Chapter 1

INTRODUCTION

1.1 Significance of the Research

The Internet has historically offered a single level of service ("best effort," without any guarantee of quality) via the Internet Protocol (IP). Its best-effort service means service in which all data packets are treated equally. The quality of service (QoS) on IP (hereinafter referred to as IP QoS) [1], [2] has not been a significant issue in the Internet community until recently. The IP datagram (packet) header includes one type-of-service (ToS) byte. ToS values can be used to indicate the different QoS needs of the datagram, and they can be taken into account by routers for choosing among the different types of datagram transport. However, the ToS byte has essentially been ignored by both end-systems (i.e., applications generating IP packets) and routers. Unlike traditional network applications such as telnet and file transfer, which employ the Transmission Control Protocol (TCP) as a transport layer, continuous media (CM) applications[1] such as video-on-demand, video conferencing, and Internet telephony have emerged and demanded more strict service requirements, including explicit delay bounds and limits on packet loss rates.

One approach to emerging CM applications is to provide abundant network bandwidth (e.g., by using high-speed transmission technology and/or optical fibers) while maintaining best-effort protocols. However, despite the astounding rate at which processing speed and link capacity are increasing, we see congestion in many places in networks today and expect to see similar situations in the foreseeable future. There will be more and more new bandwidth-demanding applications as connectivities and services of broadband networks expand and diversify. In addition, there is no guarantee that the Internet topology will be free of bottleneck links even if the transmis-

[1] A "continuous media" application is an application whose content must be displayed at the receiver as a continuous flow of data with proper timing.

sion speeds of physical networks keep increasing. (This point is particularly noteworthy because we are seeing more and more devices connected through wireless links.) TCP congestion control and the best-effort IP by themselves seem inadequate to satisfy the diverse network applications of the future. Also, from the standpoints of network pricing and the network service providers' economics, the *"same service for all"* paradigm seems inadequate for the expected future of network evolution.

Another approach is: (1) allocate network resources to different types of network traffic on the basis of their performance requirements by using more effective network protocols; and (2) perform careful management of network resources. It is deemed important today that a network should provide, to some extent, different qualities of service to different applications in accordance with their performance needs. We believe that QoS provision and effective resource management will continue to be important even in the broadband era because a greater variety of applications demanding widely different levels of network performance will be created as network speeds increase.

Simply speaking, QoS provision can be viewed as the ability of network service providers (or operators) to handle the performance needs of different types of application traffic by allocating network resources appropriately. This book explores the network provider's QoS provision mechanism, the end-system application's mechanism for adapting to the temporal variation of the network service quality, and the interaction and cooperation of the two mechanisms (the network's and the application's).

Different kinds of traffic streams are aggregated at the gateways to the service providers' networks (these are also called edge routers or boundary nodes), and intelligent packet management schemes can be used to provide QoS. QoS provisioning [3] is one of the critical issues in networked multimedia applications. As the Internet evolves, the number of diverse applications being deployed will increase significantly, and many of these applications will require more stringent performance guarantees in terms of bandwidth and end-to-end delay than the current Internet and its best-effort service can provide. Since the best-effort service in place today cannot support these expected application requirements, a great deal of effort must be expended to construct additional services to meet the demand of emerging applications.

For most multimedia applications, the QoS performance measure in the application layer is actually a subjective one based on human perception. It is often assumed that a subjective QoS measure can be translated into, or reflected by, some objective measures such as average delay, delay jitter, loss rate, etc. However, multimedia applications can have very diverse QoS

requirements. For example, applications such as medical images for remote diagnosis demand extremely reliable information delivery. Additionally, remote real-time control messages for some applications demand reliable and timely information delivery. Thus, it is critical to guarantee that no packet is lost or delayed in the network for such applications. On the other hand, other multimedia applications such as entertainment audio and video can tolerate some fraction of lost or delayed packets. Thus, it is important for a network service provider to meet the diverse QoS requirements presented by different applications.

QoS requirements can be either hard (i.e., deterministic) or soft (i.e., statistical). In the hard QoS case, guarantees are provided and strictly enforced based on a contract between the users and the service network. In the soft QoS case, guarantees are promised in a statistical sense, but may not be strictly enforced for a single instance. It must be added that even packets of the same media application may have different QoS requirements in terms of delay and packet loss preference, which leads to a soft QoS rating for the application. Soft QoS services can be divided into classes characterized by different QoS assurance levels. In the current best-effort service environment, no QoS guarantees are supported.

