Global QoS Management for Multicasting Applications

Thesis by

Yu Zhong

In Partial Fulfillment of the requirements
for the Degree of
Master in Applied Science
In Systems Science

Department of Systems Science
University of Ottawa
Ontario, Canada

December 2002

I hereby declare that I am the sole author of this thesis. I authorize the University of Ottawa to lend this paper to other institutions or individuals for the purpose of scholarly research.

Yu Zhong

I further autorize the University of Ottawa to reproduce this report by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research

Yu Zhong

Abstract

In a multicast network of a large number of receivers where transmission resources and end-systems are of high heterogeneity, the receivers require automatically selected heterogeneous QoS streams that meet their end-system requirements. In this thesis, we introduce a Receiver-Satisfaction Driven (RSD) method that takes into account user preferences stored in user profiles in order to automatically select the stream variants that maximize the overall user satisfaction. In the case of a very large number of receivers, the receivers may be partitioned into different user classes according to factors such as the bandwidth limit. Each class is allowed to send one or several representative group profiles, depending on the number of users in that group. The source processes the data from the group profiles and determines the optimal QoS parameters for a given number of stream variants. Our simulation results show that the streams variants generated based on representative group profiles give a reasonable satisfaction level for all receivers.

Acknowledgement

I would like to thank my supervisor, Dr. Gregor von Bochmann, whose comments and support have been a great help during preparation of this work. I had learned a lot from his expertise in this area. I would like to give thanks to my colleague Khalil El-Khatib in the Distributed Systems Research Group, for his kind cooperation and friendship. I would like to express my gratitude to friends who support me greatly in prayers. Without the strength from above, this work would have been impossible.

TABLE OF CONTENTS

CHAPTER I INTRODUCTION

Within recent years, the great improvement in network delivery infrastructures, digital technologies and end-system processing power have made it possible to use existing networks for multicast applications such as video conferencing, distant learning (teleteaching), remote presentation and media-on-demand. In a network of a large number of receivers where transmission resources and end-systems are of high heterogeneity, it does not make sense for users with different capacities to receive the same multimedia service for one multimedia stream content. Stream variants with different quality are required for different users. On the other hand, due to the server access link bandwidth limitation, it is not possible to allow each end-user to receive a separate multimedia stream. In the case of a very large number of users, multicasting can be used to send a single stream from the source to many receivers. It is therefore proposed to send several streams with different qualities, but same content, and each stream is multicast to those users so that they can receive the particular quality.

We can address this problem as *"bandwidth economy"* vs. *"granularity of control"*. A totally fair scheme with high granularity of control should enable receivers to receive an end-to-end application stream that meets their requirements. In that way, there would be hundreds and thousands of stream variants in the network from the same source to many destinations within the same application. This would greatly increase the network traffic and lead to congestion thus leading to high loss rate, and lower available quality of service. In contrast, using single multicast stream has very good bandwidth economy but very poor granularity of control, since the quality of the stream that is shared among all users within the same application may not be ideal (or even acceptable) for all receivers.

A lot of studies have been done that try to find out how to deliver several multicast stream variants from the source to target receivers by considering the variation of end-users' requirements as well as network bandwidth limitation. In order to adapt to its end-users' bandwidth capacity, a multicast stream may be sent in the format of a single stream, replicated streams, and a layered stream. In a single stream, a single encoded multicast stream is transmitted by the source; the source uses the feedback information from the receivers to adapt its data rate. With replicated streams the source sends several streams carrying the same information with different quality and bit rate (by encoding the streams with different compression parameters). Each stream is multicast to a different group of receivers categorized by their bandwidth capacity. Feedback from the receivers can be used to adjust the data rate of the stream they are receiving. With a layered stream, the output bit stream is divided into a base layer and one or more enhancement layers. The base layer is independently decoded and provides the basic level video quality. The enhancement layers can only be decoded together with the base layer to provide an improved quality. The receiver would then join at least the base layer group, and join as many enhancement layer groups as his/her receiving capabilities allow.

In reality, bandwidth capacity is not the only factor that determines which steam the user would like to receive. There are *personal preferences* for each individual such as the ***acceptable*** and ***ideal*** level of the quality for the user. Those *personal preference* parameters, together with the available bandwidth capacity, can be written into a *user profile* and used by a *user satisfaction function* to calculate how the user would be satisfied from a given stream. The set of stream variants that generate the best satisfaction for all users should be sent out.

