pmatrix} -\frac{2b_1(q_1^0+\frac{W}{2}r_1)}{\phi_1^3 C'} & 0 \\ 0 & -\frac{2b_2(q_2^0+\frac{W}{2}r_2)}{\phi_2^3 C'} \end{pmatrix}$$

which, under the feasibility constraints, is negative definite (since all its eigenvalues are negative). Therefore, it can be concluded that the objective function is jointly concave and hence possesses a unique maximum (maybe at a boundary point), which can be obtained by solving for the classical Kuhn-Tucker conditions.

However, by further exploring the structure of the problem at hand we can obtain the optimal solution in a more inexpensive and easy to implement manner. We illustrate the main steps of the proposed approach on the problem defined in case 4. The other optimization problems (cases 1-3) can be solved in an analogous manner (the details can be found in [65–67]). By solving the feasibility constraint $\phi_1 + \phi_2 = 1 - \frac{r_3}{C'}$ for ϕ_2 and substituting that value in the objective function we find a new

objective function of a single variable given by

$$g(\phi_1) = p_1\phi_1 C' + p_2(1 - \frac{r_3}{C'} - \phi_1)C' - b_1(\frac{q_1^0 + \frac{W}{2}r_1}{\phi_1 C'} - \frac{W}{2}) - b_2(\frac{q_2^0 + \frac{W}{2}r_2}{(1 - \frac{r_3}{C'} - \phi_1)C'} - \frac{W}{2})$$

Its first and second derivatives are given next:

$$g'(\phi_1) = (p_1 - p_2)C' + \frac{b_1(q_1^0 + \frac{W}{2}r_1)}{\phi_1^2 C'} - \frac{b_2(q_2^0 + \frac{W}{2}r_2)}{(1 - \frac{r_3}{C'} - \phi_1)^2 C'}$$

$$g''(\phi_1) = -2b_1 \times \frac{q_1^0 + \frac{W}{2}r_1}{\phi_1^3 C'} - 2b_2 \times \frac{q_2^0 + \frac{W}{2}r_2}{(1 - \frac{r_3}{C'} - \phi_1)^3 C'}$$

It can easily be seen that $g''(\phi_1) < 0$, which implies that $g(\phi_1)$ is a concave function. Plots of the objective function $g(\phi_1)$ and its first derivative $g'(\phi_1)$ are shown in Figure 6.9. The derivation of the $g'(\phi_1)$ helps us determine the optimal solution, as follows.

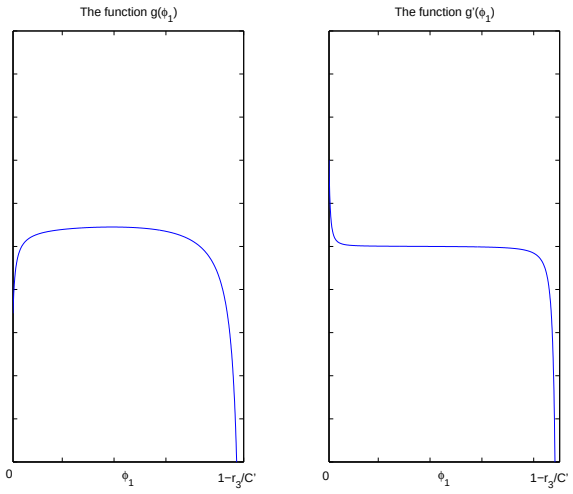


Figure 6.9: The graph of $g(\phi_1)$ and $g'(\phi_1)$

First denote the lower bound of the feasible region by B_L and the upper bound by B_U. If $g'(B_L) > 0$ and $g'(B_U) \geq 0$, then the optimal solution is given at the boundary by B_U, whereas if $g'(B_L) \leq 0$ and $g'(B_U) < 0$, then the optimal solution is given at the other boundary point B_L. Finally, if $g'(B_L) > 0$ and $g'(B_U) < 0$, then the optimal

solution lies in the interior of the interval (B_L, B_U) and must be found by numerical root finding methods, such as the bisection method, or Newton's method [11].

The globally optimal solution is then obtained by calculating first the optimal solution $\phi_1^*(k),\ k = 1, 2, 3, 4$ for the 4 cases and then keeping the maximum amongst the four. As discussed in the previous section, some of the cases may not be feasible, a fact that leads to a speed-up of the algorithm. Finally, combined with our unstable case solution, the algorithm for the optimal solution is given in pseudo-code form next.

6.5 Sensitivity Analysis of the Optimal Solution

In this subsection we briefly explore the effect of the prices charged to the users $(p_1,\ p_2)$ and the costs associated with queueing delays $(b_1,\ b_2)$ on the optimal solution.

By regarding the objective function $g(\phi_1)$ as a function of two arguments , i.e. $\tilde{g}(\phi_1, p_1)$, and taking its partial derivative with respect to the decision variable ϕ_1 and the price charged p_1 we get (for case 4 and analogously for all the other cases as well)

$$\frac{\partial^2 \tilde{g}(\phi_1, p_1)}{\partial p_1 \partial \phi_1} = C' > 0.$$

Based on Theorem 4, we can conclude that $\tilde{g}(\phi_1, p_1)$ is supermodular (proof given in Appendix I [54]). Analogously we get that $\frac{\partial^2 \tilde{g}(\phi_1, p_2)}{\partial p_2 \partial \phi_1} = -C' < 0$, thus $\tilde{g}(\phi_1, p_2)$ is submodular [54]. Notice that due to the similarities between the supermodular and submodular cases, only the definition for supermodularity and the corresponding proof are given in Appendix B.

The effect of the prices on the shape of the function $g(\phi_1)$ and its first derivative $g'(\phi_1)$ is illustrated in Figures 6.10-6.11. The above simple derivations (as well as the plots) indicate that the higher the price charged to the delay sensitive class, the higher the bandwidth allocated to that class would be, if the optimal solution is located in the interior of the feasibility region. The proof of this observation is also given in the Appendix B based on Lemma 1 [54]. Analogously, the higher the price of the loss sensitive class, the lower the bandwidth allocated to the delay sensitive

class and consequently the higher the bandwidth allocated to the loss sensitive case, in the presence of an optimal solution in the interior of the feasible region.

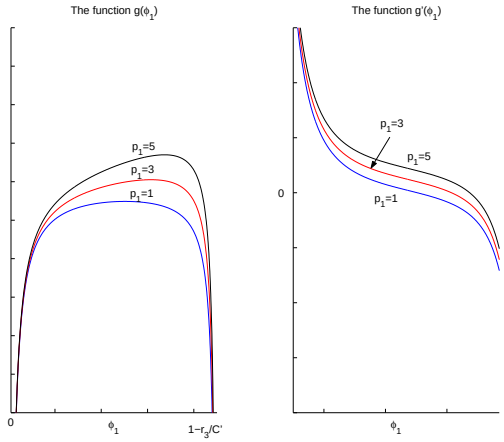


Figure 6.10: The graph of $g(\phi_1)$ and $g'(\phi_1)$ for different prices of the delay-adaptive class.

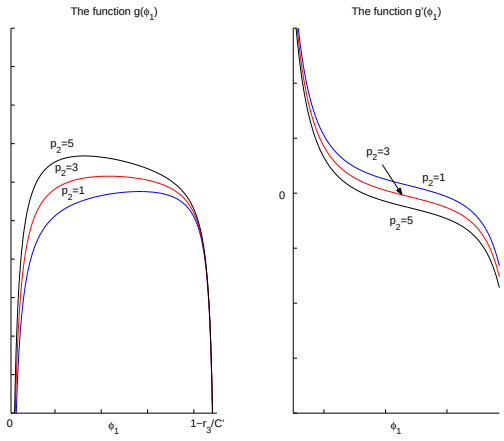


Figure 6.11: The graph of $g(\phi_1)$ and $g'(\phi_1)$ for different prices of the loss sensitive class.

We now turn our attention to the cost component. Defining a function of two variables $\tilde{g}(\phi_1, b_1)$ (for case 4) and taking its second partial derivative with respect to both arguments we get

$$\frac{\partial^2 \tilde{g}(\phi_1, b_1)}{\partial b_1 \partial \phi_1} = \frac{q_1^0 + \frac{W}{2} r_1}{\phi_1^2 C'} > 0.$$

which shows that $\tilde{g}(\phi_1, b_1)$ is supermodular. An analogous derivation shows that $\tilde{g}(\phi_1, b_2)$ is submodular, since $\frac{\partial^2 \tilde{g}(\phi_1, b_2)}{\partial b_2 \partial \phi_1} < 0$. It is easy then to conclude that the higher the cost of the delay for the delay sensitive case, the higher the bandwidth allocated to it, as Figure 6.12 also indicates. The intuitive explanation behind this result goes as follows: the higher the delay cost for the 1st class, the bigger the incentive of the provider to *decrease* the delay of that class' customers; hence, the higher the bandwidth allocated to the delay sensitive class. A similar reasoning applies to the loss sensitive class, which shows that the higher its delay cost, the higher the bandwidth allocated to it should be, which in turn implies the lower the bandwidth allocated to the delay sensitive class (as can also be seen from Figure 6.13).

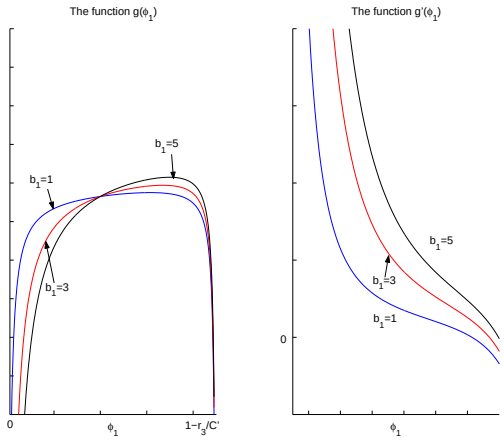


Figure 6.12: The graph of $g(\phi_1)$ and $g'(\phi_1)$ for different values of the cost of the delay sensitive class.

6.6 The Choice of W

The choice of W is crucial in determining the performance of our proposed scheme. Intuitively, if the value of W is too small, it can capture the burstiness of the incoming traffic and result in an appropriate allocation of resources. The QoS requirements will be satisfied, but the computational requirements of the proposed scheme can

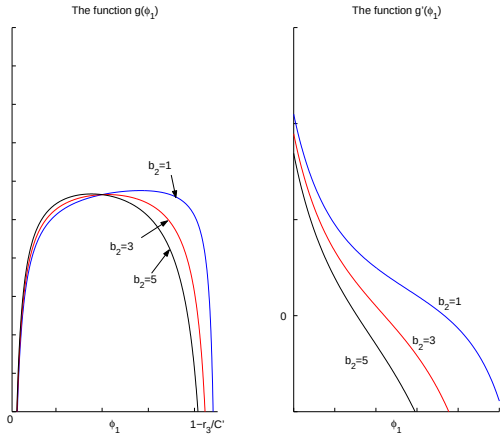


Figure 6.13: The graph of $g(\phi_1)$ and $g'(\phi_1)$ for different values of the cost of the loss sensitive class.

become rather high. On the other hand, if the value of W is too large, the measurements cannot capture well the fluctuations in the traffic processes, thus compromising performance.

We can do a similar sensitivity analysis for W to that presented in the previous section. Consider the objective function $g(\phi_1)$ as a function of two arguments, i.e. $\tilde{g}(\phi_1, W)$, and taking its partial derivative we get (for case 4 and analogously for all the other cases as well)

$$\frac{\partial^2 \tilde{g}(\phi_1, W)}{\partial W \partial \phi_1} = \frac{b_1 \times r_1}{2\phi_1^2 C'} - \frac{b_2 \times r_2}{2(1 - \frac{r_3}{C'} - \phi_1)^2 C'} \tag{6.8}$$

From the above equation, it is hard to tell whether this partial cross derivative is greater or less than 0; therefore, it is hard to conclude whether the function is submodular or supermodular, which in turn implies that W does not share the monotonically increasing or decreasing property with the optimal value of ϕ_1 for the over-provisioned (stable) case.

Nevertheless, we can obtain the critical point of equation 6.8 as follows:

$$\phi_1 = \frac{1 - \frac{r_3}{C'}}{1 + \sqrt{\frac{b_2 r_2}{b_1 r_1}}}$$

When ϕ_1 is larger than this threshold value, the function is supermodular, the optimal value of ϕ_1 increases as W increases; otherwise, it is submodular and the

optimal value of ϕ_1 decreases as W increases. Given the value of $\frac{r_3}{C'}$, the value of this threshold value depends on $\sqrt{\frac{b_2 r_2}{b_1 r_1}}$, which can be interpreted as the ratio of the traffic loads of the two higher priority classes. As the ratio between these classes becomes more unbalanced, $\sqrt{\frac{b_2 r_2}{b_1 r_1}}$ increases so that the threshold decreases. The cost prices b_1 and b_2 can be used to counteract the effect of the increase in the traffic unbalance between the delay-adaptive and rate-adaptive classes.

However, we also know that as W increases, the effective bandwidth estimation decreases [47], and all the constraints decrease too. For the under-provisioned case, the increase of W will turn out the more conservative value for the constraints, accordingly affecting the allocated bandwidth. It will result in an inferior performance for our adaptive scheduling scheme.

6.7 Performance Evaluation

The model used in our experimental investigations is a processor sharing system with three input queues that map to the delay-adaptive, the rate-adaptive and elastic class, respectively. The link capacity is set at 1 Mb/s and the model parameters employed in the simulation are given in table 6.1.

For the adaptive algorithms proposed in this dissertation, the measurement window was set to 1 sec. An on-off input Fractal Modulated Poisson process (FMPP) model, proposed in [4], was employed. Finally, for simplicity purposes, the packet size was fixed to 50 bytes and the Hurst parameter of the traffic process to 0.75. Several other simulation scenarios were considered in [64] that produced analogous results.

Table 6.1: Parameters for Different Classes

	Delay-Adaptive	Rate-Adaptive	Elastic Class
Initial Weight	2	2	1
Buffer Size (bytes)	500	5000	10000
p (cents/kbps)	5	1	–
b (cents/ms)	0.5	0.1	–

Given the links speed, the system's capacity (stability) constraint is at 0.4 ms. That is, if the mean packet inter-arrival time is larger than this value, the system is over-provisioned and relatively few packet losses are expected to occur. On the other hand, for mean packet inter-arrival times smaller than 0.4 ms, the system is unstable and highly backlogged queues together with frequent packet losses are expected. The case where the mean time is exactly 0.4 corresponds to the critical regime.

In the ensuing discussion, the classical WRR and WFQ with static allocated weights are denoted by SWRR and SWFQ, respectively, whereas their dynamic counterparts described in Scheme I are denoted by DWRQ (Dynamic Weighted Round Robin) and DWFQ (Dynamic Weighted Fair Queue).

Finally, the profit based optimization policy presented in Scheme II is denoted as ODWFQ (Optimized Dynamic Weighted Fair Queue). The performance metrics used in our study are packet loss probabilities and average queue delays. In order to facilitate the presentation of the results we separate the under- from the over-provisioned cases, due to the incorporation of 'non-starvation' thresholds in the proposed policies. In the plots that follow, the class average loss probability and average delay together with 95% confidence intervals obtained from 50 replications are shown.

In the four panels of Figure 6.14, the performance of all 5 algorithms under study for the delay-adaptive class is shown. Specifically, in Figures 6.14 (a) and (b) show that the adaptive versions of the WRR and WFQ outperform with respect to losses their static counterparts, for both the under- and over-provisioned scenarios. The ODWFQ exhibits the best performance in all cases, and in particular for stable but heavily loaded systems (range between 0.42 and 0.44 ms). Finally, as the system becomes more lightly loaded (0.48 ms and beyond) all policies become essentially equally efficient. Regarding the average delay metric (Figures 6.14 (c) and (d)), the ODWFQ dominates all other policies in the stable case and performs very well in the under-provisioned scenario. From the remaining policies, we notice the strong performance of the DWFQ policy, and the inferior performance of the DWRR policy. The latter finding is due to the inaccuracy involved with the normalization of the weights.

In the panels of Figure 6.15, the performance of the rate-adaptive class is ex-

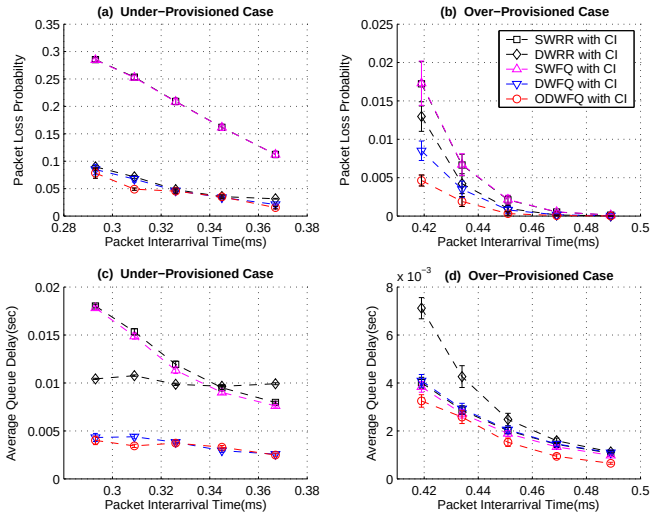


Figure 6.14: Comparison of scheduling algorithms on packet loss probability in (a) under-provisioned and (b) over-provisioned cases; on average queue delay in (c) under-provisioned and (d) over-provisioned cases for the delay-adaptive class.

amined. For this class the ODWFQ policy clearly outperforms all its competitors, for both performance metrics in the under- and over-provisioned cases. Among the remaining policies the DWFQ outperforms its static counterpart in all cases. On the other hand, the pattern for the round robin policies is more involved. For example, the DWRR clearly outperforms the SWRR policy for both performance metrics in the under-provisioned case and for the loss metric in the over-provisioned case; however, the SWRR policy exhibits a better performance with respect to delay for the over-provisioned case. Finally, it should be noted that as the incoming traffic's rate decreases (the mean interarrival time increases) and the system's load becomes lighter, the performance of all three policies become fairly similar.

In the panels of Figure 6.16 the results for the elastic class are shown. For this class the ODWFQ exhibits the worst performance for all cases and metrics, due to the fact that this class yields very low profits and hence less bandwidth is allocated to it under this policy. For this class, the static policies achieve the best performance in terms of loss and average queue delay for the elastic class in both under-provisioned and over-provisioned cases, since the dynamic policies allocate less bandwidth to this class, in order to meet the QoS requirements of the higher priority classes.