To provide QoS for media delivery, it is important to consider the interaction between the application (i.e., properties of the delivered media) and the network (i.e., network resource management). Designing and analyzing such interactions are the central themes of this book.

1.2 Scope of the Research

1.2.1 End-System's QoS Support

In the current best-effort Internet environment, there is a large amount of work placed on the end-system's (i.e., host's or application's) QoS-supporting mechanisms, including source rate control/adaptation, packet loss/error control based on forward error correction (FEC), and retransmission. Some of them assume best-effort Internet service, and some assume a moderate QoS support capability on the part of the network. For CM applications, it is possible to implement additional packet loss control techniques such as error-resilient coding and error concealment at the encoder and decoder. Many of these applications' mechanisms can be categorized as "adaptive applications" [4] from a network engineer's point of view.

For source rate control, an end-system estimates the network's status and adapts to its congestion level by changing transmission rates (e.g., changing

the source coding) or relinquishing the transmission of some packets. For example, in response to the packet loss and delay feedback obtained through TCP (or TCP-like) congestion control schemes, a video application may adjust its rate through spatial or temporal quality adjustment. This approach has been taken by researchers under the best-effort network [5], [6], [7], [8], [9], [10]. For instance, a specified frame drop order was considered in [11] based on the degree of network congestion, an example of which is the order of the B-, P-, and I-coded frames[2] of the coded bit stream. However, special attention must be paid to the dropping of I-frames, because I-frames are usually too large for dropping and because the dropping of an I-frame can cause severe video quality degradation.

Research on loss/error control can be divided into three areas:

- Feedback control with a conventional retransmission-based scheme [12] such as automatic repeat request (ARQ)
- Feedforward control using channel coding such as FEC [13] and unequal error/loss protection in a joint source/channel coding scheme [14]
- Error resilience/concealment techniques that limit the damage of packet loss by using post-processing at the decoder

This is a very broad and active research field. For readers interested in the end-system's adaptation schemes, refer to the recently published tutorial paper [15] and its references.

1.2.2 QoS Provision via Network Support

Another approach to end-to-end QoS is to have the network provide more assurance than the best-effort model in terms of the network's QoS parameters within its resource capacity. In this approach, the network node plays an active role in controlling end-to-end service quality. Various ideas related to resource reservation and prioritization have been under intensive study.

There are two well-known types of architecture for IP QoS: integrated services (IntServ) [16] and differentiated services (DiffServ, also referred to as DS) [17]. In addition, multiprotocol label switching (MPLS) [18] can be used to provide QoS guarantees for IP datagrams, although the original motivation for MPLS architecture was not primarily for IP QoS.

In the IntServ architecture, the QoS of each flow can be guaranteed by reserving a certain amount of network resources (link bandwidth, buffer

[2] In motion picture coding group (MPEG), there are three types of coding a picture or frame such as bidirectional predictive (B), predictive (P), and intra (I) coded modes.

space, etc.) on the basis of the flow's traffic characteristics at the source and the destination's choice of QoS level (or the destination's choices of QoS levels in the case of multicasting). IntServ uses the resource reservation protocol (RSVP) [19], [20], [21] to reserve network resources for a given flow. In fact, RSVP, which is basically a network signaling protocol, is a very important component of the IntServ architecture. The basic idea is to reserve the needed bandwidth and buffer space prior to the actual movement of data traffic along the path that the traffic will take.

An alternative, and simpler, solution—which can also be complementary to IntServ—has been examined by the Internet Engineering Task Force (IETF) DiffServ working group [17]. In order to avoid per-flow state maintenance, routers in the DiffServ architecture only distinguish traffic classes for which resources are allocated, as opposed to distinguishing individual flows for which resources are reserved. The ToS field of the IP header can be used to indicate the class to which a datagram belongs. (In the DiffServ architecture, the field that differentiates classes is referred to as the DS field.) Thus, the DiffServ architecture can be viewed as a stateless approach, whereas the IntServ architecture can be viewed as a stateful approach. The major goal of these architecture design efforts was to define configurable packet-forwarding mechanisms (called per-hop behaviors, or PHBs) that could locally (at each link) differentiate classes of flows, instead of differentiating individual flows. It should be noted that multiple individual flows typically belong to a single class. In the DiffServ architecture, the network service provider can define traffic classes and allocate network resources (e.g., bandwidths and buffer spaces at different links) by configuring the routers.