In this thesis, we will introduce a Receiver-Satisfaction Driven (RSD) method, that make usage of *user profile* and the *satisfaction function*, to automatically select the stream variants that maximize the overall user satisfaction. The source collects data in the format of user population profiles from the receivers, run them through a certain optimization algorithm, and generates the set of stream variants that maximize the satisfaction of all users. The stream variants are for the same application content, but different quality parameters. In the simplest case, it would be a single channel; however, in most cases, several concurrent channels would be selected. In a context of a large number of receivers, the receivers can be partitioned into different user classes according to factors such as bandwidth limitation. We propose that each class of users can be characterized by what we call a representative group profile. The source processes the data from the representative group profiles, and uses this information to select the stream variants for transmission. Our simulation results show that the streams variants generated based on representative group profiles give reasonable satisfaction levels for all receivers of an application.

The main contributions of the thesis are:

(1) We have defined an approach for automatically selecting the stream variants for multimedia information multicast to maximize overall user satisfaction.

(2) We have defined a scheme to partition the receivers into different classes and characterize the receivers of a given class by representative group profile(s) that can be used for the automated selection of stream variants.

(3) We have compared three different user class characterization approaches. The simulation results show that the satisfaction generated from the boundary-value approach is the best among the three approaches.

The layout of this thesis is as follows: In Chapter 2 we introduce the fundamental concepts of multimedia applications and quality of service. Chapter 3 is the literature review on video multicast over the Internet. In Chapter 4 we propose the Receiver-Satisfaction Driven method for automatically selecting QoS parameters. Chapter 5 presents some simulation studies and explains the results. Chapter 9 concludes the thesis.

Chapter II FUNDAMENTALS OF QOS IN MULTIMEDIA SYSTEMS

2.1 Definition and Types of Multimedia Systems

A *Multimedia system*[1] is defined as being capable of simultaneously supporting the processing and communication of several media types with at least one *time-continuous* medium in *digital* form. Any types of multimedia information have the following in common. 1. They combine at least one continuous medium, 2. The continuous medium has to be in digital form.

There are two aspects of multimedia systems: communications and computing. Multimedia communications emphasizes multimedia information capturing, transmission and presentation, while multimedia computing emphasizes processing of multimedia information such as searching, retrieval, recognition, and enhancement. Distinction between those two is not clear-cut.

Examples of multimedia system applications are Video On Demand (VOD), Information On Demand (IOD), Teleteaching, Telemedicine, Videophone and Videoconference etc. Multimedia systems can be classified into standalone and distributed systems. Standalone systems use local dedicated system resources. On the other hand, distributed systems share both system resources and information resources and can support communication among users.

2.2 Characteristics of Multimedia Systems

Most multimedia system applications, represented by digital audio, video and image, are converted from their analog counterpart. Analog to Digital Conversion (ADC) includes converting continuous time into discrete values (also called *sampling*), converting continuous sample values into discrete values (also called *quantization*), and representing quantized values digitally (also called *coding*)[2].

Major requirements of a networked multimedia system include storage capacity and bandwidth, delay and jitter constraints, error rate, retrieval techniques and synchronization.

For example:

Storage requirement (in octets) without compression = H * V * P / 8

H -- Number of pixels on each line
V – The number of lines in the image
P – The pixel depth (bits per pixel)

Both audio and video are time continuous, we normally characterize them in bits/s or Mbits/s. The bandwidth requirement of different applications varies a lot. For example, digital telephone has a data rate requirement of 64 kbit/s, but CD-Audio has data rate requirement of 1,411.2 kbit/s, and HDTV (High-Definition Television systems) asks for 864,000 kbit/s[3]. Since digital audio, image and video require huge amount of data for representation and very-high-network bandwidth for transmission, data compression is necessary for multimedia applications.

Digital audio and video are time-dependent continuous media, thus audio samples and video samples (images) must be received and played back at regular intervals. End-to-end delay is the sum of all delays in all the components of a multimedia system. Accepted delay is a very subjective matter and is also application dependent. For most applications except conversational multimedia applications, a response time of a few seconds is acceptable. Delay variation is commonly called delay jitter. Delay jitter should be kept very small. For telephone-quality voice and television-quality video, delay jitter should be below 10 ms[4]. Delay jitter values for high-quality stereo audio must be kept particular small (below 1 ms)[5]

We can tolerate some error or loss in digital audio, image and video data. For voice, we can tolerate a bit error rate of 10^{-2}. For image and video, we can tolerate a bit error rate from 10^{-4} to 10^{-6}. We can also measure the error probability by the packet loss rate. When compression is used, the allowed bit error rate will be lower because one bit error may cause decompression error of many bits.

To achieve desired effects, the retrieval and transmission of related media must be coordinated and presented so that the specified temporal relationships are maintained for presentation. Two areas of work are taking place – the first is to develop mechanism and tools to let authors specify the required temporal relationship easily. The second is to guarantee the specified temporal relationships by overcoming the indeterministic nature of communication system.