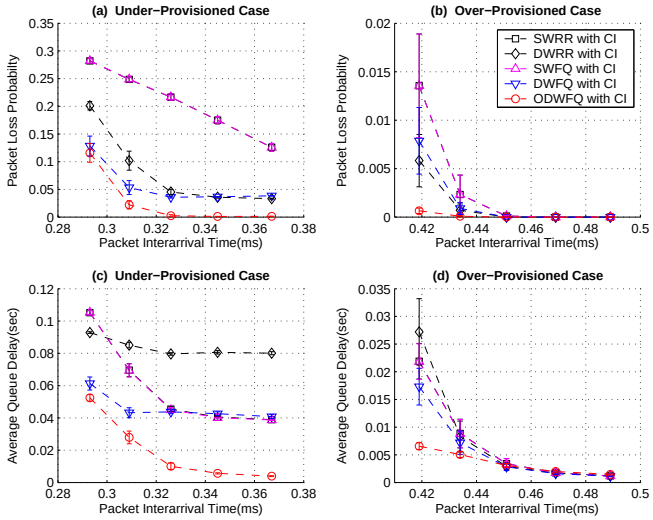


Figure 6.15: Comparison of scheduling algorithms on packet loss probability in (a) under-provisioned and (b) over-provisioned cases; on average queue delay in (c) under-provisioned and (d) over-provisioned cases for the rate-adaptive class.

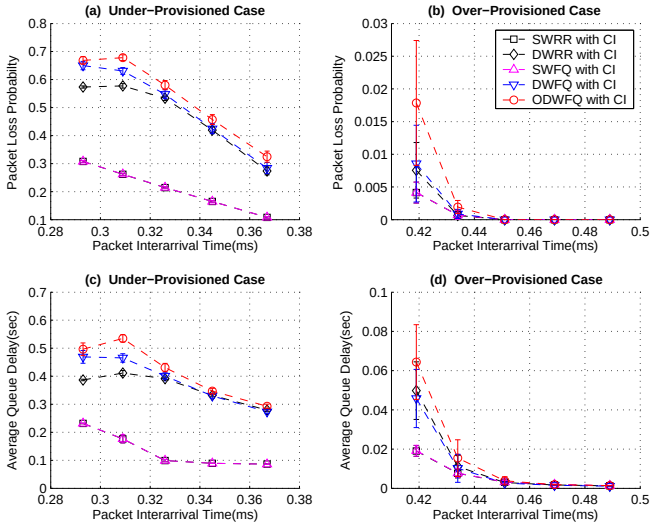


Figure 6.16: Comparison of scheduling algorithms on packet loss probability in (a) under-provisioned and (b) over-provisioned cases; on average queue delay in (c) under-provisioned and (d) over-provisioned cases for the elastic class.

For the over-provisioned case, profit maximization is an important objective. In Figure 6.17 the results of the profits obtained by the ODWFQ, DWFQ and SWFQ policies are shown. Notice that for the ODWFQ policy the profit is calculated both analytically and by simulation. It can be seen that the profit obtained by the ODWFQ policy is significantly larger than those obtained from the other two policies. In addition, the adaptive version of the WFQ policy exhibits a small advantage over the static one.

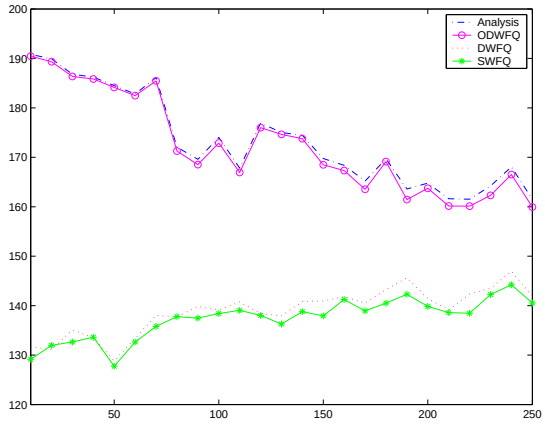


Figure 6.17: Comparison of scheduling algorithms on maximum of profit

In conclusion, we have that the proposed ODWFQ policy exhibits the best and most robust performance for both the over- and under-provisioned scenarios for the high paying traffic classes.

6.8 Conclusions

Adaptive scheduling based on measurements of traffic and queueing state has the potential of greatly improving the efficiency of resource allocation techniques. In our previous work on this topic, we have introduced a measurement-based adaptive scheduler and validated its performance with extensive simulation results. In this chapter, we have formulated the online setting of adaptive schedulers as a formal

optimization problem taking into account QoS constraints and the underlying pricing scenario. We then proceeded to study its solutions on a case-by-case basis, thus obtaining fundamental insights regarding the implementation and use of such schemes.

Continuing and extending our efforts in this area, we are working to analyze the behavior of the adaptive scheduler over time. In this chapter the optimization problem given in Section 6.3.1 is solved at every decision instant – which corresponds to the beginning of a new window W. Notice that the window size affects which constraints become binding in our optimization problem. In our previous work [64] we have empirically investigated the problem of dynamically adapting the size of the window W to changing traffic conditions. A topic of current research is to study the dynamics over time of the window size, as well as the long-term performance of the system under changing traffic patterns.

Algorithm 1 Obtaining the optimal solution

if $\sum_{i=1}^{3} r_i < C'$ **then**

Identify the N feasible cases by checking the underlying feasibility constraints

while $N > 0$ **do**

Obtain B_L^i, B_U^i for case i

if $g'(B_L^i) > 0$ and $g'(B_U^i) \geq 0$ **then**

$\phi_1^{i*} = B_U^i$

else if $g'(B_L^i) \leq 0$ and $g'(B_U^i) < 0$ **then**

$\phi_1^{i*} = B_L^i$

else

Use root finding method for solving $g'(\phi_1) = 0$ to obtain ϕ_1^{i*}.

end if

Obtain the maximum for case i, $\max f^{i*}(\phi_1^{i*}, \phi_2^{i*})$

$N = N - 1$

end while

$\max f^*(\phi_1^{i*}, \phi_2^{i*}) = \max_{i \in N} f^{i*}(\phi_1^{i*}, \phi_2^{i*})$

else

if $(\frac{r_1}{C'} + \frac{r_2}{C'}) < 1$ **then**

Constraints relaxation in Section 6.3.2

else

The threshold (θ_1, θ_2) in 4

end if

end if

Chapter 7

MBORA System under Nonlinear Pricing Models

7.1 Introduction

In chapter 6, a linear pricing model for the cost component of the profit maximization center was used and the bandwidth allocation problem studied. However, one shortcoming of the model is that it is not sensitive enough to the requirements of the more delay sensitive classes and also to the prices charged. Hence, a new delay-sensitive nonlinear pricing model that incorporates an exponential component, is examined. The exponential component proves sensitive to delay considerations and in addition the tuning parameters of the model (α and β) give additional flexibility to the network provider for allocating bandwidth optimally and balancing profits and costs due to incurred delays. Furthermore, in order to address the scalability issues a new solution strategy is examined. In particular, the fluid queueing model used to derive the average expected delay and its properties are examined in more detail. The ensuing analysis established the concavity of the objective function and hence

the existence of a global maximum that can be located through the quasi-Newton method. This in turn reduces the computational complexity and makes an online implementation feasible as outlined later on.

The remainder of the chapter is organized as follows. We first introduce linear and nonlinear pricing models for the generalized profit-oriented formulation. In Section 7.3, the new solution strategy is discussed, in which we present the concavity proofs for both pricing models and solution procedures. Section 7.4 gives more discussion and analysis on elasticity and complexity issues, while numerical studies are presented in Section 7.5. We give our concluding remarks in Section 7.6.

7.2 Pricing Models

In this section, two models for the cost component of the profit maximization center are discussed. The first model, a linear one already presented in the previous chapter, is included for completeness purposes.

7.2.1 Linear Pricing Model

The profit maximization problem is defined as:

$$\max_{\phi} f(\phi) = \sum_{i} U_i(\phi_i) = \sum_{i} R_i(\phi_i) - \sum_{i} \mathcal{C}'_i(\phi_i) \tag{7.1}$$

s.t.

$$\sum \phi_i \leq 1$$

If the linear pricing model is applied for both the utility-based revenue component and delay-incurred component, then the objective function can be written as:

$$\max_{\phi} f_{lp}(\phi) = \sum_{i} p_i \phi_i C' - \sum_{i} b_i D_i \tag{7.2}$$

s.t.

$$\sum \phi_i \leq 1$$

where p_i is the price per unit of the utilization of the system's resources for class i and b_i is the cost per unit of time incurred by class i, and with $\phi = [\phi_1\ \phi_2 \cdots \phi_N]$.

The objective of the linear pricing model is to realize QoS differentiation among the classes by the simple pricing discrepancies between the classes with the different priorities. In other words, since the higher priority class requests a better service level than the lower priority classes, it should be charged with a higher price, while at the same time the provider has to reimburse the users at a higher rate for delay violations; therefore, $p_i > p_j$ and $b_i > b_j$ for $i < j$.

7.2.2 Nonlinear Pricing Model

We present next a nonlinear model for the cost component that proves significantly more sensitive to delay considerations than the linear model. The model is given next.

$$\mathcal{C}'_{i,np}(\phi_i) = b_i e^{\beta_i (D_i(\phi_i) - \alpha_i d_i)} D_i(\phi_i) \tag{7.3}$$

where β_i is a positive number and $0 < \alpha_i \leq 1$.

As it can be seen from equation 7.3, the nonlinear pricing model is transformed from the linear pricing one by inserting the exponential factor. We also introduce the aggressive factor, β_i, and the proactive factor, α_i, for class i. These two factors will be discussed in our elasticity analysis for the nonlinear pricing model.

The advantage of this formulation is that it takes into consideration the magnitude of the violation of the delay constraint. To see this, let $\mathcal{C}'_{i,lp}$ denote the delay-incurred cost function under the linear pricing model and $\mathcal{C}'_{i,np}$ as the one under the nonlinear pricing model. It is easy to obtain the first derivative of the linear pricing model with respect to D_i, $\frac{\partial \mathcal{C}'_{i,lp}}{\partial D_i} = b_i$. It remains unchanged with no reference to the expected delay, D_i and the desired delay bound, d_i. On the other hand, the first derivative of the nonlinear model is given by:

$$\frac{\partial \mathcal{C}'_{i,np}}{\partial D_i} = b_i(\beta_i D_i + 1) e^{\beta_i (D_i - \alpha_i d_i)} \tag{7.4}$$

It can be shown that it is monotonically increasing with respect to D_i. Based on this property, we can conclude that the nonlinear pricing model is sensitive to the

desired delay bound, d_i, because $\Delta\mathcal{C}'_{i,lp}$ for $D_i > d_i$ is much larger than $\Delta\mathcal{C}'_{i,lp}$ for $D_i < d_i$ given the same ΔD_i. Hence, this model is sensitive to QoS considerations. Further, the delay-incurred cost based on the nonlinear pricing model is lower than the one under the linear pricing scheme when the delay bound is satisfied; on the other hand, it will be significantly higher (with an exponential increasing rate) when the delay bound is violated.

The objective function is then given by:

$$\max_{\phi} f_{np}(\phi) = \sum_i p_i \phi_i C' - \sum_i b_i e^{\beta_i (D_i(\phi_i) - \alpha_i d_i)} D_i(\phi_i) \tag{7.5}$$

s.t.

$$\sum \phi_i \leq 1$$

7.3 Solution Strategy

We discuss next a general solution strategy for both models in the presence of n flexible delay-bound classes.

7.3.1 Some properties of the Fluid Queueing Model

First, we will discuss a key property of the fluid queueing model, since it proves useful in calculating D_i for both pricing models.

In the fluid queueing model, the average queue length of class i, $\bar{q}_i(\phi_i)$, is calculated by the following equation.

$$\begin{aligned} \bar{q}_i &= \frac{1}{W} \int_0^{\tau_i} q_i(t) dt \\ &= \frac{\tau_i}{W} [q_i^0 + \frac{\tau_i}{2} (r_i - \phi_i C')] \end{aligned} \tag{7.6}$$

where $\tau_i = \min\{t_i^0, W\}$ with t_i^0 being the time it takes to empty the queue.

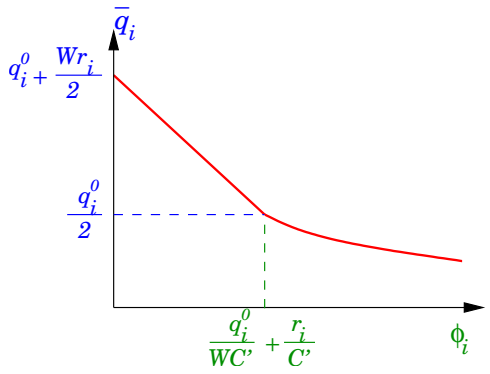


Figure 7.1: The function of average queue length, $\bar{q}_i$.

Therefore, we can obtain

$$\bar{q}_i(\phi_i) = \begin{cases} \frac{(q_i^0)^2}{2W} \times \frac{1}{\phi_i C' - r_i} & \text{if } \phi_i > \psi_i^c \\ q_i^0 + \frac{W}{2}(r_i - \phi_i C') & \text{if } \phi_i \leq \psi_i^c \end{cases} \tag{7.7}$$

where $\psi_i^c = \frac{q_i^0}{WC'} + \frac{r_i}{C'}$. It is obvious that $\bar{q}_i(\phi_i)$ is a continuous piecewise function from equation 7.7, given $\phi_i \in (0, 1)$. Also, $\bar{q}_i(\phi_i)$ has the following properties:

1. $\bar{q}_i(\phi_i)$ is differentiable everywhere for $\phi_i \in (0, 1)$, and twice-differentiable except the point $\phi_i = \psi_i^c$.

2. $\bar{q}_i(\phi_i)$ is convex for $\phi_i \in (0, 1)$.

The proofs of these properties can be found in Appendix C.

In our system, the expected delay of class i is calculated by $D_i(\phi_i) = \frac{\bar{q}_i}{\phi_i C'}$. Thus, we can obtain D_i based on equation 7.7 as follows:

$$D_i(\phi_i) = \begin{cases} \frac{(q_i^0)^2}{2Wr_i} \times [\frac{1}{\phi_i C' - r_i} - \frac{1}{\phi_i C'}] & \text{if } \phi_i > \psi_i^c \\ \frac{q_i^0 + \frac{W}{2} r_i}{\phi_i C'} - \frac{W}{2} & \text{if } \phi_i \leq \psi_i^c \end{cases} \tag{7.8}$$

Next, we will investigate the properties of $D_i(\phi_i)$ in order to obtain better insight, useful for coming up with a solution strategy for the underlying optimization problem.

Proposition 1. *$D_i(\phi_i)$ is differentiable everywhere for $\phi_i \in (0,1)$, and twice differentiable except the point $\phi_i = \psi_i^c$.*

Proof. For both sides of the piecewise function except the point, $\phi_i = \psi_i^c$, it is obvious differentiable, thus we can obtain the first derivative of $D_i(\phi_i)$ for both sides as follows:

$$\frac{\partial D_i}{\partial \phi_i} = \begin{cases} -\frac{(q_i^0)^2 C'}{2Wr_i} \times [\frac{1}{(\phi_i C' - r_i)^2} - \frac{1}{(\phi_i C')^2}] & \text{if } \phi_i > \psi_i^c \\ -\frac{q_i^0 + \frac{W}{2} r_i}{\phi_i^2 C'} & \text{if } \phi_i \leq \psi_i^c \end{cases} \tag{7.9}$$

Based on definition 3 in Appendix C, we can derive that the left derivative and right derivative at the point $\phi_i = \psi_i^c$ are equal, $D_- D_i(\psi_i^c) = D_+ D_i(\psi_i^c)$. Therefore, $D_i(\phi_i)$ is the differentiable for $\forall \phi_i \in (0,1)$.

Then we can get the second derivative of D_i for both sides as follows:

$$\frac{\partial^2 D_i}{\partial \phi_i^2} = \begin{cases} \frac{(q_i^0 C')^2}{Wr_i} \times [\frac{1}{(\phi_i C' - r_i)^3} - \frac{1}{(\phi_i C')^3}] & \text{if } \phi_i > \psi_i^c \\ \frac{2q_i^0 + Wr_i}{\phi_i^3 C'} & \text{if } \phi_i \leq \psi_i^c \end{cases} \tag{7.10}$$

However, at the point of $\phi_i = \psi_i^c$, it is easy to prove that the left second derivative is not equal to the right second derivative, thus D_i is not twice differentiable at this point but twice differentiable for all others points in the domain $(0,1)$. □

Proposition 2. *$D_i(\phi_i)$ is convex for $\phi_i \in (0,1)$.*

Proof. ¿From equation 7.10, it can be seen that

$$[\frac{1}{(\phi_i C' - r_i)^3} - \frac{1}{(\phi_i C')^3}] > 0 \quad \text{and} \quad \frac{2q_i^0 + Wr_i}{\phi_i^3 C'} > 0$$

for $\forall \phi_i \in (0,1)$. This implies that both sides of the first derivative, $\frac{\partial D_i}{\partial \phi_i}$, are monotonically increasing.

Since we prove $D_i(\phi_i)$ is differentiable everywhere in the proposition 1, we can conclude that the first derivative of $D_i(\phi_i)$ is monotonically increasing for $\phi_i \in (0,1)$. Thus, $D_i(\phi_i)$ is convex on $(0,1)$. □

Based on the discussion of the fluid queueing model and $D_i(\phi_i)$, we can conclude the piece-wise characteristics for the objective functions under both pricing models. However, the main concern is that $D_i(\phi_i)$ is not twice-differentiable at the point $\phi_i = \psi_i^c$. Therefore, optimization based on Newton's method is not feasible; in addition, calculating the Hessian matrix is an expensive task.

Nevertheless, a naive approach that would overcome this technical difficulty would be to evaluate D_i from both sides of the critical point, $\phi_i = \psi_i^c$, respectively. In this way, the corresponding objective functions can be guaranteed to be twice-differentiable and solved by Newton's method later. Subsequently, the maximal values of all the objective functions are compared, and the maximum among them corresponds to the optimal solution. However, in the presence of n classes, this approach proves non-scalable, since $O(2^n)$ evaluations would be required.

To overcome the scalability problem, we propose to evaluate D_i as the whole function in our objective functions, relying on the concavity of the objective function.