Under a given resource allocation, the QoS of each class can be regulated by regulating the traffic of that class at the ingress of the network. Therefore, the DiffServ architecture recommends establishing service level agreements (SLAs) between the network service provider and its subscribers. An SLA contains the contract between the service provider and the subscriber regarding the characteristics of traffic allowable at the ingress. Thus, in DiffServ architecture, a user tunes an application to the network services provided, while in the IntServ architecture, the network tunes its resources to an application. In terms of the ability to guarantee end-to-end QoS, DiffServ is between the IntServ architecture and the best-effort-only (same service for all) architectures.

A DiffServ network replaces the best-effort IP packet-forwarding mechanism with a more QoS-aware design. Using packet-forwarding mechanisms (PHBs) to differentiate classes can be done in a number of ways. Obviously, a particular packet-scheduling scheme results in a specified sharing of

link bandwidth among different classes. For example, different classes may have their bandwidth protected through weighted fair-queueing [22] packet scheduling. Or, as another example, packets from different classes may be served at a link in accordance with a strict priority.

Beyond packet scheduling, queue management for different classes results in a different set of service qualities for the classes. For example, packets of Class A may be dropped after packets of Class B when the buffer is full [23], [24]. Service providers can use these tools to create their own service classes.

The stateless approach of the DiffServ architecture pushes most of the complexity to the network's edges. Traffic flows are aggregated into a few classes handled by routers in accordance with a set of PHBs. This class-based service is more scalable than the per-flow approach of the IntServ architecture.

Most of the research presented in this book focuses on networks that support QoS through the DiffServ framework. From the Internet QoS point of view, MPLS [18] can be conceptually viewed as a method of establishing a virtual circuit for IP datagrams. We note that the IP layer provides a connectionless (datagram) service and that IP by itself does not establish virtual circuits. MPLS adds a small header (the MPLS shim) to IP datagrams with a "label" field, which is used as the virtual channel number [25]. By allocating certain network resources (bandwidth, buffer space, etc.) along a virtual circuit, the QoS of packets using the virtual circuit can be protected. Thus, the concept of utilizing MPLS for QoS is the same as that of Asynchronous Transfer Mode (ATM) [26] networks.

1.2.3 QoS Interaction Between End-Systems and QoS-Enabled Networks

In our work, the major role of an application entity is limited to assigning important indices to source packets, because application layer entities know only the priority order of their own CM streams (e.g, the measure of importance for each packet to experience a low delay or low probability of loss from the point of view of an application layer QoS performance parameter). QoS-enabled network has a mechanism to provide various service types with different unit costs. At this point, new issues arise in the QoS interaction between CM applications and a DiffServ network's service classes. We must address the following problems:

- How to categorize CM source segments according to their importance in order to have low loss rate and low delay.
- How to provide optimal or cost-effective QoS mapping from CM source categories to service classes provided by a DiffServ network.
- How a DiffServ network will maintain stable and persistent differentiated network QoS under a time-varying network load condition.
- What can be done at the boundary between the DiffServ domain and the subscriber domain to control the QoS of streams merging at the boundary.
- How to incorporate rate adaptation performed by the end-system and DiffServ provided by the network.
- What is the most effective way to perform multicasting over DiffServ?
- How to evaluate the performance of combining application-level and network-level unequal error protection (UEP) in RSVP/IntServ or DiffServ.

These problems are clearly described and addressed in this book.

1.3 Contribution of the Research

One main contribution of this research is to set up a framework within which end-systems and a DiffServ network can cooperate for better end-to-end QoS provision. This framework includes the following key components and addresses the following requirements:

- An application's data segments of single or multiple video streams (or other multimedia streams) are packetized and then categorized according to the application-level QoS's sensitivity to packet loss and delay. A quantitative index is given to each packet to reflect its importance relative to receiving good QoS from the network. This mechanism guides an end-system to intelligently differentiate, on the basis of data content, the QoS levels of the network service to be requested.
- The mapping from the application data's QoS categories to the network service classes, which will often be called "QoS mapping" in this book, must be cost-effective. The QoS mapping should be designed with an awareness of both the meaning of the application's QoS categorization

and the QoS provided by the network side (e.g., the number of DiffServ classes and QoS parameters and their meaning, etc.)