2.3 Quality of Service (QoS)

Multimedia systems require high-bandwidth, large-storage space and high transfer rate; delay and jitter bound; temporal and spatial synchronization. QoS (Quality of Service) is a contract of a set of requirement parameters negotiated and agreed among the users of a multimedia application and the service and communication provider. When a user needs to start a session, his/her local work station submits a request with the required QoS to the multimedia system. The system will either reject the request or accept the request after some negotiation. Once the request is approved, a contract between the system and the application is signed and the system is about to provide the required QoS. This guarantee can be satisfied fully (deterministic guarantee), a guarantee with a certain probability (statistic guarantee), and no guarantee at all (best effort), which is the traditional system-sharing policy.

QoS can only be guaranteed when the required system resources are properly managed. Each system component should have a resource manager. When it receives a new session request, it will carry out an admission test. If the available resources are sufficient to support the new request and the admission of the new session does not interfere with the existing sessions, the new session will be admitted. Otherwise, a new set of QoS parameters may be suggested to the application based on the available resources. If the suggestion is acceptable to the application, the new session is started. In all other cases, the new session is rejected.

CHAPTER III PRACTICE OF VIDEO MULTICAST OVER THE INTERNET

3.1 Issues in Real-time Multimedia Multicasting

In real-time multimedia (such as video) multicasting, the Internet's transmission resources and end-systems are of high heterogeneity. Ideally, each receiver should receive video that is commensurate with its own capability and the capability of the path leading to it from the source, regardless of the capability of other users. But distributing video using individual feedback-controlled point-to-point streams may result in unrealistic bandwidth requirement when there are a large number of receivers, and lead to great loss and delay. Thus, there is a trade off between bandwidth and granularity of control. A lot of research has been done to find an effective way to distribute multiple multimedia streams to meet the end users' requirements at a lower network cost (i.e., without taking too much bandwidth). When picking up the number and content of the streams, not only should we consider the end users' bandwidth limitation, but also the end users' personal taste (we call it user profile in our context). The first section of the following literature survey talks about adaptive rate control techniques to adjust video traffic characteristics without considering user profiles, the second section talks about techniques that considering user profiles.

3.2 Adaptive Bit-Rate Approaches without Considering User Preferences

Three different approaches to the multicast transmission of digital video are going to be studied in the following context: single and replicate stream, as well as other approaches.

3.2.1 Single Stream Approach

A single encoded video stream is transmitted by the source with feedback being returned from the receivers to the source. The source uses the feedback information to adapt the data rate, as well as the compromise between frame rate and resolution. A potential problem with this approach is the problem of "*feedback implosion*" which can occur if there are a large number of receivers attempting to return feedback to the source.

A representative of the single stream approach is IVS (the INRIA Video-conference System) which is presented by Bolot, Turletti and Wakeman[6]. They described a videoconference system for the Internet that employs a single stream adaptive approach. To adjust the output rate of the video coder, three parameters are considered: the refresh rate, the quantizer and the movement threshold. The specific requirements of the application determine which of the three parameters will be adjusted when adapting the output rate of the encoder. Feedback information is based on packet loss. A reception reports including this information is sent back to the sender through RTP and RTCP.

As we mentioned before, the single-stream approach may cause network congestion and the sender gets overwhelmed. A mechanism called *probabilistic multicast*[7] addresses this problem. A *probabilistic multicast* message is only accepted by members in the multicast group with a certain probability, and only those members respond. Other than that, Bolot, Turletti and Wakeman proposed a *probing mechanism*, which is similar to *probabilistic multicast,* to solicit feedback information in a scalable manner[8].

3.2.2 Replicate-Stream Approach

In this approach, the source sends multiple streams carrying the same video with different quality parameters like frame rate and resolution. Each stream is multicast with receivers being able to choose the stream that is commensurate with their capability by joining the corresponding group. Feedback from the receivers can be used to adjust the data rate of the stream, within certain limits.

A typical implementation of this approach aimed to improve fairness significantly at a small bandwidth cost is **Destination Set Grouping (DSG)**[9]. In DSG, a source maintains a small number of video streams with the same content, but targeting receivers with different capabilities. Each stream is feedback-controlled by its group of receivers**.**

Receivers may move among groups. An ***Intra-stream Protocol*** is used by receivers listening to the same stream to adjust the data rate of the stream within its prescribed limits. Receivers use packet loss rate to indicate *unloaded, loaded* and *congested.* The source adjusts the data rate (through changes in compression parameters "down" if CONGESTED, "up" if "UNLOADED"). An ***Inter-stream protocol*** is used by receivers to change to a higher or lower quality stream as their needs change. Changing may occur when (1) a stream is operating at its low quality end, receivers are either satisfied or want to lower the video quality; (2) a stream is operating at its high quality end, receivers are either satisfied or can handle even better quality; (3) a stream is operating at some point with some receivers capable of handling better quality video but have not been able to move the quality up because other low capability receivers in the same group. ***Stream Advance Solicitation*** is a mechanism to allow high-capacity receivers to leave a group that is rate-controlled by some low capacity members.