7.3.2 Concavity Proofs under Linear Pricing Model

Definition 1. *(Monotone Function): The mapping $f : \mathbb{R}^N \to \mathbb{R}^N$ is said to be monotone on $\mathbb{R}^N$ when, for all u and u' in $\mathbb{R}^N$*

$$\langle f(u) - f(u'), u - u' \rangle \geq 0 \tag{7.11}$$

where $\langle ., . \rangle$ represents the inner product operator.

Theorem 1. *Suppose f is a differentiable function on $\mathbb{R}^N$. Then f is convex on $\mathbb{R}^N$ if and only if its gradient ∇f is monotone on $\mathbb{R}^N$.*

Proposition 3. *Assume $f_{lp}^2 = -\sum_{i=1}^N b_i D_i(\phi_i)$. Then it is concave for $\phi \in R^N$, where $D_i(\phi_i) = \frac{\bar{q}_i}{\phi_i C'}$.*

Proof. Suppose $f_{lp}^2 = -\widetilde{f}_{lp}^2$, then the function $\widetilde{f}_2 = \sum_{i=1}^N b_i \frac{\bar{q}_i}{\phi_i C'}$. In this way, the concavity proof of f_2 is equivalent to the convexity proof of $\widetilde{f}_{lp}^2$.

We can see that $\bar{q}_i$ is one of factors in $\widetilde{f}_{lp}^2$. According to proposition 8, it is easy to prove that $\widetilde{f}_{lp}^2$ is differentiable everywhere for $\phi_i \in (0,1)$ and twice-differentiable except the vector of ϕ with one of elements equal to $\phi_i = \psi_i^c$.

First, we can obtain the gradient of $\widetilde{f}_{lp}^2$ as following:

$$\begin{aligned} \nabla \widetilde{f}_{lp}^2 &= [\frac{\partial \widetilde{f}_{lp}^2}{\partial \phi_1} \quad \frac{\partial \widetilde{f}_{lp}^2}{\partial \phi_2} \quad \cdots \quad \frac{\partial \widetilde{f}_{lp}^2}{\partial \phi_N}] \\ &= [b_1 \frac{\partial D_1}{\partial \phi_1} \quad b_2 \frac{\partial D_2}{\partial \phi_2} \quad \cdots \quad b_N \frac{\partial D_N}{\partial \phi_N}] \end{aligned}$$

Computing the gradient inner product (GIP)

$$\mathrm{GIP}(\phi, \phi') \triangleq \langle \nabla \widetilde{f}_{lp}^2(\phi) - \nabla \widetilde{f}_{lp}^2(\phi'), \phi - \phi' \rangle$$

where $\phi, \phi' \in \mathbb{R}^N$, we obtain

$$\mathrm{GIP}(\phi, \phi') = \sum \mathrm{gip}(\phi_i, \phi_i') \tag{7.12}$$

where

$$\mathrm{gip}(\phi_i, \phi_i') = [(\frac{\partial \widetilde{f}_{lp}^2}{\partial \phi_i}(\phi_i) - \frac{\partial \widetilde{f}_{lp}^2}{\partial \phi_i}(\phi_i')] \times [\phi_i - \phi_i']$$

As proved in proposition 2, D_i is convex and its first derivative is monotonously increasing for $\phi_i \in (0,1)$.

Therefore, given $\phi_i > \phi_i'$, $\frac{\partial \widetilde{f}_{lp}^2}{\partial \phi_i}(\phi_i) > \frac{\partial \widetilde{f}_{lp}^2}{\partial \phi_i}(\phi_i')$. Then we can derive that

$$\text{gip}(\phi, \phi') > 0 \quad \text{and correspondingly,} \quad \text{GIP}(\phi, \phi') > 0.$$

Using Theorem 1, we see that $\widetilde{f}_{lp}^2$ is convex, and correspondingly, f_{lp}^2 is concave. □

Theorem 2. *Let $f_1, \cdots f_m$ be concave/convex on $\mathbb{R}^N$, $a_1, \cdots a_m$ be nonnegative numbers, and assume that there is a point where all the f_j's are finite. Then the function*

$$f \triangleq \sum_{j=1}^{m} a_j f_j$$

is concave/convex on $\mathbb{R}^N$.

Proposition 4. *The objective function of linear pricing model $f_{lp}(\phi)$ is concave on $\mathbb{R}^N$.*

Proof. We can rewrite the objective function under linear pricing model, composed by the following two parts:

$$f_{lp} = f_{lp}^1 + f_{lp}^2$$

where $f_{lp}^1 = \sum_{i=1}^{N} p_i \phi_i C'$ and $f_{lp}^2 = -\sum_{\phi_i=1}^{N} b_i \frac{\bar{q}_i}{\phi_i C'}$.

Since f_{lp}^1 is an linear function, it is concave on $\mathbb{R}^N$. Also we prove that f_{lp}^2 is concave in Proposition 3. Applying Theorem 2, we can see that the objective function f_{lp} is concave. □

7.3.3 Concavity Proofs under Nonlinear Pricing Model

Proposition 5. *Let I be a nonempty interval of $\mathbb{R}$. A function $f : I \to \mathbb{R}$ is convex if and only if, for all $x_0 \in I$, the slope function*

$$x \mapsto \frac{f(x) - f(x_0)}{x - x_0} =: s(x)$$

in increasing on $I \backslash \{x_0\}$.

Theorem 3. *Let f be defined on $\mathbb{R}$. Then the function $g(x) := xf(x)$ is convex on $\mathbb{R}$ if and only if f is also convex on $\mathbb{R}$.*

Proof. Suppose f is convex on $\mathbb{R}$; the slope-function of g can be obtained as

$$s_g(x) := \frac{g(x) - g(x_0)}{x - x_0} = \frac{xf(x) - x_0 f(x_0)}{x - x_0}$$

Then we have

$$\begin{aligned} s_g(x) &= \frac{f(x_0)[x_0 - x] + x[f(x) - f(x_0)]}{x - x_0} \\ &= f(x_0) + x \cdot \frac{f(x) - f(x_0)}{x - x_0} \\ &= f(x_0) + x \cdot s_f(x) \end{aligned}$$

When x increases, $s_f(x)$ increases according to proposition 5, thus $s_g(x)$ increases and g is convex on $\mathbb{R}^+$. □

Proposition 6. *If $g(y)$ is a nondecreasing, single-variable convex function, and $h(x)$ is convex, $f(x) \triangleq g(h(x))$ is convex.*

Proposition 7. *Under nonlinear pricing model,* $\mathcal{C}'_{np}(\phi) = \sum \mathcal{C}'_{i,np}(\phi_i)$ *is convex on* $\mathbb{R}^N$ *for* $\forall b_i,\ \beta_i > 0$ *and* $0 < \alpha_i \leq 1$, *where* $\mathcal{C}'_{i,np}(\phi_i) \triangleq b_i D_i(\phi_i) e^{\beta_i (D_i(\phi_i) - \alpha_i d_i)}$ *for* $D_i(\phi_i) : \mathbb{R} \to \mathbb{R}$, $\phi_i \in \mathbb{R}$.

Proof. Suppose $g_i(D_i) = e^{\beta_i (D_i - \alpha_i d_i)}$, then we can rewrite $\mathcal{C}'_{i,np}$ in terms of g_i.

$$\mathcal{C}'_{i,np}(D_i) = b_i D_i g_i(D_i)$$

It is obvious that $g_i(D_i)$ is convex on $\mathbb{R}$ with respect to D_i. By applying Theorem 3, we can prove that $\mathcal{C}'_{i,np}(D_i)$ is convex on $\mathbb{R}$.

Also, since D_i is a nonnegative value and the first derivative of $\mathcal{C}'_{i,np}(D_i)$ is positive, we can conclude that $\mathcal{C}'_{i,np}(D_i)$ is a nondecreasing function with respect to D_i. And from the previous subsection, we proved that the first derivative of $D_i(\phi_i)$ is monotonously increasing, therefore $D_i(\phi_i)$ is convex on $\mathbb{R}$. According to proposition 6, we can prove that $\mathcal{C}'_{i,np}(D_i(\phi_i))$ is convex on $\mathbb{R}$.

Finally, since $\mathcal{C}'_{i,np}$ is only function of ϕ_i and it is convex on ϕ_i, we can prove that $\mathcal{C}'_{np}(\phi)$ is convex on $\mathbb{R}^N$ by applying Theorem 2. □

¿From proposition 7, we proved that $\mathcal{C}'_{np}(\phi)$ is convex on $\mathbb{R}^N$. Similar to the proofs for the linear pricing case, we can prove that $f_{np}(\phi)$ is concave on $\mathbb{R}^N$, with the application of Theorem 2, etc.

7.3.4 Solution Procedures

Next, after we prove the concavity for both linear and nonlinear pricing models, the global maximizer can be easily found by the existing standard algorithms, such

as Newton's method [26]. In this subsection, we will discuss the solution procedures to obtain the global maximizer for both pricing models, due to the similarity of their procedures.

First, it is obvious that our generalized profit oriented problem in equation 7.1 is a nonlinear optimization problem with inequality constraints, which can be solved by ensuring that the Karush-Khun-Tucker (KKT) conditions are satisfied.

According to the KKT conditions, we first need to transform the constrained problem into an unconstrained one, so that we can obtain the new objective function in terms of the Lagrangian function:

$$L(\phi, \lambda_1) = f(\phi) - \lambda_1(\sum \phi_i - 1) \tag{7.13}$$

where $\phi = [\phi_1\ \phi_2 \cdots \phi_N]$.

Its optimal solution should satisfy the following equations:

$$\nabla_\phi L(\phi^*, \lambda_1^*) = 0$$
$$\lambda_1^* \geq 0, \sum \phi_i \leq 1, \text{ and } \lambda_1^*(\sum \phi_i - 1) = 0$$

where $\nabla_\phi L = [\frac{\partial L}{\partial \phi_1}\ \frac{\partial L}{\partial \phi_2}\ \cdots \frac{\partial L}{\partial \phi_N}]$.

Specifically, each element of the vector, $\nabla_\phi L$, should be equal to 0 as follows:

$$\frac{\partial L}{\partial \phi_i} = \frac{\partial f}{\partial \phi_i} - \lambda_1^* = 0, \quad \forall i = 1, \cdots, N \tag{7.14}$$

Now, we want to investigate the property of $\frac{\partial f}{\partial \phi_i}$. For the linear pricing model, we obtain

$$\frac{\partial f_{lp}}{\partial \phi_i} = p_i C' - b_i \times \frac{\partial D_i}{\partial \phi_i}$$

As was proved in equation 7.9 in the previous section, we know that $\frac{\partial D_i}{\partial \phi_i} < 0$. Thus, $\frac{\partial f_{lp}}{\partial \phi_i} > 0$.

Similarly, we can obtain the following for the nonlinear pricing model:

$$\frac{\partial f_{np}}{\partial \phi_i} = p_i C' - b_i \times (\beta_i D_i + 1) \times e^{\beta_i (D_i - \alpha_i d_i)} \times \frac{\partial D_i}{\partial \phi_i}$$

Also, we can find that $\frac{\partial D_i}{\partial \phi_i} < 0$ and the rest of factors in this equation are positive. Therefore, $\frac{\partial f_{np}}{\partial \phi_i} > 0$.

The observations of $\frac{\partial f}{\partial \phi_i}$ for both pricing models indicate that $\lambda_1^* > 0$, in order to make equation 7.14 equal to 0. Thus, we obtain an important conclusion that the optimal solutions for both pricing models satisfy

$$\sum \phi_i^* = 1$$

Now the optimization problems actually become the classical Lagrange multiplier problems with equality constraints.

This global maximizer can be obtained by the well-know Newton's method [7,48]. However, considering that the function f is not twice differentiable for the vector ϕ for one of its elements, $\phi_i = \psi_i^c$, we can apply Quasi-Newton method to solve the problem [7]. An advantage of the Quasi-Newton method is that it avoids calculating the very computationally expensive Hessian. Instead, it constructs an approximation of the inverse Hessian matrix in order to converge fast to the optimal solution, given first derivative information and an initial positive definite matrix [7]. In this paper, we adopt the *Broyden-Fetcher-Goldfarb-Shanno (BFGS)* method regarded as one of the best Quasi-Newton method solvers.

Remark: By designating linear and nonlinear pricing models to different priority of classes, we can obtain the generalized profit formulation under the mixed pricing model. In such a case, it is still very easy to prove its concavity based on the analysis on the previous section. Moreover, it remains the Lagrange multiplier problem with the equality constraints, since $\frac{\partial f}{\partial \phi_i} > 0$, that in turn can be solved by Quasi-Newton method. Therefore, we can conclude the following as our generalized solution strategy:

For any pricing scheme that makes the objective function f concave and $\frac{\partial f}{\partial \phi_i} > 0$, we can adopt the Quasi-Newton method to obtain the global maximizer for the generalized profit-oriented formulation under that pricing scheme.

7.4 Properties of the Optimal Solution and some Discussion

In this section, we will continue our analysis and discussion of elasticity and computational complexity issues.

7.4.1 Elasticity Analysis

In this subsection, we will discuss the elasticity effect of the proactive factor α_i and the aggressive factor β_i under the nonlinear pricing model.

Let us start with the analysis on the proactive factor α_i. Given $\beta_i = 1$, we can plot a group of nonlinear pricing based cost functions by varying α_i, against a linear pricing based function as shown in Figure 7.2.

Our first observation is that $\mathcal{C}'_{i,np}$ decreases accordingly as α_i increases with the order of $\alpha''_i < \alpha'_i < 1$. Mathematically, if we derive the first derivative of $\mathcal{C}'_{i,np}$ with respect to α_i, we obtain the following:

$$\frac{\partial \mathcal{C}'_{i,np}}{\partial \alpha_i} = -b_i \beta_i d_i e^{\beta_i (D_i(\phi_i) - \alpha_i d_i)} D_i(\phi_i) < 0$$

Therefore, it validates that $\mathcal{C}'_{i,np}$ decreases as α_i increases.

Moreover, from equation 7.4, we can conclude that the increasing rate of $\mathcal{C}'_{i,np}$ is greater as α_i is becoming smaller given the same D_i. Thus, the smaller α_i is, the more aggressive the nonlinear pricing scheme is. We also observe that the crossing point that nonlinear pricing function over linear counterpart is right-shifted as α_i increases in Figure 7.2. And this crossing point gives the helpful insight for the network provider to leverage the delay-incurred cost against the desire delay requirement by tuning the proactive factor, α_i.

Due to the multiplexing nature for the shared resources in this system, a deterministic delay bound is beyond our control without an appropriate admission control and traffic shaping mechanism. Thus, the stochastic nature of the delay bound is inherent for our proposed MBORA system. In order to prevent QoS violations due

to a possible fluctuation of delays in the neighborhood of the desired delay bound, the network provider can choose a smaller value of α_i, less than 1. In this way, the more aggressive nonlinear delay-incurred cost function is applied in the optimization problem, that in turn results in higher bandwidth allocation. Overall, the value of α_i determines the degree that the network provider is willing to protect/guarantee QoS constrains for class i.

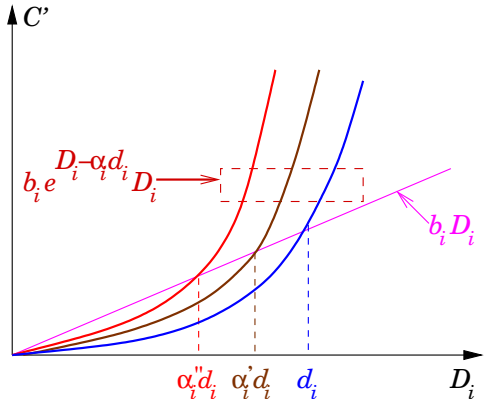


Figure 7.2: Elasticity analysis on α_i, where $\alpha_i'' < \alpha_i' < 1$ and $\beta_i = 1$.

Next, we analyze the elasticity of the progressive factor β_i. Assuming $\alpha_i = 1$, we can plot a group of nonlinear cost functions by varying β_i, against a linear one as shown in Figure 7.3.

Since α_i is fixed, we can see that all three nonlinear pricing based cost functions cross over the linear function at the same point, $D_i = d_i$. With the analogous approach, we can obtain the first derivative of $\mathcal{C}'_{i,np}$ with respect to β_i as follows:

$$\frac{\partial \mathcal{C}'_{i,np}}{\partial \beta_i} = b_i(D_i(\phi_i) - \alpha_i d_i)e^{\beta_i(D_i(\phi_i)-\alpha_i d_i)}D_i(\phi_i)$$

Therefore, it is easy to conclude that $\mathcal{C}'_{i,np}$ decreases as β_i increases when $D_i(\phi_i) < \alpha_i d_i$; while $\mathcal{C}'_{i,np}$ increases as β_i increases when $D_i(\phi_i) > \alpha_i d_i$. And this property is reflected in Figure 7.3 with $\alpha_i = 1$.

For each class i, the β_i factor reflects the fact that system is charged less for satisfying the QoS requirements, while it is charged more for violating them. From

the other perspective, if we map the value of β_i to the priority of class i, β_i can be interpreted as the aggressive factor in the nonlinear pricing function. The larger β_i is, the higher priority the class will be given. Therefore, the higher priority class can be marked with a bigger value of β_i to demonstrate its aggressiveness. In such a scheme, the network provider can have a low delay-incurred cost for QoS guarantee to the high priority class, and risk the high penalty involved for violation at the same time.

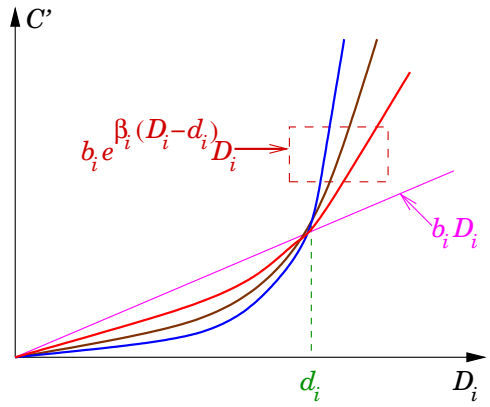


Figure 7.3: Elasticity analysis on the different values of β_i with $\alpha_i = 1$.

Our analysis shows how to control the combination of α_i and β_i. Given the priority of the class, we can choose the appropriate α_i and β_i in order to realize the corresponding level of the aggressiveness in the nonlinear pricing scheme and stochastic concerns in terms of QoS guarantees.

7.4.2 Complexity Analysis

In this subsection, we will explore the computational complexity for our proposed solution strategy, given a system with n classes.