- For a service contract to be constructed, optimal or effective QoS mapping per flow or per aggregated class requires a balance between the QoS requests assigned by a user and the limited number of QoS levels of a DiffServ network.
- Our framework includes proper resource management schemes (i.e., queue management, packet scheduling, and traffic conditioning), which are to be employed by the network to realize stable and consistent differentiation of QoS levels among different classes under time-varying network load conditions.
- Our framework includes an intelligent traffic-conditioning mechanism at boundary nodes (or gateways), which is necessary to optimize performance while meeting the SLA between the access or customer network and the network service provided.
- Our framework includes a rate adaptation module at the end-system, because a layered or scalable source-encoded stream is required.
- We address IP multicasting over a DiffServ network as a method for eliminating unnecessary transmission of duplicate packets across the network backbone.
- The effective combination of application-level and network-level efforts needs to be considered for QoS support.

The contributions of this research include the following:

- We present a QoS mapping framework between the prioritized or categorized CM stream segments and the service levels of the QoS-enabled network in terms of packet loss and delay performance.
- We propose a normalized and unified indexing scheme for the QoS request of an application, which we call the relative priority index (RPI). An RPI is obtained by combining different video factors in a video stream and categorizing video data segments according to their importance with respect to receiving good QoS in delivery.
- We investigate optimal or effective QoS mapping between a video stream and a QoS-enabled network. Under a given total pricing budget, several packets from a video stream, categorized on the basis of

the RPI, can be forwarded to the QoS mapping mechanism to achieve improved end-to-end video quality.

- We propose an adaptive packet-forwarding algorithm to provide relative service differentiation in terms of packet loss and delay. This algorithm enables the measured network DS level to stay within a stable range and not fluctuate too much under variable network load conditions.

- Using feedforward and feedback QoS control, we developed dynamic QoS mapping control at a special-purpose device called the video gateway to enhance stable and persistent differentiated services for incoming pre-classified packets.

- We propose a seamless integration of rate adaptation, prioritized packetization, and simplified differentiation for MPEG-4 fine granular scalability (FGS) video streaming.

- We propose a joint source network (JSN) adaptation to provide efficient and cost-effective end-to-end quality in layered streaming video through the combination of error-resilient packetization, selectively used FEC, and QoS-controlled networks such as IntServ and DiffServ networks.

- We present a priority-marking model of receiver-driven multicast video layers that conforms to the priority-marking architecture of DiffServ networks.

1.4 Outline of this Book

The main objective of this research was to construct a system in which multimedia applications and the network service cooperated positively to realize efficient end-to-end QoS provision. With this goal in mind, the research content can be conceptually delineated as: (1) the efforts to be made by the application side (i.e., the application's functions); (2) the efforts to be made by the network side to facilitate the cooperation; and (3) QoS mapping from the application's content classes to the network's service classes.

On the application side, we investigated simple categorization of the contents of coded video. In video coding, each frame tends to have different QoS requirements in terms of packet loss and delay tolerance due to the interframe dependency relationship. We discuss how to categorize (prioritize) coded video content on the basis of the impacts that delay or loss have on

video quality. On the network side, we explore how some details of network service provision affect the "cooperative efforts" between the application and network service and thus affect the end-to-end CM application's QoS. In the context of facilitating this "cooperation," we discuss the adaptive packet-forwarding algorithm to be used by the network to achieve service differentiation. One of the key concepts introduced in this book is QoS mapping. We present a conceptual basis for and a formulation of QoS mapping. The remainder of this book is organized as follows.

In Chapter 2, we present an overview of issues and challenges related to network support for QoS provision from the DiffServ perspective. Relative and absolute service differentiation approaches are examined. They provide different solutions while complementing each other. Also, related works on the interaction between video applications and network resource management are surveyed.

In Chapter 3, we develop an overall QoS mapping framework from encoded and prioritized video packets to service levels of DiffServ-enabled networks. First, we investigate the use of video factors to be used to differentiate, classify, and packetize a video stream into different priority classes in terms of packet loss and delay preference. As a result, we derive an optimal (or effective) per-flow QoS mapping guideline. In addition, pre-categorized packets in a flow are mapped different network QoS levels to get best end-to-end video quality under a given total cost constraint. We present a simple, online-computable RPI for categorizing video content, which we evaluated through simulation. Specifically, we simulated, with two-state Markov models, network conditions represented by the different packet loss rates of several network DiffServ levels.