While DSG is attractive in resolving the heterogeneity issue, it has the potential problem of network overloading caused by the transmission of multiple replicated streams. *Bandwidth Control*[10] mechanisms can be used to effectively control the effect of stream replication to the point where it does not hinder network operation. The following bandwidth control mechanisms may be used to control the possible adverse effects of stream replication.

- Congestion History Checking

Replication is caused by receivers oscillating back and forth between different data rate streams. Each receiver maintains a *congestion history database* to keep track of congestion caused by its stream group change. An entry is made into the database when a receiver switches from stream group S_0 to S_n. Future switches are accepted or abandoned depending on the time since the last switch.

- Local Area Bandwidth limitation

This method is to limit the number of streams by letting receivers coordinate their stream changes with other receivers in the same locality through a simple exchange of status messages.

The first kind of schemes put a strict limitation to allow no more than n streams over the same subnet. Each receiver multicasts STATUS messages with its current stream number, and keeps a record of active streams by listening to STATUS messages from the other receivers. This scheme is implemented by exchanging current and intended stream numbers through 3 messages: REQ, ACK and SYN. A receiver R can move from stream S_0 to S_n when 1) S_n is already an active stream over the subnet, or 2) there are no more than n-1 streams on the subnet; or 3) R is the only one receiving S_0 on this subnet, otherwise, other receivers of S_0 must also be willing to switch from S_0 to $S_{n.}$

The second category of schemes is developed to control the bandwidth over a large geographical scope. Instead of imposing a strict limitation on the number of streams allowed over TTL, a receiver tries to move from S_0 to S_n with a certain probability. A SYN message is also sent if it really moves, to allow other receivers in S_0 to move as well.

- Overall Bandwidth Limitation

In a network where a source maintains a small number of video streams that carry the same video but each targeted at receivers with different capabilities, the overall bandwidth is the total bandwidth of all streams leaving the source for the different receivers. It is proposed to add a “macro” control at the source that can be used to limit the overall bandwidth needed for multicasting all streams into the network. This paper assumes that the source is unable to calculate the exact bandwidth used by the steams, controlling the overall bandwidth will help minimize problems that may be caused by replicated streams.

The following formula are used to calculate the upper bound, lower bound, and the estimated total bandwidth usage for total number of active streams:

$B_{max} = \sum_i B_i N_i L_i$

$B_{min} = \sum_i B_i(Lmax_i + D_i - 1)$

$B_{est} = (B_{max} + B_{min})/2$

where we use the following notations:

$Lmax_i$ – where the maximum hop distance to its receivers for stream i, calculated using the protocol maximum TTL estimation technique of the DSG protocol[11]

L_i – the average receiver hop distance for stream i

D_i – the number of distinct distances among receivers for stream i

N_i – estimation of the number of receivers for stream i

B_i – data rate transmitted on stream i

B_{max} – upper bound on the bandwidth usage in the network. It is realized when there are no shared links among any of the receivers for any particular stream.

B_{min} – lower bound on the bandwidth usage in the network. It's realized if for a certain stream, all receivers with the same hop distance are on the same subnet and receivers with different distances share links as much as possible.

B_{est} – estimated total bandwidth used. This estimate is far from being accurate; it will be correlated with the actual bandwidth usage

The goal is to keep B_{est} under a certain predetermined threshold β. $\beta = B' * \gamma$, where B' is the estimated overall bandwidth when all receivers are receiving the same stream and the stream is at the highest data rate. Determining γ is a hard problem, which is still under research. The source periodically checks to see if the estimated bandwidth (B_{est}) has increased beyond the threshold value (β), if it does, the request for quality increase is abandoned.

3.2.3 Layered Video Streams Approach

In this approach, the output stream is divided into layers, one *base layer* and one or more *enhancement layers*. The base layer can be independently decoded and provides a low quality of service, while enhancement layers can only be decoded together with the base the layer to provide an enhanced quality of service. The source sends each layer to a different multicast group. Besides the base layer, receivers can subscribe to as many enhancement layers up to their allowed network capability. Video layering can be supported by many video compression techniques. And there are various protocols to implement this approach. In the following context, we are going to present two major protocols. One is the Receiver-driven Layered Multicast (RLM), and the other is the Layered Video Multicast with Retransmissions (LVMR).

- Receiver-driven Layered Multicast (RLM)

RLM is proposed by McCanne, Jacobson and Vetterli[12]. In RLM, the burden of adaptation is moved from the sender to the receivers, resulting in enhanced system scalability. Each video layers is sent to a separate IP multicast group. Each receiver subscribes to a certain set of video layers by joining the corresponding IP multicast group. When a receiver detects congestion, it drops a layer. When there is spare bandwidth available, it adds a layer. Congestion is detected based on the packet loss rate. There is no explicit feedback on whether the current reception level is too low. Thus *join experiments* are carried out by a receiver to find out if it is capable of handling an additional video layer. A failed join experiment may bring congestion to the network. Therefore, a learning algorithm and a join-timer are proposed to minimize the frequency and duration of join experiments, in order to bring down the possibility of congestion caused by join-experiments as much as possible.