As we discussed in the previous section, the simple solution approach attempts to calculate and evaluate the optimal solutions for up to 2^n cases, that leads to its computation complexity to be $O(2^n)$.

In our proposed solution strategy, the advantage is to eliminate this dimensionality problem by considering D_i as the whole function together. Moreover, for Quasi-

Newton method adopted in our approach strategy, it only requires roughly $O(n^2)$ in general, compared with $O(n^3)$ of Newton method [7]. It is a huge drop in computation complexity, in contrast with the one of the simple solution approach.

Therefore, our proposed solution strategy is more friendly for online adaptive algorithms, that can be easily implemented in the real-time environment for the access networks.

7.5 Performance Evaluation

In this section, a comprehensive numerical study of all the parameters involved in our pricing schemes is undertaken, in order to understand they affect the optimal allocation rates, given the various scenarios. Besides the analytical study in our approach, we also attempt to interpret it from the engineering perspective.

For simplicity, we assume that two classes of traffic share an access node (or edge point) in the access networks, with the link having capacity of 10 Mbps. It is accordingly two-queue system and class 1 is designated with higher priority than class 2. Also, the desired delays are given by ($d_1 = 5$, $d_2 = 10$) ms. Under the adaptive and optimal scheduling scheme introduced in the previous chapters, we set the adaptive window, $W = 1$ ms.

For clarity purposes, we will discuss two acronyms for the important variables used in our case study. Given the initial queue length of class i, we can obtain the initial delay at best scenario if the whole link capacity is devoted to serve the queue, that can be calculated by q_i^0/C. By dividing this initial delay with the desired delay of class i, d_i, we define this result as initial projected delay ratio, that can be denoted as our first acronym, IPDR. This ratio is used to describe the initial delay latency at the best scenario compared with the desired delay for the same class i. The larger this ratio is, the higher possibility of QoS violation is in delay respect. Therefore, we attempt to designate IPDR with the level of QoS violation tendency. If $\text{IPDR}_\text{i} \geq 0.5$, we define it as 'High', otherwise we tag it as 'Low', for class $i = 1, 2$.

Similarly, we can define EGRR as the acronym of the variable, estimated gen-

eralized rate ratio, calculated by the estimated generalized rate of class i over the link capacity, r_i/C. The larger the ratio is, the higher traffic volume is expected for the next adaptive window. In our numerical study, we tag this ratio as 'High' if $\mathrm{EGRR_i} \geq 0.5$, otherwise we interpret it as 'Low', for class $i = 1, 2$. Furthermore, when $\sum_i \mathrm{EGRR_i} > 1$, the system can be described as under-provisioned, otherwise, it can be interpreted as over-provisioned.

Under the described architecture, we evaluate the following metrics under both the linear and nonlinear pricing schemes: the maximum of the objective function, the optimal allocation, ϕ^*, and the analytical expected delay, D using the Matlab platform. And these evaluation are studied based on the solo effect of p, b, α and β, or the combination of them. It should be noticed that p is the unit price charged per Mbps and b is the unit price charged per ms for the classes.

Before the case studies for the above parameters, we want to give the pictorial view of the objective function $f(\phi_1, \phi_2)$ vs. (ϕ_1, ϕ_2), with the global maximizer around the region of $(\phi_1 = 0.5, \phi_2 = 0.5)$ in Figure 7.5. Clearly, the arch shape of the objective function demonstrates its joint concavity in the plot.

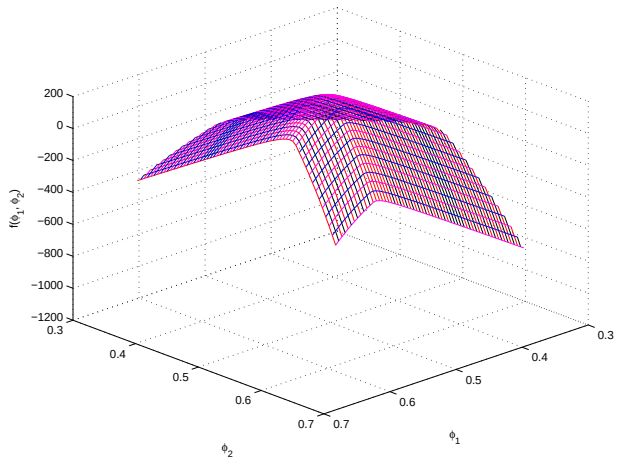


Figure 7.4: The objective function $f(\phi_1, \phi_2)$ vs. (ϕ_1, ϕ_2) with the global maximizer around the region of $(\phi_1 = 0.5, \phi_2 = 0.5)$.

7.5.1 Case Study I – The Effect of p

We begin our case study with the effect of $p(p_1, p_2)$ and discuss it under the two pricing models considered in this chapter. In order to evaluate the effect of p only, we assume the equivalent conditions for all the other parameters, such as ($b_1 = 1$, $b_2 = 1$) and ($\text{IPDR}_1 = 0.2$, $\text{IPDR}_2 = 0.1$)$\sim$ (Low, Low).

Linear Pricing Model

First, by fixing $p_2 = 1$, we investigate the effect Δp_1 on the optimal allocation of ϕ^*, as given in table 7.1. In this table, we list the corresponding optimal solutions by doubling the price of p_1 under the different combinations of EGRR for two classes.

Under a (Low, Low) pattern for EGRR, we can find a huge gain in ϕ_1^*, about 25% more bandwidth, for class 1 when p_1 is doubled from 1 to 2, while the gain of ϕ_1^* becomes smaller, and even lower than 1%, although the price of p_1 continues to be doubled. Meanwhile, we can observe similarly increasing tendencies for ϕ_1^* under (Low, High) and (High, Low) patterns for EGRR, although the gain in ϕ_1^* is relatively smaller upon each double price of p_1, compared with the case of (Low, Low) pattern. Finally, for the last case with (High, High) pattern for EGRR, the gain of ϕ_1^* is consistently small for each double price of p_1.

Based on these observations, we can conclude the following: First, for the first three cases, the optimal allocations are mainly determined by the estimated generalized rates for the classes with ($p_1 = 1$, $p_2 = 1$), since there is no price differentiation or bias between the classes. Second, the system is in the under over-provisioned state for these 3 cases and the price change of p_1 does give class 1 the leverage in the optimal allocation solution as p_1 doubles from 1 to 2. However, the more aggressive charging in p_1 (from 2 to 8) does not guarantee the equivalent or huge gain in the optimal bandwidth allocation, once the optimal solution reaches a saturation point. That is because our generalized profit model will guarantee the estimated generalized rates of all classes first when the system is in an over-provisioned state, while the rest of the bandwidth is shifted depending on the prices of the higher priority classes. Third,

the system is clearly in an under-provisioned state for the last case; hence, p does not make a difference in terms of bandwidth allocation, given the already saturated nature of the system and no delay differentiations between the classes in this setting are observed.

Table 7.1: The effect of Δp_1 on the optimal solutions for linear pricing model

	$(\text{EGRR}_1 = 0.3, \text{EGRR}_2 = 0.2) \sim$ (Low, Low)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.5357,0.4643)	(0.7842,0.2158)	(0.7909,0.2091)	(0.7940,0.2060)
	$(\text{EGRR}_1 = 0.2, \text{EGRR}_2 = 0.6) \sim$ (Low, High)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.3234,0.6766)	(0.3909,0.6091)	(0.3936,0.6064)	(0.3947,0.6053)
	$(\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.1) \sim$ (High, Low)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.7632,0.2368)	(0.8780,0.1220)	(0.8843,0.1157)	(0.8872,0.1128)
	$(\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.6) \sim$ (High, High)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.5192,0.4808)	(0.5202,0.4798)	(0.5221,0.4779)	(0.5259,0.4741)

Moreover, Figure 7.5 is drawn for 4 cases by changing p_1 from 0.6 to 1.5, in order to further understand the relationship between p_1 and p_2. Besides the (High, High) case, we can observe that ϕ_1^* stays close to its EGRR_1 when $p_1 < p_2$. On the other hand, there is a dramatic bandwidth shift once p_1 is larger than p_2, then ϕ_1^* reaches the saturation due to the limit of capacity and guarantee for EGRR_2 in the system. And the (High, High) case shows no difference along the changes as expected. Figure 7.5 further strengthens our understanding of the effect of p under the linear pricing model.

Finally, given the (Low, Low) scenario for $(\text{EGRR}_1, \text{EGRR}_2)$, we plot how the optimal ϕ_1^* changes with (p_1, p_2) in Figure 7.6(a) and the maximum value of the objective function varies against (p_1, p_2) in Figure 7.6(b).

Figure 7.6(a) can be regarded as a 3D extension of Figure 7.5. It demonstrates that the huge gain in ϕ_1^* happens only if $p_1 > p_2$ for an over-provisioned system. In

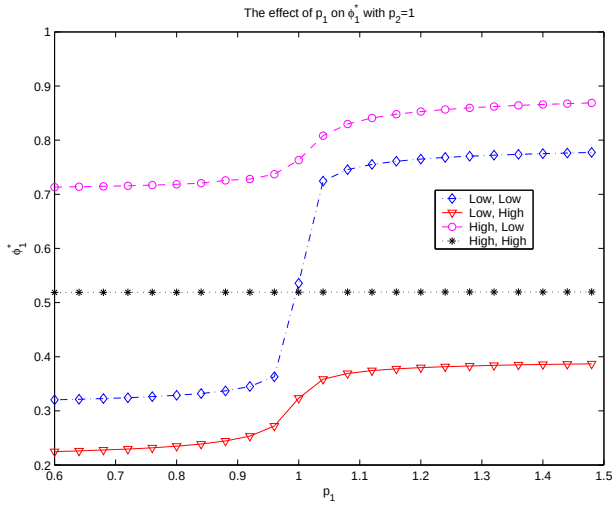


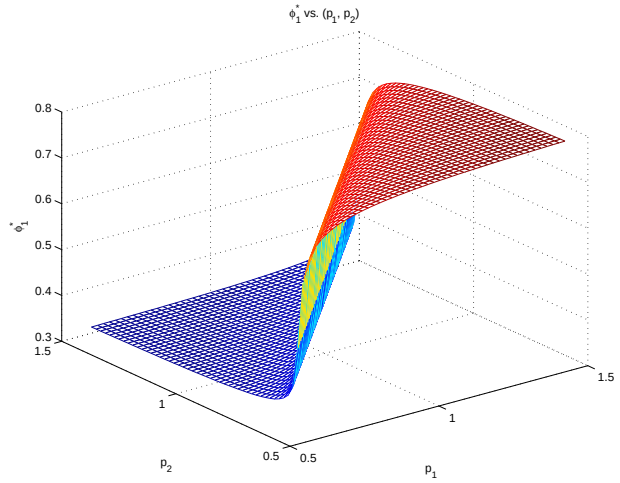
Figure 7.5: The effect of p_1 on ϕ_1^* with $p_2 = 1$ for linear pricing model.

this case, the surplus bandwidth tends to switch over to the class with the higher price charge on the bandwidth. Figure 7.6(b) demonstrates that the maximum profit of the objective function for the linear pricing model monotonically increases as p_1 and p_2 increase.

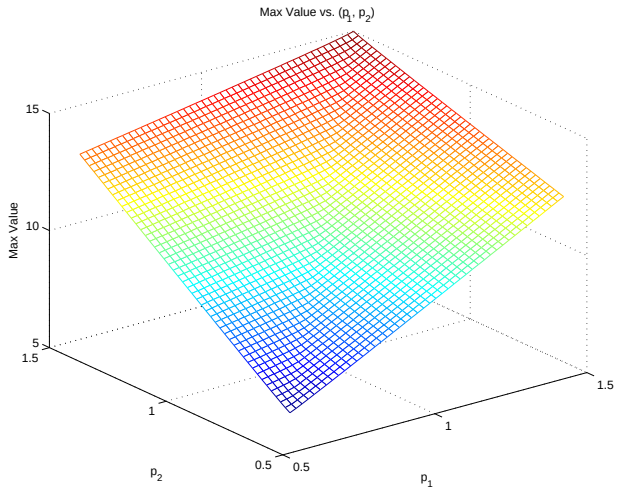
Nonlinear Pricing Model

Next, we discuss the effect of p under the nonlinear pricing model, given the same settings as for the linear case. In addition, we set $(\alpha_1 = 1, \alpha_2 = 1)$ and $(\alpha_1 = 1, \alpha_2 = 1)$ in order to investigate the solo effect of p under this nonlinear pricing scheme by weakening the effect of the exponential component in the nonlinear pricing model.

We obtain the results of the optimal solutions in table 7.2 as p_1 doubles. As expected, ϕ_1^* shows very similar trends for all four cases as the ones under the linear pricing model. There is only a slight difference for ϕ_1^* for a certain value of (p_1, p_2), compared with the one under the linear pricing model. However, this slight difference is due to the addition of the exponential component in the nonlinear pricing model, even though its effect is lessened in the present setting.



(a) ϕ_1^* vs. (p_1, p_2)



(b) Maximum value vs. (p_1, p_2)

Figure 7.6: (a) ϕ_1^* and (b) maximum value vs. (p_1, p_2) given (Low, Low) type for $(\text{EGRR}_1, \text{EGRR}_2)$ and $(b_1 = 1, b_2 = 1)$.

Table 7.2: The effect of Δp_1 on the optimal solutions for nonlinear pricing model

	$(\text{EGRR}_1 = 0.3, \text{EGRR}_2 = 0.2) \sim$ (Low, Low)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.5383,0.4617)	(0.7850,0.2150)	(0.7913,0.2087)	(0.7943,0.2057)
	$(\text{EGRR}_1 = 0.2, \text{EGRR}_2 = 0.6) \sim$ (Low, High)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.3246,0.6754)	(0.3913,0.6087)	(0.3950,0.6050)	(0.3967,0.6033)
	$(\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.1) \sim$ (High, Low)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.7642,0.2358)	(0.8790,0.1210)	(0.8877,0.1123)	(0.8919,0.1081)
	$(\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.6) \sim$ (High, High)			
$p(p_1, p_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.5360,0.4640)	(0.5360,0.4640)	(0.5361,0.4639)	(0.5361,0.4639)

Similarly, Figure 7.7 depicts the effect of p_1 on ϕ_1^* for the nonlinear pricing model. It obviously shows almost the same tendency on ϕ_1^* for all 4 cases.

In all, we can conclude the following: First, p has the similar effect on the optimal solution for both linear and nonlinear pricing models; Second, in the over-provisioned case, the distribution of the surplus bandwidth will be in favor of the higher-paying classes; Third, simple aggressive charging on bandwidth can not guarantee the corresponding return in the bandwidth due to the limit of the link capacity and multiplexing nature in this system; Fourth, p is not a good factor to use for allocation differentiation in the under-provisioned case.

7.5.2 Case Study II – The Effect of b

In this subsection, we continue our investigation and analysis of the effect of $b(b_1, b_2)$ under the two pricing models. In order to evaluate b's effect only, we assume similar conditions for all the other parameters, such as ($p_1 = 1$, $p_2 = 1$) and ($\text{EGRR}_1 = 0.2$, $\text{EGRR}_2 = 0.2$)$\sim$ (Low, Low).

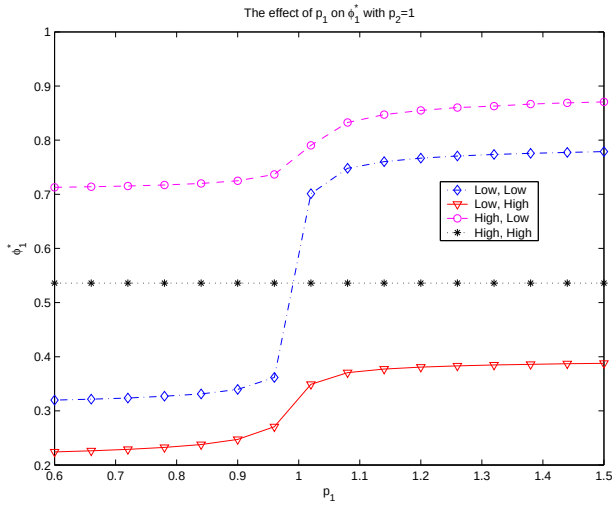


Figure 7.7: The effect of p_1 on ϕ_1^* with $p_2 = 1$ for nonlinear pricing model.

Linear Pricing Model

Adopting the same approach as we did in the p case, we begin with the analysis about the effect Δb_1 on the optimal allocation of ϕ^* by fixing $b_2 = 1$. In table 7.3, we give the corresponding optimal solutions by doubling the price of b_1 under the different combinations of IPDR for two classes.

Under a (Low, Low) pattern for IPDR, class 2 obtains more bandwidth allocation than class 1 when $(b_1 = 1, b_2 = 1)$. That is because the system tends to allocate more bandwidth to class 2, the one with the larger initial queue size, given the same delay-incurred unit pricing for two classes. But as b_1 doubles, the delay differentiation is activated and we can observe that ϕ^* increases consistently. For the rest of cases, we find the similar increasing tendencies of ϕ^* as b_1 doubles, even though the pattern of IPDR changes.

Next, Figure 7.5.2 shows how ϕ_1^* reacts to the change of b_1 from 1 to 20 for all 4 cases, with the fixed $b_2 = 1$ and (Low, Low) type for $(\text{IPDR}_1, \text{IPDR}_2)$. As we discussed in table 7.3, it depicts the similar increasing trends of ϕ_1^* for all 4 cases as

Table 7.3: The effect of Δb_1 on the optimal solutions for linear pricing model

	$(\text{IPDR}_1 = 0.2, \text{IPDR}_2 = 0.2) \sim$ (Low, Low)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.4168,0.5832)	(0.4578,0.5422)	(0.4999,0.5001)	(0.5422,0.4578)
	$(\text{IPDR}_1 = 0.2, \text{IPDR}_2 = 1) \sim$ (Low, High)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.2736,0.7264)	(0.2964,0.7036)	(0.3239,0.6761)	(0.3559,0.6441)
	$(\text{IPDR}_1 = 1, \text{IPDR}_2 = 0.2) \sim$ (High, Low)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.6081,0.3919)	(0.6440,0.3560)	(0.6761,0.3239)	(0.7035,0.2965)
	$(\text{IPDR}_1 = 0.8, \text{IPDR}_2 = 0.9) \sim$ (High, High)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.4037,0.5963)	(0.4437,0.5563)	(0.4856,0.5144)	(0.5279,0.4721)

b_1 increases. It should be noted that the system is in an over-provisioned state for all 4 cases; thus, the optimal allocation of the bandwidth is more affected by the initial queue size of the class and the delay-incurred pricing differentiation involved. With the higher price of b_1, class 1 can benefit more from the optimal bandwidth allocation by the system.