In Chapter 4, we propose an adaptive packet-forwarding mechanism to maintain stable and robust relative service differentiation under time-varying network load conditions. The main network QoS tools include queue management and scheduling. The proposed packet-forwarding is based on the widely accepted multiple random early detection (RED)-based queue management and weighted fair queueing (WFQ)-based scheduling schemes. We evaluated the proposed mechanism through network simulator (NS) [27].

Chapter 5 considers the issue of dynamic and aggregated QoS mapping control at the edge of a DiffServ domain, which can be regarded as a gateway. We introduce a system (device) that we call a video gateway (VG), which is placed at the border between a DiffServ domain and a content provider's site for processing aggregated video streams for the site. The VG is responsible for coordinating the QoS mapping between video applications and the DiffServ-enabled network. To achieve reliable and consistent end-to-end

video streaming with awareness of relative service differentiation, dynamic QoS mapping is performed at the VG using dynamic feedforward/feedback QoS control. The mapping from the categorized index to the DS level can be dynamically adjusted in accordance with the load variation of the DiffServ networks in three granularity levels: packet-, session-, and class-based granularities. For class-based QoS control, an adaptive traffic-conditioning mechanism was analyzed and is proposed. In particular, we focus on issues resulting from variations in QoS demand of CM applications (e.g., varying priorities from aggregated/categorized packets) and variations in QoS supply of a DiffServ network (e.g., varying loss/delay due to fluctuating network loads). Enhanced quality provisioning for CM applications is demonstrated under a given pricing model with proposed QoS control schemes in both feedforward and feedback control fashion.

In Chapter 6, we focus on the seamless integration of rate adaptation, prioritized packetization, and simplified differentiation for MPEG-4 FGS video streaming. The proposed system consists of three key components: (1) rate adaptation with scalable source encoding; (2) content-aware prioritized packetization; and (3) loss-based differential forwarding. More specifically, constant-quality rate adaptation is first achieved by optimally truncating the over-coded FGS stream based on the embedding rate-distortion (R-D) information (obtained from a piecewise linear R-D model). The rate-controlled video stream is then packetized and prioritized based on the loss impact of each packet. Prioritized packets are transmitted over the underlying network, where packets are subject to differentiated dropping and forwarding. By focusing on the end-to-end quality, we establish effective working conditions for the proposed video streaming, and the superior performance is verified by simulated MPEG-4 FGS video streaming.

In Chapter 7, we propose and discuss a JSN adaptation to provide efficient and cost-effective end-to-end quality in layered streaming video through the combination of error-resilient packetization with selectively used FEC and QoS-controlled networks such as IntServ and DiffServ networks. Our JSN approach uses a simple network price mechanism to achieve the best end-to-end quality with guaranteed minimum service quality under given total cost constraints. We also propose performance metrics for video. In addition to objective metrics like peak signal-to-noise ratio (PSNR), we propose using the corrupted/frozen frame ratios (CFR, FFR) to monitor the perceptual quality the user feels. The experimental results in the proposed DiffServ case of our JSN approach clearly show better quality than the case in which network QoS is supported only as a cost-effective and practically local-optimal way.

In Chapter 8, we present an enhanced network service model for layered video multicast applications in DiffServ networks, which consists of enhanced active queue management (AQM) and hierarchical priority marking. Our experiments showed that the receiver-driven layered multicast(RLM) protocol applied directly in DiffServ networks performs even more poorly than the protocol applied in the best-effort Internet environment. The main reason for this is that the joint experiment of RLM-style applications behaves like an unresponsive flow and conflicts with the DiffServ network's queues. Thus, we present three sets of extended multiple RED parameters as the enhanced AQM model, which is designed to be generally applicable to service protection as well as to DiffServ network support for RLM traffic. Then, in order to take advantage of the priority service in a DiffServ network, we present a priority-marking model of RLM video layers that conforms to the priority-marking architecture of DiffServ networks. Through a number of simulation experiments, we show that our network service enhancement model improves joint experiment stability and reduces packet loss satisfactorily for supporting RLM traffic in DiffServ.

Concluding remarks and extensions of this research are given in Chapter 9.