RLM proposes *shared learning* for all users to learn from other receivers' failed join-experiments, so that the receivers can add and drop layers indiscriminately. Before a receiver starts a join-experiment, it multicast to the whole group about the specific layer it is going to try to join. This requires each receiver to maintain a variety of state information. Also, the exchange of control information may decrease usable bandwidth. Therefore the shared-learning scheme helps to improve scalability, but with a potential cost of bandwidth usage and message processing overhead to a lot of receivers.

SCUBA (Scalable ConsenUs Bandwidth Allocation)[13] is a related protocol that enables a video source to intelligently account for receiver-interest in their rate-adaptation algorithms. It reflects receiver interest back to the source by using a scalable control mechanism. A “flat delivery” variant is to supplement sender-based adaptation and a “layered delivery” variant is to supplement receiver-based layer adaptation. For “layered delivery”, SCUBA assigns layers by distinguishing more important sources from less important ones, and assigns layers from different sources to network channels with different priorities.

- Hierarchical Rate Control in LVMR

Layered Video Multicast with Retransmissions (LVMR)[14] also uses layered coding over the Internet. This scheme improves the quality of reception within each layer by re-transmitting lost packets and applying an adaptive playback point scheme to help achieve more successful retransmission. It also adapts to network congestion and heterogeneity using a Hierarchical Rate Control (HRC) mechanism.

HRC distributes information among the sender, receiver and the agents in the network so that each entity maintains information relevant to itself. In addition, HRC enables intelligent decisions on conducting join/leave experiments or choosing one of several possible experiments based on minimum state information at the agents.

In order for the receivers to make meaningful decisions on adding or dropping a layer, it is fundamental that results of join/leave experiments done by other receivers have to be multicast. LVMR uses *intelligent partitioning* of the knowledge base and distributes relevant information to the members in an efficient way.

3.2.4 Other Rate-adaptive Approaches

Thin Streams is a solution proposed by Wu, Sharma and Smith[15] for the join/leave problem in layered video multicasting. Each video layer is divided into multiple thin streams of limited bandwidth, and each thin stream is multicast in a separate multicast group. Thin streams help to decrease network oscillation caused by joining and leaving receives, but they increase network-supporting overhead of multicast groups and receive processing overhead synchronizing multiple thin streams.

An application level *gateway* is implemented by *Robinet, Au* and *Banerurjea*[16] to connect adaptive multimedia applications using hierarchically encoded video across ATM and IP networks. The receivers utilize RSVP signaling (in the IP network) and UNI signaling (in ATM network), together with local processing of load information and packet loss information. This information and network capacity information are sent back to the video source to control the bit-rate of the video layers.

3.3 Approaches that Consider User Preferences

The techniques we discussed in the last section focus on solving the problem of how can the end users communicate with the sender effectively and get the multimedia stream that is most close to its maximum bandwidth capability. On the other hand, they ignored that for the end user, bandwidth limitation is not the only factor that decides which stream to choose. Just like each individual person has his own taste, the choices of the heterogeneous end users are impacted by factors like:

- Different hardware and/or software resources available in the workstation
- Different user-level QoS parameters (i.e. user preferences). Such as compromise between low-cost service or high reception quality (which may imply higher cost); or compromise between different QoS aspects, such as frame rate, color, resolution, disturbance through packet losses, etc.
- Different transmission-level QoS parameters (provided by the network) due to specific network architecture and interconnection structure.

The following context is going to present the studies that shows how user preferences have influence on the choice of multimedia streams for the end users.

3.3.1 QoS Management Using MPEG-4/DMIF

In replicate streams or layered stream scenarios, it is necessary to distribute part of the QoS management process and allow each user process to make QoS decisions based on its local context. Bochmann and Yang proposed a design of a teleteaching system, which uses a special paradigm for QoS negotiation[17]. In this paradigm, each source provides several *stream variants* of the same application content in order for the receivers to make a choice based on their QoS parameters. They also explain how the Delivery Multimedia Integration Framework (DMIF) of MPEG-4 can be adapted as a session protocol for such an application. This paper developed a framework for QoS management of tele-teaching applications. It was assumed that different stream variants (same content, but different QoS characteristics) are multicast to the students' workstations and the so-called QoS agent at each workstation may select the stream that is most appropriate after processing the student's preference profile. The structure of the system is shown in Figure 1.