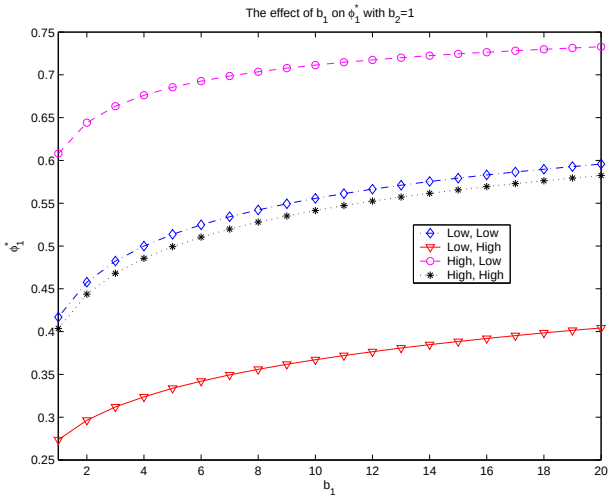


Figure 7.8: The effect of b_1 on ϕ_1^* with $b_2 = 1$ for linear pricing model.

Finally, given a (Low, Low) scenario for $(\text{IPDR}_1, \text{IPDR}_2)$, we plot how the optimal

ϕ_1^* changes upon (b_1, b_2) in Figure 7.9(a) and the maximum value of the objective function varies against (b_1, b_2) in Figure 7.9(b).

Figure 7.9(a) demonstrates that ϕ_1^* increases as b_1 increases and b_2 decreases. The difference between b_1 and b_2 is becoming bigger, the more bandwidth allocation is distributed towards class 1. In Figure 7.6(b), we can observe that the maximum profit of the objective function for the linear pricing model decreases as both b_1 and b_2 increase.

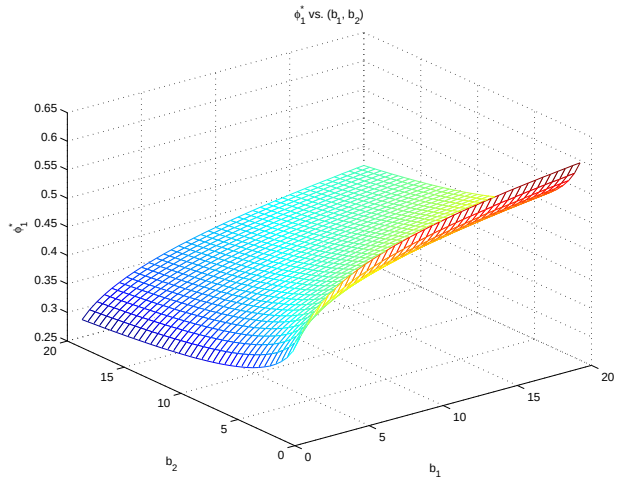
Nonlinear Pricing Model

We continue our discussion of the effect of b under the nonlinear pricing model, given the same settings as in the linear case. As before, we set $(\alpha_1 = 1, \alpha_2 = 1)$ and $(\alpha_1 = 1, \alpha_2 = 1)$ in order to evaluate the solo effect of b under this nonlinear pricing scheme.

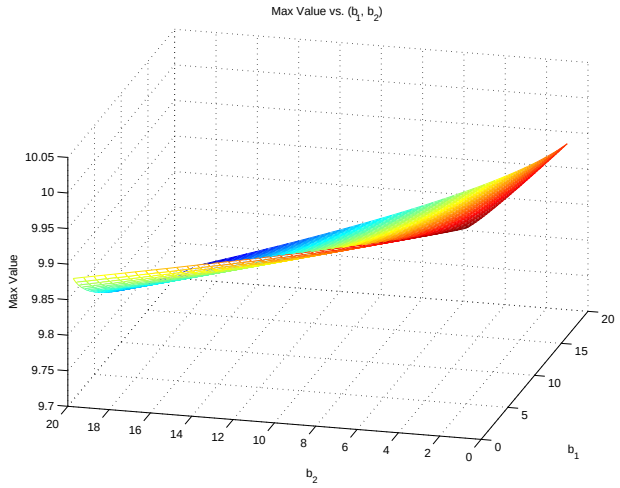
In the similar approach, we obtain the results of the optimal solutions in table 7.4 as b_1 doubles. Again, ϕ_1^* increases in similar ways for all four cases as in the linear pricing model. The slight difference of ϕ_1^* between the linear and nonlinear pricing models is observed, given the certain value of (b_1, b_2). It is still derived from the addition of the exponential component in the nonlinear pricing model.

Table 7.4: The effect of Δb_1 on the optimal solutions for nonlinear pricing model

	$(\text{IPDR}_1 = 0.2, \text{IPDR}_2 = 0.2) \sim$ (Low, Low)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.4198,0.5802)	(0.4608,0.5392)	(0.5031,0.4969)	(0.5452,0.4548)
	$(\text{IPDR}_1 = 0.2, \text{IPDR}_2 = 1) \sim$ (Low, High)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.2750,0.7250)	(0.2982,0.7018)	(0.3260,0.6740)	(0.3582,0.6418)
	$(\text{IPDR}_1 = 1, \text{IPDR}_2 = 0.2) \sim$ (High, Low)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.6108,0.3892)	(0.6465,0.3535)	(0.6783,0.3217)	(0.7053,0.2947)
	$(\text{IPDR}_1 = 0.8, \text{IPDR}_2 = 0.9) \sim$ (High, High)			
$b(b_1, b_2)$	(1 , 1)	(2 , 1)	(4 , 1)	(8 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.4063,0.5937)	(0.4465,0.5535)	(0.4884,0.5116)	(0.5307,0.4693)



(a) ϕ_1^* vs. (b_1, b_2)



(b) Maximum value vs. (p_1, p_2)

Figure 7.9: (a) ϕ_1^* and (b) maximum value vs. (b_1, b_2) given (Low, Low) type for $(\text{IPDR}_1, \text{IPDR}_2)$ and $(b_1 = 1, b_2 = 1)$.

Moreover, the effect of b_1 on ϕ_1^* for the nonlinear pricing model is demonstrated in Figure 7.10. As in the linear case, similar increasing patterns are shown for all 4 cases.

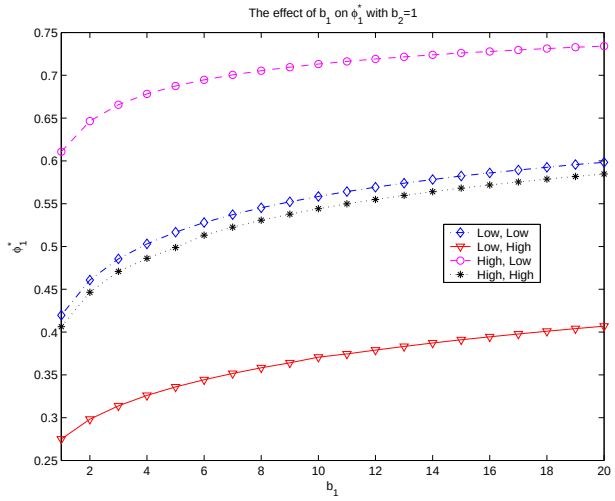


Figure 7.10: The effect of b_1 on ϕ_1^* with $b_2 = 1$ for nonlinear pricing model with $(\alpha_1 = 1, \alpha_2 = 1)$ and $(\beta_1 = 1, \beta_2 = 1)$.

Overall, we conclude that the increase of b for a certain class also increases its competitiveness for the shared resources in the system. Further, it will lead to better sharing of bandwidth for the class and a better delay performance. To some extent, it achieves the differentiation among the classes from the delay perspective.

7.5.3 Case Study III – The Effect of α

In this subsection, we will discuss the effect of α under the nonlinear pricing model. In its initial setting, we assume that $(p_1 = 1, p_2 = 1)$, $(b_1 = 1, b_2 = 1)$, $(\beta_1 = 1, \beta_2 = 1)$ and (Low, Low) type for $\text{IPDR}_1, \text{IPDR}_2$. Furthermore, we set the desired delays at 5 ms for both classes, so that we can more accurately evaluate the solo effect of α.

For clarification, our discussion is classified into two cases: the over- and under-provisioned ones.

Over-provisioned Case

By fixing $\alpha_2 = 1$, we first analyze the effect of $\Delta\alpha_1$ on the optimal allocation of ϕ^* and expected delay, D, for the over-provisioned case.

In table 7.5, we give the corresponding optimal solutions and expected delays as α_1 decreases. It is not difficult to see that the optimal allocation ϕ_1^* slightly increases and D_1 slightly decreases as α_1 decreases. Under this setting, the delay should be relatively small for the over-provisioned case. And considering it together with the small value in β, α effect is neutralized to some extent so that the exponential component of nonlinear pricing model is not that effective.

Table 7.5: The effect of $\Delta\alpha_1$ on ϕ^* and D (measured in ms) for over-provisioned case

	$(\text{EGRR}_1 = 0.4, \text{EGRR}_2 = 0.3) \sim$ (Low, Low)			
$\alpha(\alpha_1, \alpha_2)$	(1 , 1)	(0.7, 1)	(0.4 , 1)	(0.1 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.5415,0.4585)	(0.5416,0.4584)	(0.5416,0.4584)	(0.5417,0.4583)
$D(D_1, D_2)$	(0.6524,0.6882)	(0.6521,0.6885)	(0.6518,0.6888)	(0.6515,0.6891)

We plot ϕ_1^* and the corresponding D_1 against α_1 for both models in Figure 7.11. Figure 7.11(a) shows that ϕ_1^* under the nonlinear model is decreasing as α_1 increases, but still is higher than its value under the linear pricing model. Accordingly, D_1 is increasing consistently as α_1 increases, but still lower than the one under the linear model. It should be noticed that the expected delays for both linear and nonlinear cases are well below the desired delays, since the system is in the over-provisioned state.

Under-provisioned Case

Next, we will discuss the effect of $\Delta\alpha_1$ for the under-provisioned case.

In table 7.6, we can observe that ϕ_1^* increases slightly as α_1 decreases. And expected delays for both classes are much bigger than their expected delays. This huge delay violation is inevitable due to the insufficient bandwidth in the system. Moreover,

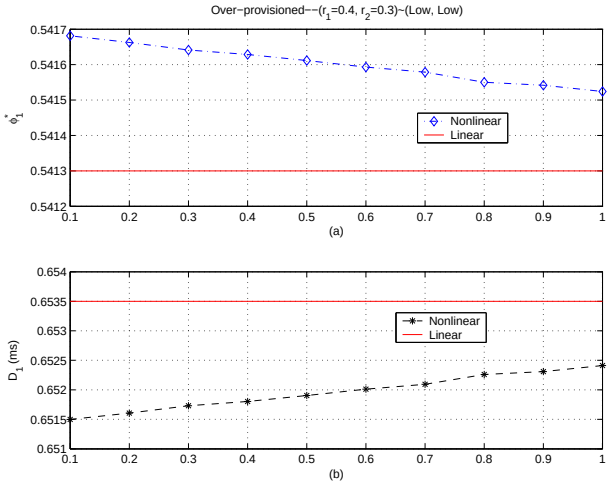


Figure 7.11: The effect of α_1 on (a) ϕ_1^* and (b) corresponding D_1 compared with linear pricing model for over-provisioned case, where ($\text{EGRR}_2 = 0.4, \text{EGRR}_2 = 0.3$)∼(Low, Low).

under this optimal allocation, we can see that the desired delay is almost negligible compared with the expected delay for class 1 and 2. Therefore, the exponential factors are more decided by β and the expected delays and that is why $\Delta\alpha_1$ has small effect on the optimal solutions. Finally, considering ($\beta_1 = 1, \beta_2 = 1$), ϕ_1^* and ϕ_2^* are close to be even since there is little differentiation in the delay-incurred cost.

Table 7.6: The effect of $\Delta\alpha_1$ on ϕ^* and D (measured in ms) for under-provisioned case

	($\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.6$) ∼ (High, High)			
$\alpha(\alpha_1, \alpha_2)$	(1 , 1)	(0.7, 1)	(0.4 , 1)	(0.1 , 1)
$\phi^*(\phi_1^*, \phi_2^*)$	(0.5268,0.4732)	(0.5270,0.4730)	(0.5271,0.4729)	(0.5272,0.4728)
$D(D_1, D_2)$	(166.24,136.15)	(166.09,136.30)	(165.94,136.46)	(165.80,136.62)

Figure 7.12 demonstrates the different patterns in ϕ_1^* and D_1 against α_1 for the nonlinear pricing model, as we discussed in table 7.6. We also observe that the ϕ_1^* for the nonlinear pricing model is larger than the one for the linear pricing model, which in turn results in a better D_1.

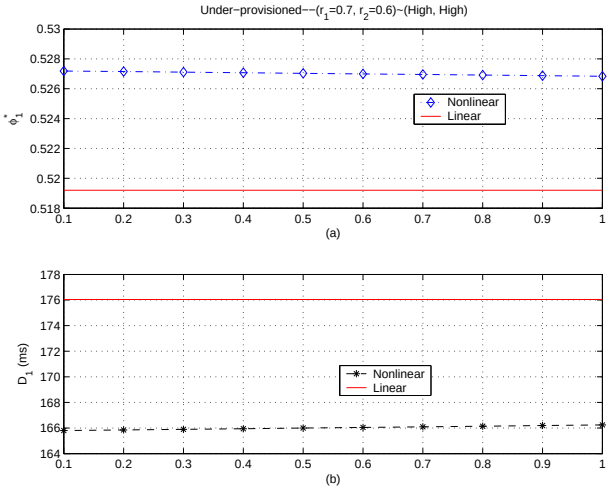


Figure 7.12: The effect of α_1 on (a) ϕ_1^* and (b) corresponding D_1 compared with linear pricing model for under-provisioned case, where $(\mathrm{EGRR}_1 = 0.7, \mathrm{EGRR}_2 = 0.6)\sim$(High, High).

Overall, we conclude that the decrease of α for one class leads to a higher bandwidth allocation to the class, no matter whether the system is under- or over-provisioned. Even thought its solo effect seems small, it will be certainly magnified by the combination with β, the one we discuss in the next subsection.

7.5.4 Case Study IV – The Effect of β

In this subsection, we will investigate the effect of another factor in our nonlinear pricing model, the aggressive factor β. Except setting $(\alpha_1 = 1, \alpha_2 = 1)$, we use the same set of parameters as the one for the study of α, in order to validate the solo effect of β more accurately. In the same way, we attempt to discuss the effect of β under two cases: over-provisioned and under-provisioned cases.

Over-provisioned Case

Let us start our study with β_1 effect on the optimal solutions and expected delays for over-provisioned case.

In table 7.7, we can find that ϕ_1 decreases as β_1 increases with $\beta_2 = 1$. Accordingly, the expected delay for class 1, D_1, is increasing slightly. As we can see, the analytical expected delays are well below the desired delays for both classes, since the system is capable to handle the incoming traffic. This will result in the negative sign for $D_i - \alpha_i d_i$, where $\alpha_i = 1$. As β_1 is becoming bigger in this scenario, the exponential factor of class 1 is getting even smaller, compared with the one of class 2. Therefore, our nonlinear pricing model tends to allocate higher bandwidth to the second class, in order to increase the provider's profit.

Table 7.7: The effect of $\Delta\beta_1$ on ϕ^* and D (measured in ms) for over-provisioned case

	$(\mathrm{EGRR}_1 = 0.4, \mathrm{EGRR}_2 = 0.3) \sim$ (Low, Low)			
β (β_1, β_2)	(1 , 1)	(13, 1)	(25 , 1)	(37 , 1)
ϕ^* (ϕ_1^*, ϕ_2^*)	(0.5415, 0.4585)	(0.540, 0.460)	(0.5385, 0.4615)	(0.5369, 0.4631)
D (D_1, D_2)	(0.652, 0.688)	(0.662, 0.679)	(0.6707, 0.6706)	(0.6799, 0.6622)

Then we plot the effect β_1 on ϕ_1^* and corresponding D_1 and also compare them with ones under linear pricing model for over-provisioned case, as shown in Figure 7.13. The degrading ϕ_1^* is observed as β_1 increases, and it is even worse than the one under linear pricing model after β_1 is larger than 3. Meanwhile, D_1 increases accordingly, but still well below the desired delay of 5ms. Thus, for the over provisioned case, the more aggressive β is, the more that class risks to give in the part of its share of bandwidth to the class with a less aggressive value of β.

Under-provisioned Case

Next, we continue our study on β effect for under-provisioned case.

In table 7.8, we can see the huge increase on ϕ_1^* and dramatic drop-down on D_1 as β increases; this is a marked difference from the results obtained for the over-provisioned case. Since the system can not handle the incoming traffic, the expected delays for both classes are extremely high. Now, the exponential factor of class i will have a huge effect since $D_i - \alpha_i d_i > 0$ becomes positive. With the increasing

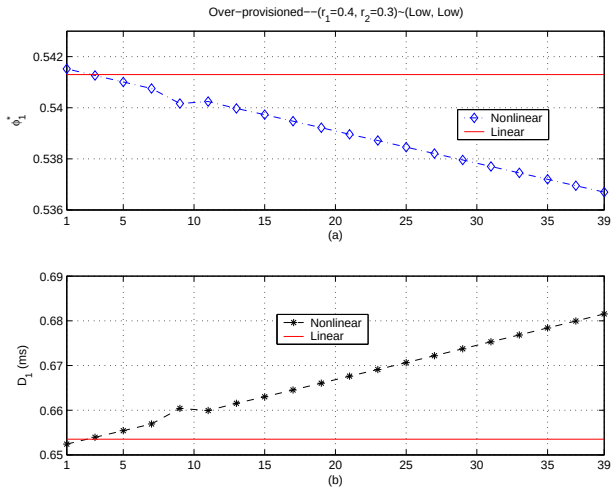


Figure 7.13: The effect of β_1 on (a) ϕ_1^* and (b) corresponding D_1 compared with linear pricing model for over-provisioned case, where $(\text{EGRR}_1 = 0.4, \text{EGRR}_2 = 0.3) \sim$(Low, Low).

aggressiveness of β_1, the system will switch more bandwidth to class 1 in order to lower its huge penalty cost caused by exponential factor. Finally, QoS and priority differentiation is achieved by the aggressive value of β, as the better service and QoS guarantee to class 1 in table 7.8 shows.