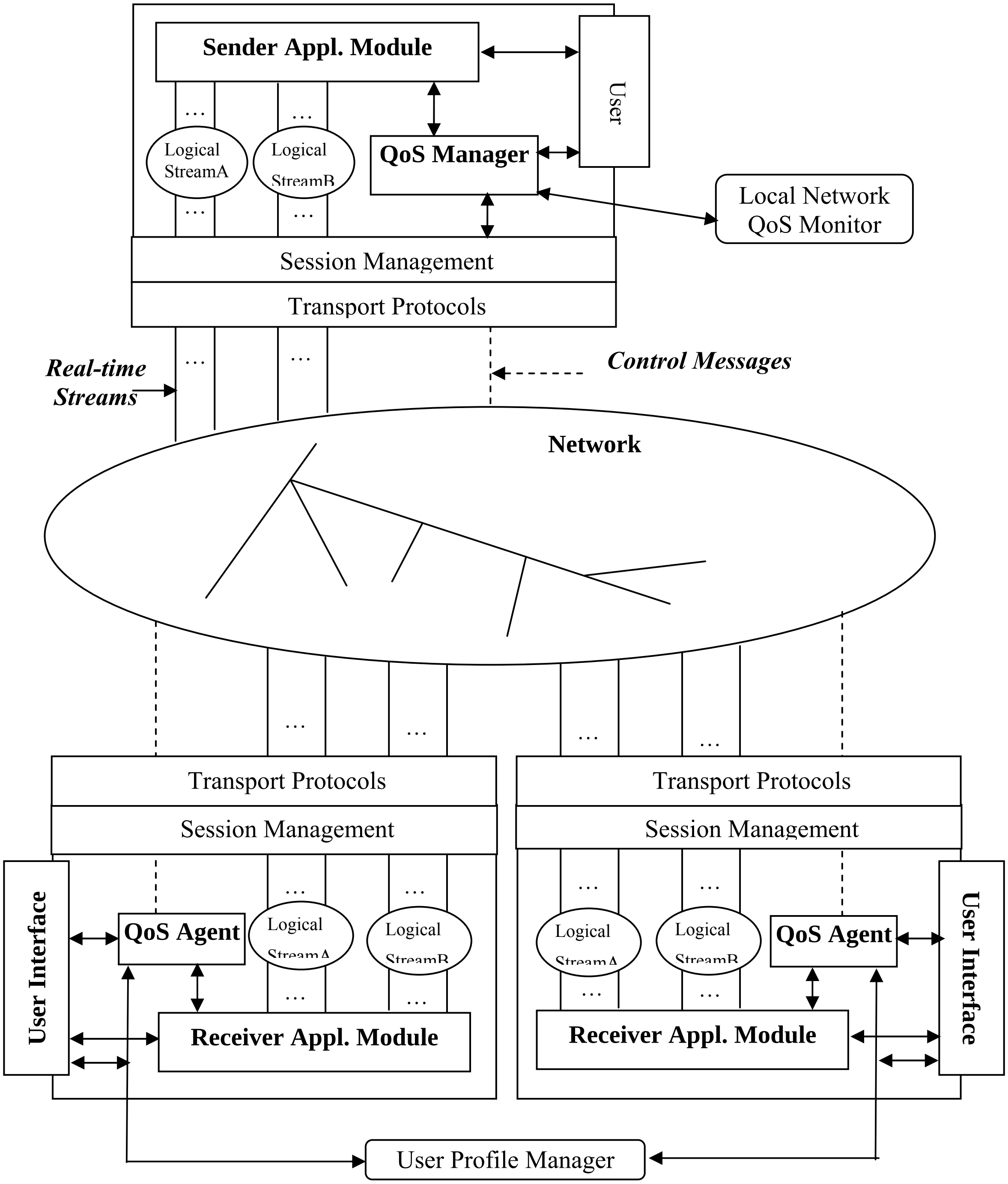


Figure 1. General System Architecture for Teleteaching Application Using MPEG-4/DMIF

The paper introduced the concept of *user preferences* into QoS negotiation, but it also leaves certain issues open, like:

- What are the composition of user preferences, and how can they be quantified in order to be used to determine QoS choice
- What are the appropriate number and content of the stream variants that sender need to multicast to the receivers so that for all end-users, their QoS agents can have ample choices to be able to select the stream best suited for the user's preferences?

Papers related to these issues and further studies in this area will be discussed in following subsections.

3.3.2 Mapping User Level QoS from a Single Parameter

Antony Richards et. al proposed an end-to-end network QoS architecture that introduces the concept of "satisfaction" to quantify the QoS provided by the system[18]. In the paper, they explored the relationship between the Application layer and the User layer, and came up with a translation function that provides the many-to-one mapping between the application specific parameters (such as frame rate and resolution) and the user satisfaction parameters. Likewise, the relationship between the Application layer and the Resource layer (such as bandwidth capacity) is also mapped to the translation function. The approach discussed in the paper enables the system to configure itself by taking into account the relative importance, to the user, of variables such as frame rate and resolution, based on the users' setting of the satisfaction or cost functions. The main proposals of this paper are the following:

- Specifying User Preference

For a single media type, the user has a minimum acceptable quality level (i.e. the lower bound) M, anything below this level brings 0 satisfaction; and an ideal quality level (i.e. upper bound) I, the user's satisfaction reaches 1 at this point and can not be higher.

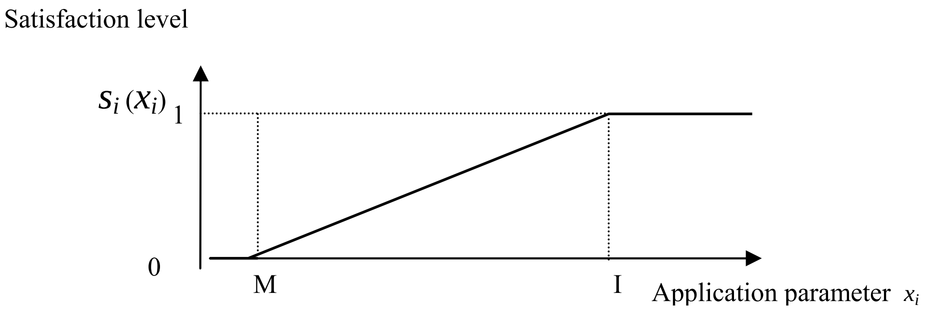
Reception of media stream is also constrained by the user's network resource capacity, indicated by bandwidth limit L. Anything beyond the user's bandwidth limit cannot be decoded by the user and brings 0 satisfaction level.

- Translating a Single Application Parameter to its Component Satisfaction

The component satisfaction S_i for a single application parameter is a function of the value x_i of this parameter:

$$s_i = g_i(x_i)$$

Between the lower bound M and the upper bound I, S_i is a monotonically increasing function of x_i, as shown in Figure 2:



$S_i\,(X_i) = 0$ ***for*** $X_i < M$

$S_i\,(X_i) = (X_i - M)/(I-M)$ ***for*** $M <= X_i <= I$

$S_i\,(X_i) = 1$ ***for*** $X_i > I$

I -- receiver's "ideal" level
M-- Receiver's minimum accepted level

Figure 2: User satisfaction level as a function of single QoS parameter

When there are more than one application parameter, a combined satisfaction S(tot) should be computed as:

$$S(tot) = f(s_1, s_2, \ldots s_n) = n/(\sum(1/S_i))$$

The reason for choosing this formula is that it has the following properties: if the user can not get satisfied at all from one single application parameter (S_i=0 for some *i*), the combined satisfaction is 0. On the other hand, if the user gets fully satisfied from every application parameter (S_i = 1 for every *i*), the combined satisfaction level is also 1.

- Choosing an Application Level Operating Point within the Bandwidth Limit

Within the bandwidth limitation of a given user, there can be multiple value combinations of some set of application parameters. For example, we can get many combinations of frame rate and resolution by changing the values of the parameters yet restrain the needed bandwidth within a given limit. From the satisfaction function above, those combinations result in different levels of user satisfaction. We should therefore search for the combination that maximizes the satisfaction level within the bandwidth limit[19].

3.3.3 QoS Negotiation Based on Device Capabilities and User Preferences

Most multimedia applications require the user to select certain QoS parameters. But as the range of hardware, the software, and media types are getting wider, the need for an automated system to do the selecting process is increasing. K. El-Khatib, X.He and G.v.Bochmann developed a scheme for representing and storing the user preferences in a user profile[20]. The device capabilities and capacities are also stored in a device profile. The design of automated QoS-aware applications is presented.

Each end user has a device profile and a user profile. The device profile may cover the hardware platform (CPU model, the size of the memory, microphone…), system software (operating system, list of audio and video encoders…) and applications (JMF, Vic, Vat…) available on the device. The user profile may capture the personal properties and preferences of the user, such as the user's personal information, his/her choice of receiving/sending audio and video such as the frame rate, the resolution, the audio quality and the degree of preference for each media type. The device and user profiles are expressed in the Composite Capabilities/ Preferences Profile (CC/PP)[21] using the Resource Description Framework (RDF), so that the selection of media types and QoS parameters can be done automatically when the information reaches the sender.

There are enormous combinations of the QoS parameters at the sender side. Optimization has to be done based on the two profiles from the multiple end-users. Each combination is run through the algorithm that is derived from Richards' mapping method (introduced in Section 3.3.2, to quantify the user preferences and device capability into a user satisfaction level). The combination that produces the best satisfaction level is sent out through multicasting.