Table 7.8: The effect of $\Delta\beta_1$ on ϕ^* and D (measured in ms) for under-provisioned case

	$(\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.6) \sim$ (High, High)			
β (β_1, β_2)	(1 , 1)	(13, 1)	(25 , 1)	(37 , 1)
ϕ^* (ϕ_1^*, ϕ_2^*)	(0.527, 0.473)	(0.622, 0.378)	(0.646, 0.354)	(0.701, 0.299)
D (D_1, D_2)	(166.24, 136.15)	(64.60 295.63)	(43.29, 350.43)	(0.71, 506.69)

In Figure 7.14, we can see the consistent increase of ϕ_1^* and decrease in D_1 as β_1 increases. Around the region of $\beta_1 = 27$, we can observe even a dramatic jump in ϕ_1^*, then its increase reaches saturation after that value. That is because after that point, the expected delay of class is well controlled under the desired delay and the exponential factor of class 1 is no more effective. Thus, the system is not rewarded with more profit after that critical point.

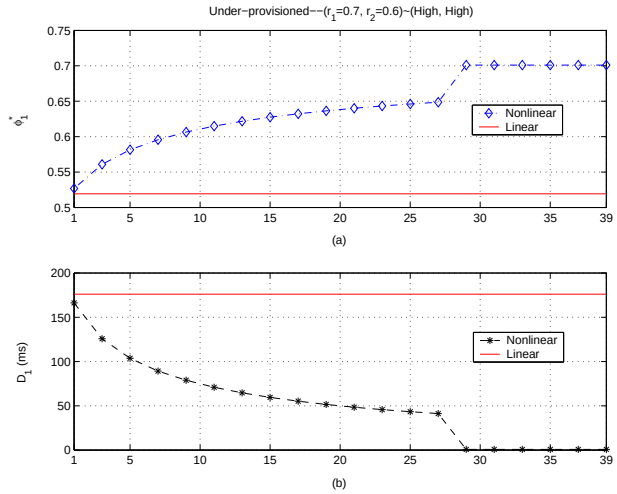


Figure 7.14: The effect of β_1 on (a) ϕ_1^* and (b) corresponding D_1 compared with linear pricing model for under-provisioned case, where $(\mathrm{EGRR}_1 = 0.7, \mathrm{EGRR}_2 = 0.6)\sim$(High, High).

Overall, we have the following general remarks: First, the aggressiveness of β can degrade QoS performance a little but still satisfy QoS requirements when the system is over-provisioned. Second, this aggressiveness can also reward the class with more bandwidth and better QoS performance when the system is under-provisioned. Third, it can use empirical method to set the appropriate value of β in order to guarantee QoS and realize differentiation between classes. Fourth, the network provider can also make β adaptive, that is to make it small when the sum of EGRR shows the system is over-provisioned and make it big when the sum of EGRR shows the system is under-provisioned. In this way, the high priority class can always be guaranteed to obtain better service.

7.5.5 Case Study V – The Joint Effects of the α and β Parameters

After studying the individual effects of α and β, we investigate next their combined effect. By fixing $(p_1 = 1, p_2 = 1)$, $(b_1 = 1, b_2 = 1)$, we will discuss one case example,

that is (High, Low) pattern for ($\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.4$) when the system is under-provisioned.

Figure 7.15 to Figure 7.17 demonstrate the combo effect of α_1 and β_1 on the maximum value, ϕ_1^*, and D_1 respectively.

In Figure 7.15, the maximum value increases as β_1 increases. With the same β_1, it increases as α_1 decreases. This combination effect further strength our previous judgement on α and β.

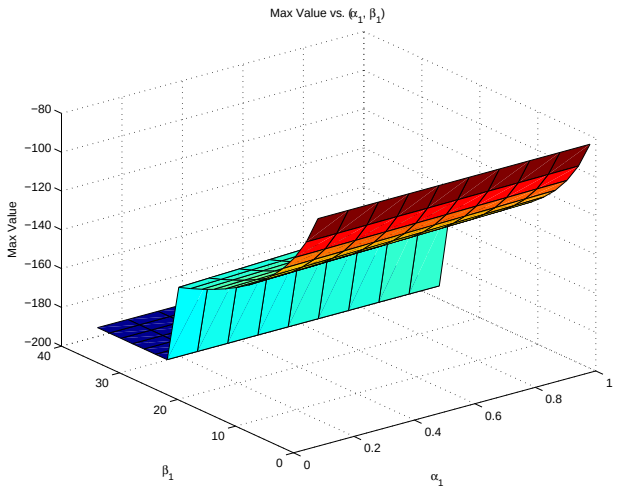


Figure 7.15: An example of the combo effect of α_1 and β_1 on the maximum value for under-provisioned case, where ($\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.4$)$\sim$(High, Low).

Figure 7.17 shows ϕ_1^* changes at the same rate as the maximum value, thus validating the result from a different angle. In Figure 7.17, D_1 varied in the reverse trend, contrast with one of ϕ_1^*. Overall, we conclude that the combination of α and β achieves the better QoS performance than the individual effect of either factor.

7.5.6 Case Study VI – The Comparison Between Linear and Nonlinear Pricing Models

Finally, after understanding the behavior and the impact of all the parameters involved in the linear and nonlinear pricing models, we compare them with respect

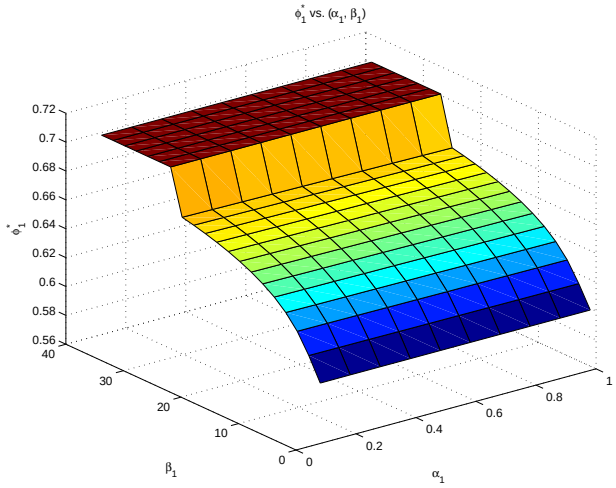


Figure 7.16: An example of the combo effect of α_1 and β_1 on ϕ_1^* for under-provisioned case, where $(\text{EGRR}_1 = 0.7, \text{EGRR}_2 = 0.4)\sim$(High, Low).

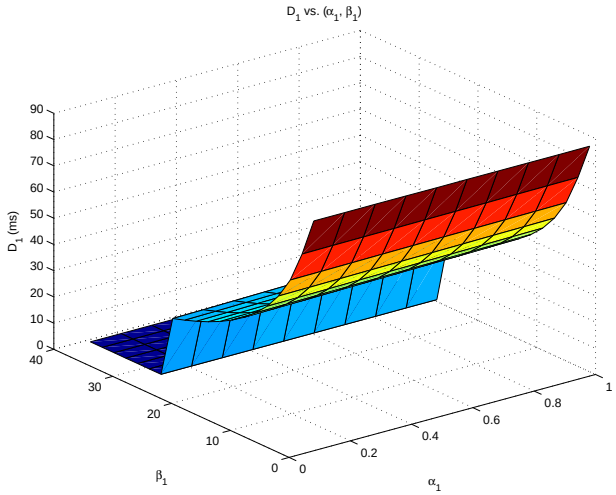


Figure 7.17: An example of the combo effect of α_1 and β_1 on D_1 for under-provisioned case, where $(\text{EGRR}_1 = 0.7, \text{EGRR}_1 = 0.4)\sim$(High, Low).

to maximum profit, optimal allocation and the expected delays.

For sake of comparison, we set $(p_1 = 1.2, p_2 = 1)$ and $(b_1 = 2, b_2 = 1)$ for both linear and nonlinear pricing models. In addition, $(\alpha_1 = 0.8, \alpha_2 = 1)$ and $(\beta_1 = 30, \beta_2 = 1)$ are set for the nonlinear pricing model. By changing the pair of $(\mathrm{EGRR}_1, \mathrm{EGRR}_2)$, we can evaluate maximum profit, the optimal allocation and the expected delays from under-provisioned and over-provisioned cases.

In Figure 7.18, we can observe a similar maximum profit for both pricing models when the system is over-provisioned, but actually the maximum profit of the nonlinear one is slightly larger. However, the maximum value under the nonlinear pricing model is dramatically greater than the one for the linear pricing model. That is because the exponential effect of nonlinear pricing model is in charge and makes the differentiation between the classes when the system is over-provisioned.

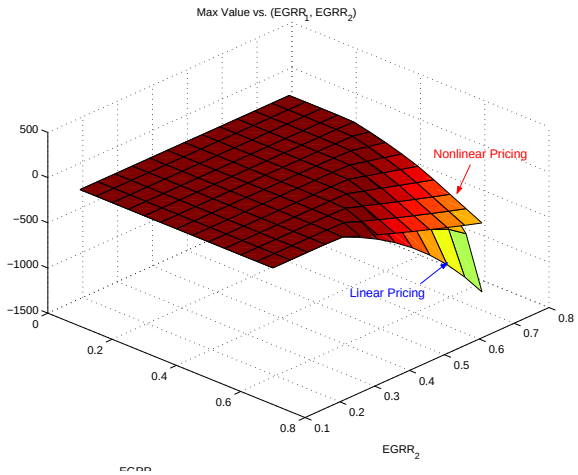


Figure 7.18: The comparison of the maximum value for objective function between linear and nonlinear pricing models.

Figure 7.19 also shows the similar value of ϕ_1^* under both pricing models for over-provisioned case, while demonstrates the huge difference when the system is under-provisioned. That is nonlinear pricing model gives the favor to higher priority of class 1 in the case that the system is short of resources.

In Figure 7.19, D_1 values are close for both pricing models for over-provisioned case while the nonlinear one demonstrates much lower than the linear one when the

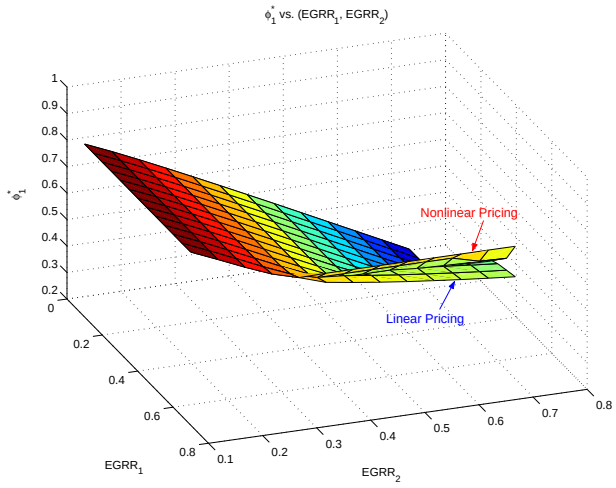


Figure 7.19: The comparison of ϕ_1^* between linear and nonlinear pricing models.

system is under-provisioned.

Overall, we can conclude that when the system is over-provisioned, the nonlinear pricing model can show a slightly better QoS performance and service; it demonstrates much better QoS guarantee and differentiation when the system is under-provisioned. Therefore, differentiation and priority are achieved by the appropriate pricing schemes.

7.6 Conclusion

In this chapter, we investigated a new nonlinear pricing model that is sensitive to the desired delay requirement of each class. Under this pricing model, the network provider has more flexibility in optimal control of the shared resources. Moreover, computationally complexity issues were successfully addressed by establishing the global concavity of the objective function as a function of the delay.

The numerical evaluation shows that the nonlinear pricing model guarantees better QoS performance compared to its linear counterpart, especially when the system is under-provisioned. Also, some interesting observations on tuning parameters give

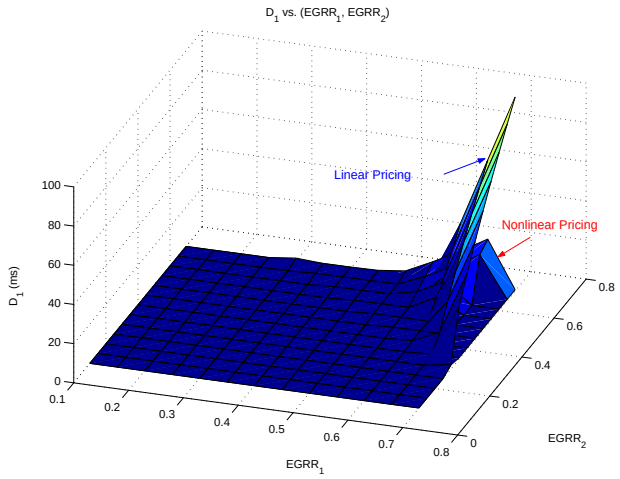


Figure 7.20: The comparison of D_1 between linear and nonlinear pricing models.

the insights on the implementation from the engineering perspective.

Chapter 8

Concluding Remarks

The evolution of network technology will ultimately touch every respect of human life, that drives the network provider or carrier to provide a variety of services for end users who access the networks. In such multi-service networks, QoS provisioning and pricing schemes are important issues for researchers to investigate. In this dissertation, a efficient approach addressing these issues has been proposed and analytically and numerically investigated.

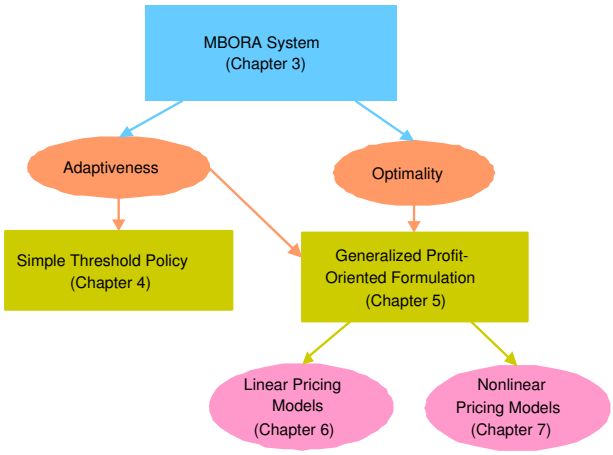


Figure 8.1: The evolution of our research and contributions.

In Figure 8.1, we demonstrate the evolution of our research and contributions during the course of study.

Based on the modeling of multi-service networks, we classify the end users into two major categories: the classes that demand deterministic delay-bound and the classes that require a flexible delay-bound. Accordingly, we proposed a generalized adaptive and optimal control framework to differentiate the resource allocation for these two categories. After excluding the deterministic delay-bound classes that require strict bandwidth reservation, we focus on the adaptive and optimal control for the flexible delay-bounded classes that are multiplexed using shared resources. Therefore, the resource allocation is in fact delivered by the subsystem of this generalized framework, measurement-based optimal resource allocation (MBORA) system. For the rest of the dissertation, we attempt to evaluate this system in terms of its adaptiveness and optimality regarding resource allocation.

Then we validated the benefits of the adaptivity of our proposed framework, based on a simple threshold decision policy and measurement estimations. Under this scenario, we also employed WRR and WFQ as the scheduling algorithms to execute the resource allocation. After extensive simulations, we concluded that Dynamic WFQ achieves the best performance, in contrast to its static counterpart.

In the next phase of our research, we formulated a generalized *profit center* optimization model in order to address the optimal control component of our proposed framework. Besides the delivery of QoS, this model also provided the network provider with a reference measure in terms of profit, while distributing the resources to the classes.

Using generalized service models, we investigated a profit-oriented formulation, specifically under a linear pricing model, subject to average queue delay constraints. A fast, low complexity algorithm was developed accordingly, for the online dynamic resource allocation problem. The excellent performance of this approach was validated through simulation experiments under different traffic scenarios.

Finally, we proposed a delay-sensitive nonlinear pricing model for the generalized profit-oriented formulation. Its effect on the objective function and the optimal solution were analyzed. Moreover, we successfully solved the dimensionality problem

we encountered previously, by proposing a generalized solution strategy valid for all linear, nonlinear or mixed pricing models. An extensive numerical study of our strategy's performance demonstrated that the nonlinear pricing model is superior to the linear one, especially when the system is under-provisioned.

Overall, it was established that the proposed framework provides an effective and efficient mechanism, that can be applied for online implementation in multi-service networks. Some issues that require additional research include the incorporation of other delay constraints (beyond the average delay case) and non-linear revenue schemes.

Bibliography

[1] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification. *IEEE Std. 802.11*, 1999.

[2] S. Keshav A. Demars and S. Shenker. Analysis and Simulation of a Fair Queuing Algorithm. In *ACM SIGCOMM*, pages 3–12, 1989.

[3] P. Abry and D. Veitch. Wavelet Analysis of Long-Range Dependent Traffic. *IEEE Trans. Info. Theory*, 44:2–15, 1998.

[4] B. Ryu and S. Lowen. Fractal Traffic Models for Internet Simulation. In *IEEE International Symposium on Computers Communications*, pages 271–370, France, 2000.

[5] J. Bennett and H. Zhang. WF^2Q: Worst-case Fair Weighted Fair Queueing. In *Proc. of IEEE INFOCOM*, pages 120–128, 1996.

[6] Jon C.R. Bennett and H. Zhang. Hierarchical Packet Fair Queueing Algorithms. *IEEE/ACM Trans. Networking*, pages 675–689, Oct 1997.

[7] Dimitri P. Bertsekas. *Nonlinear Programming.* Athena Scientific, second edition, 1999.

[8] Benny Bing. *Wireless Local Area Networks.* John Wiley and Sons, Ltd, 2002.

[9] Bo Ryu and Steve Lowen. Point Process Approaches to the Modeling and Analysis of Self-Similar Traffic: Part I – Model Construction. In *Proc. of IEEE INFOCOM'96*, pages 271–370, San Francisco, CA, March 1996.

[10] R. Buerin, S. Kamat, A. Orda, T. Przygienda, and D. Williams. QoS Routing Mechanisms and OSPF Extensions. *Work in Progress*, Internet Draft, March 1997.

[11] Richard L. Burden and J. Douglas Faires. *Numerical Analysis*. Brooks-Cole Publishing, seventh edition, 2001.

[12] Abhishek Chandra, Weibo Gong, and Prashant Shenoy. Dynamic Resource Allocation for Shared Data Centers Using Online Measurements. In *Proceedings of ACM/IEEE Intl Workshop on Quality of Service (IWQoS)*, pages 381–400, 2003.

[13] C. Courcoubetis and R. Weber. Buffer Overflow Asymptotics for a Switch Handling Many Traffic Sources. *Journal of Applied Probability*, 33:886–903, 1996.

[14] J. Heinanen et al. Assured Forwarding PHB Group. RFC 2597, June 1999.

[15] S. Blake et al. An Architecture for Differentiated Services. RFC 2475, Dec. 1998.

[16] V. Jacobson et al. An Expedited Forwarding PHB. RFC 2598, June 1999.

[17] K. Etemad. *CDMA 2000 Evolution*. John Wiley and Sons, Ltd, 2004.

[18] D. Ferrari and D. Verma. A Scheme for Real-Time Channel Establishment in Wide-Area Networks. *IEEE Journal on Selected Areas in Communications*, 8:368–379, April 1990.

[19] Sally Floyd. Comments on Measurement-based Admissions Control for Controlled-Load Services. 1996. To be submitted to Computer Communications Review.

[20] N. Fonseca, G. Mayor, and C. Neto. On the Equivalent Bandwidth of Self-Similar Sources. *ACM TOMACS*, 10:104–124, 2000.

[21] Frank Kelly. Notes on Effective Bandwidth. In F. P. Kelly, S. Zachary, and I. Ziedins, editors, *Stochastic Networks: Theory and Applications*, pages 141–168. Oxford University Press, 1996.

[22] Leonidas Georgiadis, Roch Guerin, Vinod Peris, and Kumar N. Sivarajan. Efficient Network QoS Provisioning based on Per Node Traffic Shaping. *IEEE/ACM Trans. Networking*, 4(4):482–501, August 1996.

[23] S. J. Golestani. A Self-Clocked Fair Queueing Scheme for Broadband Applications. In *Proc. of IEEE INFOCOM*, pages 636–646, 1994.

[24] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks. *IEEE J. Select. Areas Commun.*, 9(7):968–981, Sept. 1991.

[25] Q. Hao, S. Tartarelli, and M. Devetsikiotis. Self-Sizing and Optimization of High Speed Multiservice Networks. In *Proceedings of IEEE GLOBECOM*, volume 3, pages 1818–1823, 2000.

[26] J. B. Hiriart-Urruty and C. Lemarecal. *Convex Analysis and Minimization Algorithm I.* New York: Springer-Verlag, 1993.

[27] H. Holma and A. Toskala. *WCDMA for UMTS.* John Wiley and Sons, Ltd, 3rd edition, 2004.

[28] Shivkumar Kalyanaraman. Better-than-best-effort: Qos, int-serv, diff-serv, rsvp, rtp. ip2001-Lecture14-6pp.pdf, 2001.

[29] Carlos Alberto Kamienski. *An Architecture for Providing End-to-End QoS-based Advanced Services in the Internet.* PhD thesis, Universidade Federal de Pernambuco, Brazil, August 2002.

[30] F. P. Kelly. Charging and Rate Control for Elastic Traffic. *European Transactions on Telecommunications*, 8:33–37, 1997.

[31] F.P. Kelly, A.K. Maulloo, and D.K.H. Tan. Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability. *Journal of the Operational Research Society*, 49:237–252, 1998.

[32] E. Knightly and N. Shroff. Admission Control for Statistical QoS: Theory and Practice. *IEEE Network*, 13(2):20–29, 1999.

[33] Anjali Kulkarni and Yi-An Chen. QoS (Intserv and Diffserv). presentation.ppt, 2000.

[34] L. Breslau and S. Jamin, and S. Shenker. Comments on the Performance of Measurement-based Admission Control Algorithms. In *Proc. of IEEE INFOCOM*, pages 1233–1242, 2000.

[35] R. J. La and V. Anantharam. Utility-Based Rate Control in the Internet for Elastic Traffic. *IEEE/ACM Transactions on Networking*, pages 272–286, 2002.

[36] W.E. Leland and D.V. Wilson. High Time-Resolution Measurement and Analysis of LAN Traffic: Implications for LAN Interconnection. In *Proc. IEEE INFOCOM'91*, pages 1360–1366, Bal Harbor, FL, 1991.

[37] Yong Liu, Francesco Lo Presti, Vishal Misra, Don Towsley, and Yu Gu. Fluid Models and Solutions for Large-Scale IP Networks. In *Proceedings of ACM/SIGMETRIC 2003*, 2003.

[38] J. K. MacKie-Mason and H. R. Varian. Pricing the Internet. In B. Kahin and J. Keller, editor, *Public Access to the Internet*. Prentice Hall, 1994.

[39] J. K. MacKie-Mason and H. R. Varian. Pricing Congestible Network Resources. *IEEE J. Select. Areas Commun.*, 13(7):1141–1149, 1995.

[40] Andrew William Moore. Measurement-based management of network resources. Technical Report UCAM-CL-TR-528, University of Cambridge, England, April 2002.

[41] N. G. Duffield and O' Connell. Large Deviations and Overflow Probabilities for the General Single-Server queue, with Applications. In *Math, Proc. Cambridge Philos. Soc.*, pages 363–375, 1995.

[42] S. Nalatwad and M. Devetsikiotis. Self-Sizing Networks: Local vs. Global Control. In *Proceedings of IEEE ICC 2004*, volume 4, pages 2163– 2167, 2004.

[43] S. Nelakuditi, S. Varadarajan, and Z. Zhang. On Localized Control in QoS Routing. *IEEE Transactions on Automatic Control*, 47(6):1026–1032, June 2002.

[44] I. Norros. On the Use of Fractional Brownian Motion in the Theory of Connectionless Networks. *IEEE Journal on Selected Areas in Communications*, 13:953–962, 1994.

[45] A. Parekh and R. Gallager. A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single Node Case. *IEEE/ACM Trans. Networking*, pages 344–357, June 1993.

[46] Vern Paxson and Sally Floyd. Wide-Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Trans. Networking*, 3(3):226–244, June 1995.

[47] J. Qiu and E. Knightly. Measurement-based Admission Control with Aggregate Traffic Envelopes.

[48] Tamas Rapcsak. *Smooth Nonlinear Optimization in* R^n. Kluwer Academic Publishers, 1997.

[49] Rudolph H. Riedi, M.S. Crouse, V.Ribeiro, and R.G.Baraniuk. A Multifractal Wavelet Model with Application to Network Traffic. *IEEE Trans. Info. Theory*, 45(3):992–1018, April 1999.

[50] S. Jamin and S. J. Shenker, and P. B. Danzig. Comparison of Measurement-based Admission Control Algorithms for Controlled-Load Service. In *Proc. of IEEE INFOCOM*, 1997.

[51] S. Shenker. Fundamental Design Issues for the Future Internet. *IEEE Journal on Selected Areas in Communication*, 13(7):1176–1188, 1995.

[52] M. Shreedhar and G. Varghese. Efficient Fair Queueing using Deficit Round Robin. In *ACM SIGCOMM*, pages 231–242, 1995.

[53] V. Sivaraman and F. M. Chiussi. Providing End-to-End Statistical Delay Guarantees with Earliest Deadline First Scheduling and Per-Hop Traffic Shaping. In *Proc. of IEEE INFOCOM 2000*, 2000.

[54] Donald M. Topkis. *Supermodularity and Complementarity*. Princenton University Press, 1998.

[55] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the Self-Similar Nature of Ethernet Traffic. In *Proc. ACM Sigcomm'93*, pages 183–193, San Francisco, CA, 1993.

[56] Jean Walrand. *An Introduction to Queuing Networks*. Prentice Hall, 1988.

[57] Walter Willinger and Murad S. Taqqu and Ashok Erramilli. A Bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks. In F. P. Kelly, S. Zachary, and I. Ziedins, editors, *Stochastic Networks: Theory and Applications*, pages 339–366, Oxford, 1996. Clarendon Press.

[58] H. Wang, C. Shen, and K. G. Shin. Adaptive-Weighted Packet Scheduling for Premium Service. In *Proceedings of the IEEE ICC*, 2001.

[59] Wei Wei. Qos and video delivery. QoS_Video.ppt.

[60] Walter Willinger, Vern Paxson, and Murad S. Taqqu. *Self-Similarity and Heavy Tails: Structural Modeling of Network Traffic*. Birkhauser, Boston, 1998.

[61] X. Xiao and L. Ni. Internet QoS: The Big Picture. *IEEE Network Magazine*, 13:8–18, Mar 1999.

[62] Xipeng Xiao. *Providing Quality of Service In the Internet*. PhD thesis, Michigan State University, Michigan, 2000.

[63] Peng Xu. Measurement-based Quality of Service Scheduling. Technical report, ECE Department, North Carolina State University, NC, 2003.

[64] Peng Xu, Michael Devetsikiotis, and George Michailidis. Adaptive Scheduling using Online Measurements for Efficient Delivery of Quality of Service. Technical Report TR2004-12, SAMSI, RTP, NC, March 2004.

[65] Peng Xu, Michael Devetsikiotis, and George Michailidis. Online Scheduling for Optimal Resource Allocation in Flexible Heterogeneous Networks. *39th Conference on Information Sciences and Systems*, March 2005.

[66] Peng Xu, Michael Devetsikiotis, and George Michailidis. Profit-Oriented Resource Allocation Using Online Scheduling in Flexible Heterogeneous Networks. *accepted by Telecommunication Systems*, 2005.

[67] Peng Xu, George Michailidis, and Michael Devetsikiotis. Online Scheduling for Resource Allocation of Differentiated Services: Optimal Settings and Sensitivity Analysis. Technical Report TR2004-13, SAMSI, RTP, NC, March 2004.

[68] Dong Xuan. Part i: Introduction. NA6.ppt.

[69] J. Yan. Adaptive Configuration of Elastic High Speed Multiclass Networks. *IEEE Communication Magazine*, pages 116–120, May 1998.

[70] L. Zhang. VirtualClock: A New Traffic Control Algorithm for Packet Switching Networks. *ACM Transactions on Computer Systems*, pages 101–124, May 1991.

[71] Z. Zhang, C. Sanchez, B. Salkewicz, and E. Crawley. Quality of Service Extensions to OSPF. *Work in Progress*, Internet Draft, Sept. 1997.

[72] Z. L. Zhang and J. Kurose D. Towsley. Statistical Analysis of Generalized Processor Sharing Scheduling Discipline. *IEEE J. Select. Areas Commun.*, 13(6):1071–1080, 1995.

Appendices

Appendix A

Extended Study on Performance Evaluation of Chapter 4

In this appendix, we will continue our performance evaluation study of Chapter 4.

First, we examine the effect of the pre-specified threshold parameters (θ_1, θ_2) on the performance of the proposed scheme. It is easy to see that the higher the threshold θ_1 (for the delay-sensitive class), the better performance our proposed scheduling algorithms would achieve for that particular class. We thus briefly turn our attention to the loss-sensitive class, and explore the effect of θ_2 on the various classes. Two values are investigated, namely, $\theta_2 = 0.90$ and 0.95. Figures A.1 (a) and A.1 (b) show that the proposed scheduling algorithms achieve a better performance for the underlying loss-sensitive class, as expected, but more importantly the performance of the remaining classes is not significantly affected. A more extensive evaluation of the effect of the thresholds is currently under study.

We evaluate next how the interval over which traffic patterns change over time affects the performance of the proposed scheme in the under-provisioned case. We do this by defining an interval, called the pattern change interval (PCI), over which the

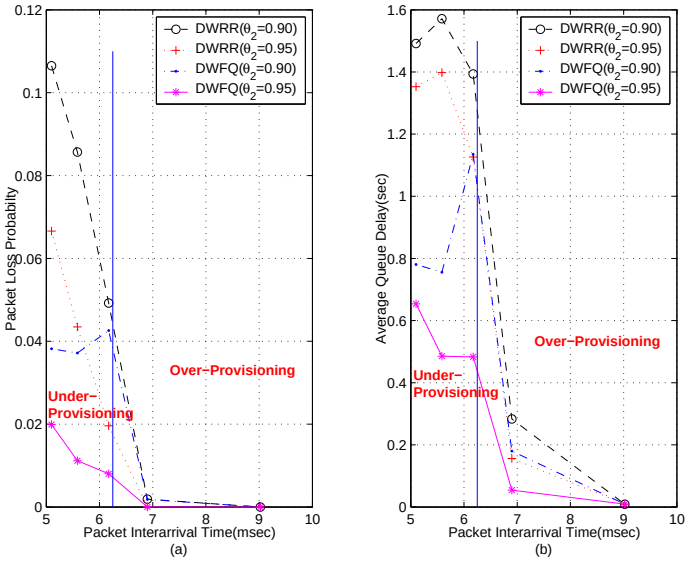


Figure A.1: Comparison of DWFQ and DWRR on (a) packet loss probability and (b) average queue delay with $\theta_2 = 0.90$ and $\theta_2 = 0.95$.

traffic load distribution for each class changes.

In this evaluation all the other tuning parameters remain fixed for the duration of the experiment. The PCI ranges for 10 to 90 sec, while the adaptive window size is 1 sec. Therefore, the dynamic policies have enough time to adapt to the changing traffic conditions and thus deliver good performance.

It can be seen from Figures A.2 (a) and A.2 (b), that the performance of DWFQ is not affected at all by changes in the PCI for the delay-sensitive class and similar conclusions can be reached to a large extent for the loss-sensitive one. On the other hand, the DWRR class exhibits an improving performance over larger PCIs for the delay-sensitive class and a degrading one over the loss sensitive (see Figure A.3). Finally, both policies exhibit some variability for the best effort class over the range of PCI (see Figure A.4). In terms of relative performance with respect to the static policies, the conclusion previously reached still apply. A topic that is currently under investigation is at what point the performance of the proposed dynamic allocation scheme degrades significantly as a function of PCI, or in other words for what values of the ratio of W/PCI the performance exhibited is deemed satisfactory.

Next, we briefly report on our investigations regarding the composition of the

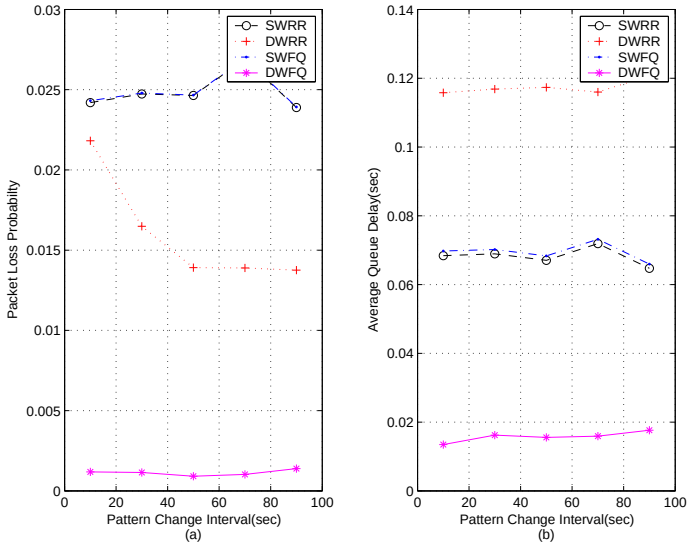


Figure A.2: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for delay-sensitive class over pattern change interval.

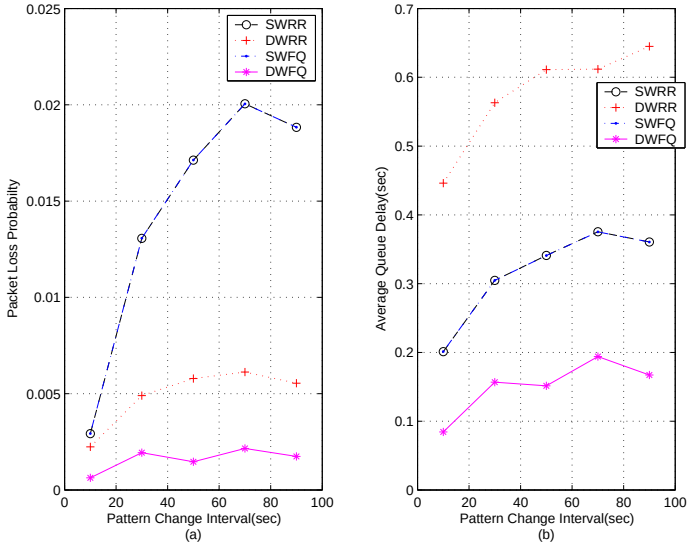


Figure A.3: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for loss-sensitive class over pattern change interval.

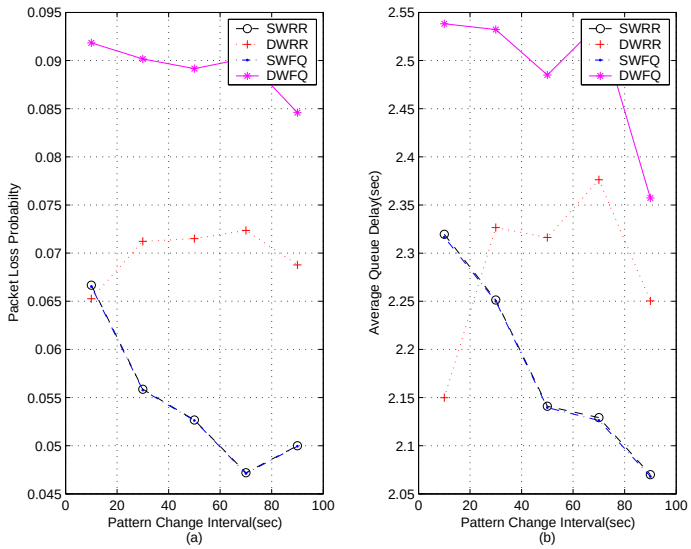


Figure A.4: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for best effort class over pattern change interval.

traffic and its effect on performance. The benchmark is a uniform distribution over the classes (i.e., equal proportions of 1/3, 1/3, and 1/3) and we also report the results for other unbalanced distribution such as those given in Table A.1, where over non-overlapping time intervals of length 100 sec, the composition of traffic follows those proportions; namely, at some point in time approximately 17% of the total traffic would come from the delay-sensitive class, an equal proportion from the loss-sensitive class and the remaining from the best effort class, while at another point in time, approximately 67% of the traffic would belong to the delay-sensitive class, and 17% to the other two classes. The various traffic load distributions examined are given in Table A.1, where σ =(max proportion/min proportion). A plot of the distribution of traffic as it is varying over time is given in Figure A.5, that provides some insight into the changing nature of traffic.

Figure A.7 (a), shows that the performance of the dynamic algorithms is superior to that of the static ones for unbalanced traffic distributions for the delay-sensitive class, regarding losses. It is also clear that the DWFQ enjoys a large margin over its competitors with respect to both performance metrics. On the other hand, the performance of DWRR with respect to delays is rather problematic, due to the nor-

Table A.1: Initial Proportions of Traffic distribution and corresponding σ values.

	Delay-Sensitive Class	Loss-Sensitive Class	Best Effort Class
$\sigma = 1$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$
$\sigma = 2$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$
$\sigma = 3$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{3}{5}$
$\sigma = 4$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{2}{3}$

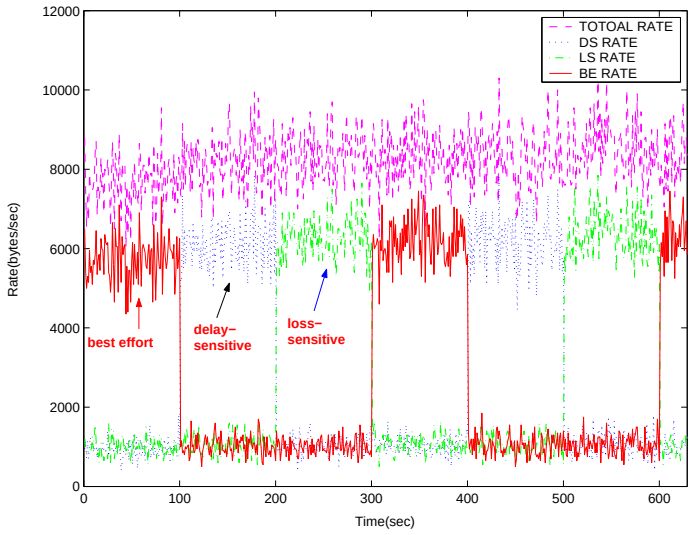


Figure A.5: Total traffic rate and traffic rates for different classes when $\sigma = 4$.

malization step used. Analogous conclusions can be reached for the loss-sensitive case for unbalanced distributions as Figure A.8 shows. For the best effort class, due to its low priority given by the dynamic scheme, the static algorithms outperform their dynamic counterparts, although the differences become smaller for more unbalanced load distributions.

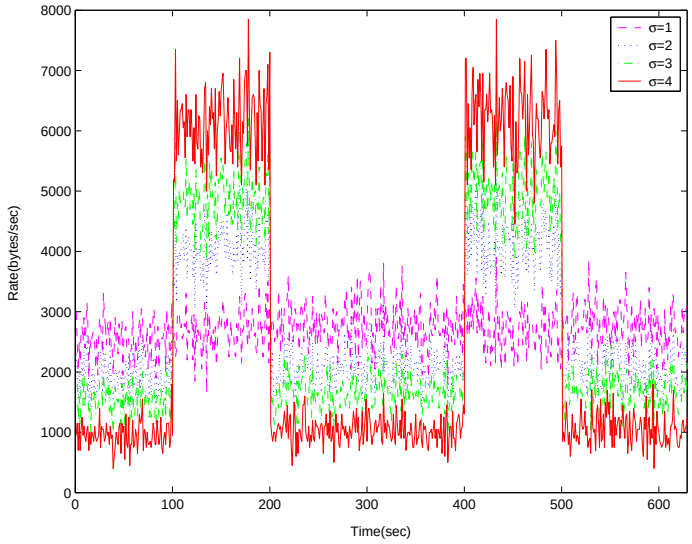


Figure A.6: Traffic rates for delay-sensitive class with different σ.

In Figures A.9 (a) and (b) the performance of the DWRR and the DWFQ policies is inferior to their static counterparts, as expected, due to the allocation of a large percentage of the available bandwidth to higher priority classes.

Overall, for very unbalanced composition of the traffic, our proposed scheme significantly outperforms the static algorithms, due to its flexibility and adaptiveness over time.

Finally, we briefly report on preliminary results obtained about the effect on performance of the adaptive window W. In [47], the adaptive window was chosen as a multiple of the measurement slot τ. Thus, the choice of τ is a key factor in determining the accuracy of the obtained measurements, which in turn affect the performance of the proposed scheme. If the value of τ is too small, it can capture the burstiness of the coming traffic and result in over-allocation of resources. The QoS requirements

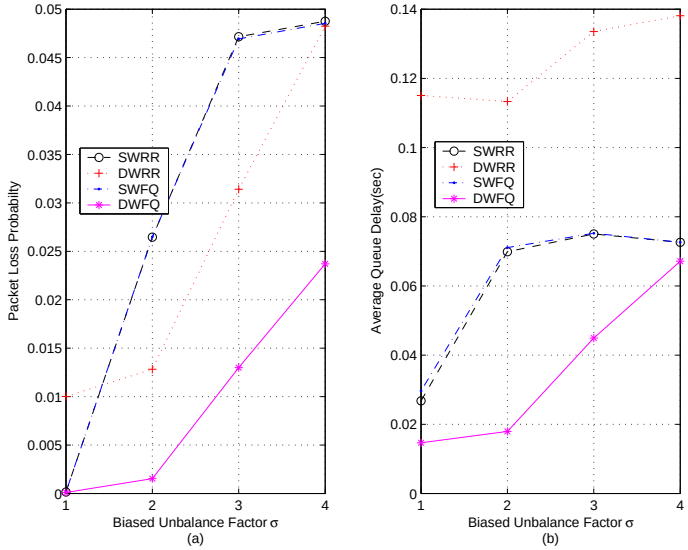


Figure A.7: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for delay-sensitive class as σ changes.

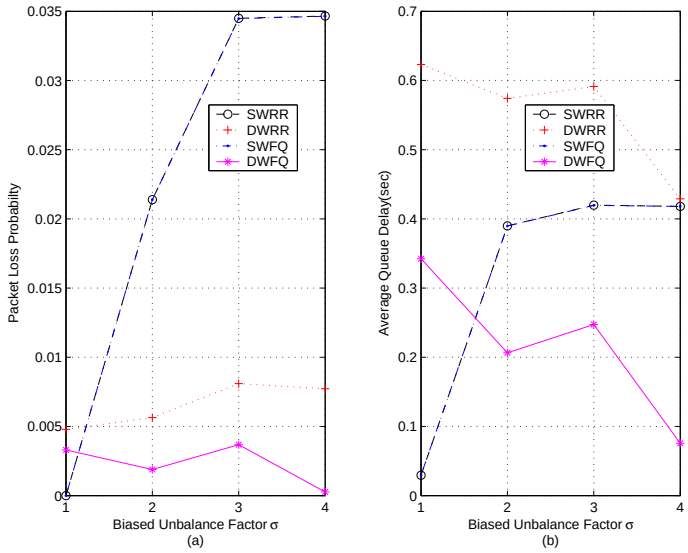


Figure A.8: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for loss-sensitive class as σ changes.

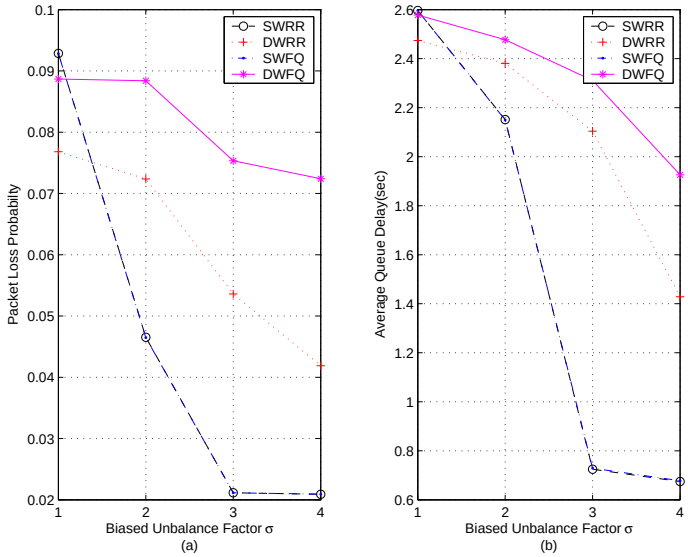


Figure A.9: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for best effort class as σ changes.

would be satisfied, but the computational requirements of our scheme become quite high. On the other hand, if the value of τ is too large, the measurements cannot capture well the fluctuations in the traffic processes, thus compromising performance.

It should be noted that in a rapidly changing traffic environment a near-optimal selection of τ becomes crucial; however, this represents a topic of current research. In this dissertation, the objective is to demonstrate the effect of τ on performance.

In Figures A.10 (a) and A.10 (b), the loss probability and average queue delay increase as τ increases, which is consistent with our intuition and our analysis. Moreover, the confidence intervals for the performance of the proposed scheme, and in particular for the DWFQ policy, are significantly narrower than those for the static algorithms. This reflects the more robust nature of our scheme.

Figures A.11 (a) and A.11 (b) show analogous results for the loss-sensitive class. However, for windows larger than 1.5 sec the delay performance of the DWFQ policy becomes inferior to that of the static algorithms. This indicates that the choice of W (and its relationship with the time scale over which traffic patterns change) is important for delivering the required QoS.

Finally, in Figures A.12 (a) and A.12 (b) it can be seen that for small values of

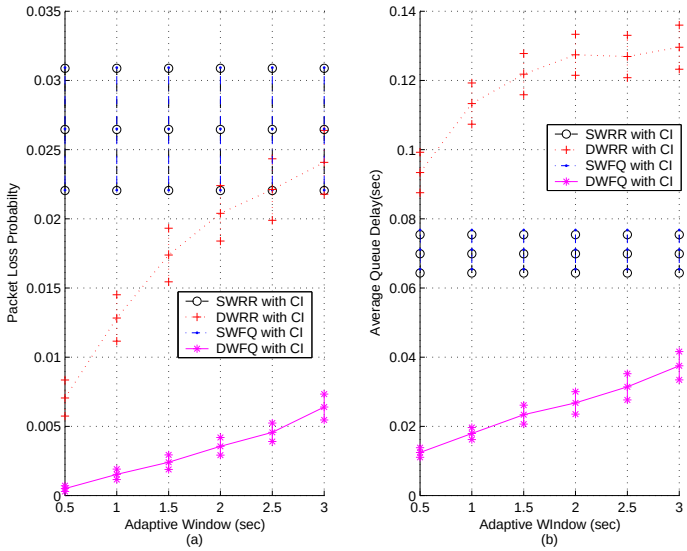


Figure A.10: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for delay-sensitive class as W changes.

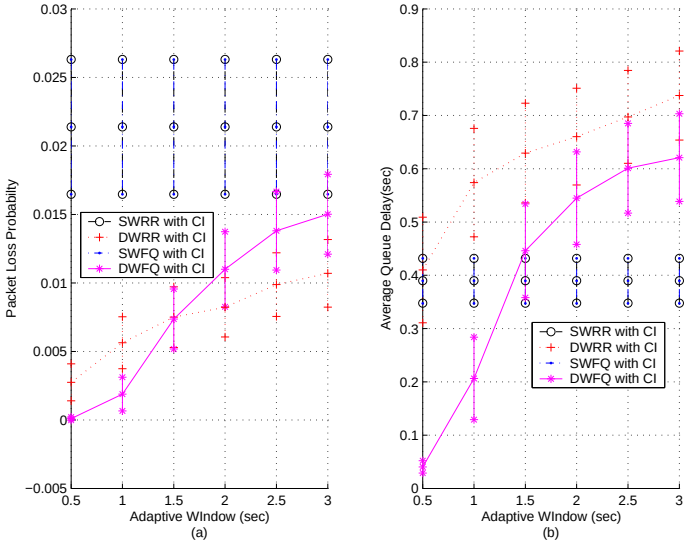


Figure A.11: Comparison of scheduling algorithms on (a) packet loss probability and (b) average queue delay for loss-sensitive class as W changes.

τ, the over-allocation for the higher priority classes results in an under-allocation for the best effort class. As τ increases, the proportion of bandwidth reserved for the higher priority classes gradually shifts to the best effort class, which yields a better QoS performance.

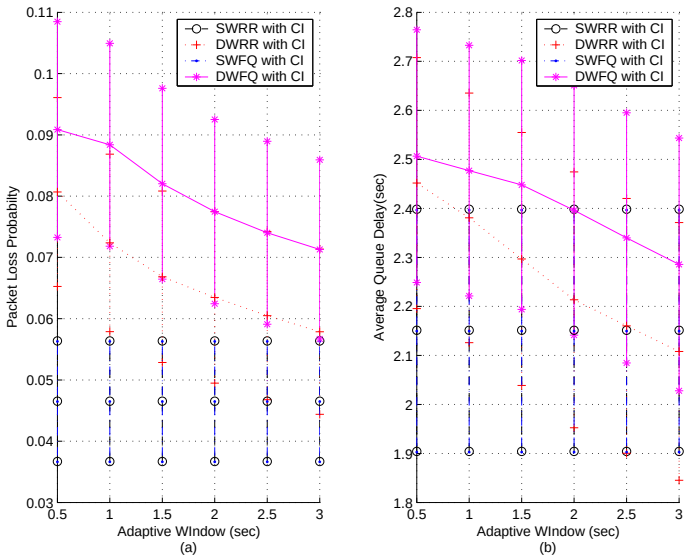


Figure A.12: Comparison of scheduling algorithms on (a) packet loss probability and b) average queue delay for best effort class as W changes.

Appendix B

Supermodular Definition and Related Theorems for Chapter 6

Definition 2. *A function $f : S \to \Re$ is supermodular if for all $x, y \in S$, $f(x)+f(y) \leq f(x \wedge y) + f(x \vee y)$.*

Theorem 4. *Let $S = [\underline{x}, \overline{x}]$ be an interval in $\Re^n$. Suppose that $f : \Re^n \to \Re$ is twice differentiable on some open set containing S. Then f is supermodular on S if and only if for all $x \in S$ and all $i \neq j$, $\frac{\partial^2 f}{\partial x_i \partial x_j} \geq 0$.*

Proof. The function, $\tilde{g}(\phi_1, p_1)$, is obviously twice differentiable. Since $\frac{\partial^2 \tilde{g}(\phi_1, p_1)}{\partial p_1 \partial \phi_1} = C' > 0$, it is supermodular based on Theorem 4. □

Lemma 1. *Let $f : X \times \Re \to \Re$ be supermodular function and define $x^* \equiv \arg\max_{x \in S(t)} f(x, t)$. If $t \geq t'$ and $S(t) \geq S(t')$, then $x^*(t) \geq x^*(t')$.*

Proof. The function, $\tilde{g}(\phi_1, p_1)$, was shown above to be supermodular. For the optimal value ϕ_1^*, it can be written as $\phi_1^*(p_1)$, a function of p_1. Suppose there is $p_1' > p_1$; then from the discussion about the constraints given in Section 3, it is easy to conclude that the feasible range of ϕ_1 is independent of p_1, thus $S(p_1) = S(p_1')$. From Lemma 1, $\phi_1^*(p_1) \geq \phi_1^*(p_1')$. □

Appendix C

Related Definitions and Proofs for Chapter 7

Definition 3. *Suppose $f \in \mathbb{R}$ and all x_0 in the interior of its domain, the left-derivative and right-derivative at x_0 can be defined as:*

$$\begin{aligned} D_- f(x_0) &:= \lim_{x \uparrow x_0} \frac{f(x) - f(x_0)}{x - x_0} \\ D_+ f(x_0) &:= \lim_{x \downarrow x_0} \frac{f(x) - f(x_0)}{x - x_0} \end{aligned}$$

Proposition 8. *Suppose $\phi_i \in (0, 1)$, the function of $\bar{q}_i(\phi_i)$ is differentiable everywhere in domain, and it is twice-differentiable except the point $\phi_i = \psi_i^c$.*

Proof. For both sides of the piecewise function except the point, $\phi_i = \psi_i^c$, it is obvious differentiable, thus we can obtain the first derivative of $\bar{q}_i(\phi_i)$ for both sides as the

following:

$$\frac{\partial \bar{q}_i(\phi_i)}{\partial \phi_i} = \begin{cases} -\frac{(q_i^0)^2 C'}{2W} \times \frac{1}{(\phi_i C' - r_i)^2} & \text{if } \phi_i > \psi_i^c \\ -\frac{WC'}{2} & \text{if } \phi_i \leq \psi_i^c \end{cases}$$

Meanwhile, at the point of $\phi_i = \psi_i^c$, we can derive the left-derivative is equivalent to the right-derivative as the following equation:

$$D_- \bar{q}_i = D_+ \bar{q}_i = -\frac{WC'}{2} \tag{C.1}$$

Thus, $\bar{q}_i$ is differentiable at the point, $\phi_i = \psi_i^c$ and it is differentiable everywhere on $\phi_i \in (0, 1)$.

Similarly, we can get the second derivative of $\bar{q}_i$ for both sides as the following:

$$\frac{\partial^2 \bar{q}_i(\phi_i)}{\partial \phi_i^2} = \begin{cases} \frac{(q_i^0 C')^2}{W} \times \frac{1}{(\phi_i C' - r_i)^3} & \text{if } \phi_i > \psi_i^c \\ 0 & \text{if } \phi_i \leq \psi_i^c \end{cases} \tag{C.2}$$

However, at the point of $\phi_i = \psi_i^c$, it is easy to prove the left second derivative is not equal to the right second derivative, thus $\bar{q}_i$ is not twice differentiable at this point but twice differentiable for all others points in the domain $(0, 1)$. □

Proposition 9. *The function of $\bar{q}_i(\phi_i)$ is convex for $\phi_i \in (0, 1)$.*

Proof. When $\phi_i \in (0, \psi_i^c)$, $\bar{q}_i$ is a linear function, therefore its first derivative is constant and we can get the following relationship:

$$\frac{\bar{q}_i(\psi_i^c) - \bar{q}_i(0)}{\psi_i^c - 0} = \sup_{\phi_i \in (0, \psi_i^c)} D_+ \bar{q}_i(\phi_i) = D_+ \bar{q}_i(\psi_i^c) = -\frac{WC'}{2}$$

While $\phi_i \in (\psi_i^c, 1)$, the right derivative in this range has the increasing property:

$$D_+\bar{q}_i(\psi_i^c) \leq \inf_{\phi_i \in (\psi_i^c, 1)} D_+\bar{q}_i(\phi_i) \leq \frac{\bar{q}_i(1) - \bar{q}_i(\psi_i^c)}{1 - \psi_i^c}$$

Therefore, the right derivative of $\bar{q}_i$ on $(0, 1)$ is increasing. Then $\bar{q}_i$ is convex on $(0, 1)$. □