In this paper, the authors also presented an extension to the framework presented by Richards to enable weight assignment to different parameters and participants. As an example:

$$\begin{aligned} S_{tot}^{user} &= f_{comb}(s_1, \ldots s_n, w_1, \ldots, w_n) \\ &= n * avg(w) / \sum(w_i/s_i) \end{aligned}$$

where w_i is the weight for the individual satisfaction s_i

El-Khatib's paper incorporates the teleteaching architecture introduced in Section 3.3.1 and Richards' mapping algorithm introduced in Section 3.3.2. It gives a detailed study on how to generate the user profile and device profile, and what QoS should be selected. It also introduces an effective way to transfer those data. Furthermore, it extended Richards' framework by enabling weight assignment. Yet in a network that has a large number of end-users, it is unrealistic to deliver the profiles of all end-users' to the sender. Besides, the sender has to optimize the number of stream variants that are going out. These are issues that are addressed in this thesis.

CHAPTER IV RECEIVER SATISFACTION DRIVEN QOS MANAGEMENT FOR MULTICAST APPLICATIONS

4.1 Open Issues

In Section 3.3, we discussed the earlier works that have been done to develop QoS management approaches that involve user preferences. Section 3.3.1 presents the using of the DMIF session management protocol for multicasting applications. The protocol aimed at distributing part of the QoS management process between source and receivers; The QoS manager in the source node determines the list of potential steam variants for each logical multimedia stream, and informs the receivers about these variants. Based on the user profile, the QoS agent at each receiver node selects the stream that gives the highest level of appreciation to the receiver. DMIF is used for session management of a tele-teaching application including different QoS alternatives for the participating users.

As proposed in Section 3.3.2, each user specifies the *minimum acceptable* and *ideal* value for each QoS parameter (such as frame rate and resolution). A satisfaction function maps the actual QoS value of the user satisfaction onto a range between 0 and 1. QoS parameters that are higher than the ideal value will not increase the satisfaction level, similarly, any parameter value that are lower than the minimum will not reduce the satisfaction level below 0. The overall satisfaction level for all users is calculated as the weighted average of satisfaction for each individual receiver. Different users have different bandwidth limits; the users are classified according to their bandwidth limits. Bandwidth limit and user preference values are applied to the satisfaction level function to determine which multimedia stream is best for the receiver.

Section 3.3.3 presented an architecture for taking personal preferences into account in the context of personal telecommunication services. Each user has a profile that covers his QoS preferences and policies for handling incoming and outgoing calls. Based on user profile information from all communicating parties and the candidates' device limitations, the middleware architecture is presented to select the device and QoS parameters that best suit all the parties. Even though it is about end-to-end personal telecommunication, the way it stores user profiles in a machine-understandable format and uses it to optimize QoS parameters can also be applied to multicast applications.

As discussed earlier, a certain number of important system aspects have been defined, such as: (a) the system architecture which uses DMIF as session management control panel; (b) a function to translate user preferences into user satisfaction level; and (c) a machine-understandable format of user and device profiles to allow information communication and the automatic calculation of the user satisfaction from those profiles. Still, there are some open issues regarding the sender-side intelligence and the communication efficiency between sender and receivers. In the earlier works, it was assumed that the sender manually sends out all combination of QoS parameters (for the same application content) so that the QoS agent at the receiver side have enough choices to apply its intelligence to select the stream whose combination best suits its user. In reality where there are a variety of QoS parameters and each parameter can have a wide range of values, it is almost impossible to spell out all the combinations. Even if this was possible, delivery and storage of this information could be a problem.

It would be nice to have a QoS negotiator at the sender side to run the intelligence. It needs to have some knowledge about the receivers to determine the value of the QoS parameters of the stream variants to be sent. It would determine not only the value of the QoS parameters, but also the number of stream variants that are sent out. In order to be in a position to make meaningful decisions, the QoS negotiator should collect the user profiles that contain user preferences and device information from all the QoS agents in different receivers. In the case of a small group of receivers, it is feasible for each group members to send their user profile. But when there are a large number of users distributed over a wide geographical area, this approach is not feasible because of the large number of reports to be received by the sender. We therefore propose to partition the receivers into different classes. Since receivers with the same bandwidth limit are likely to have similar end-system requirements and QoS preferences, we tend to partition the receivers based on their bandwidth limits, and each class sends one copy of a representative user profile that represents, approximately, all the receivers in this class. The partitioning could also be done by region in addition to bandwidth limit if there is a very large multicast group; each class would then send several representative group profiles. We are going to discuss the detail of this grouping (partitioning) approach in Section 4.2. The work of this thesis is meant to present some solutions to these problems.

4.2 Receiver-Satisfaction Driven (RSD) Solution and System Architecture

The solution introduced in this section is meant to find the optimal number of streams and the optimal QoS parameters for these streams that produce the best **satisfaction level for all users,** which is a quantified QoS calculated based on user preferences and bandwidth limits of all users. Before we start to unwrap the different aspects of the solution, let us take a look at the basic system architecture shown in Figure 3:

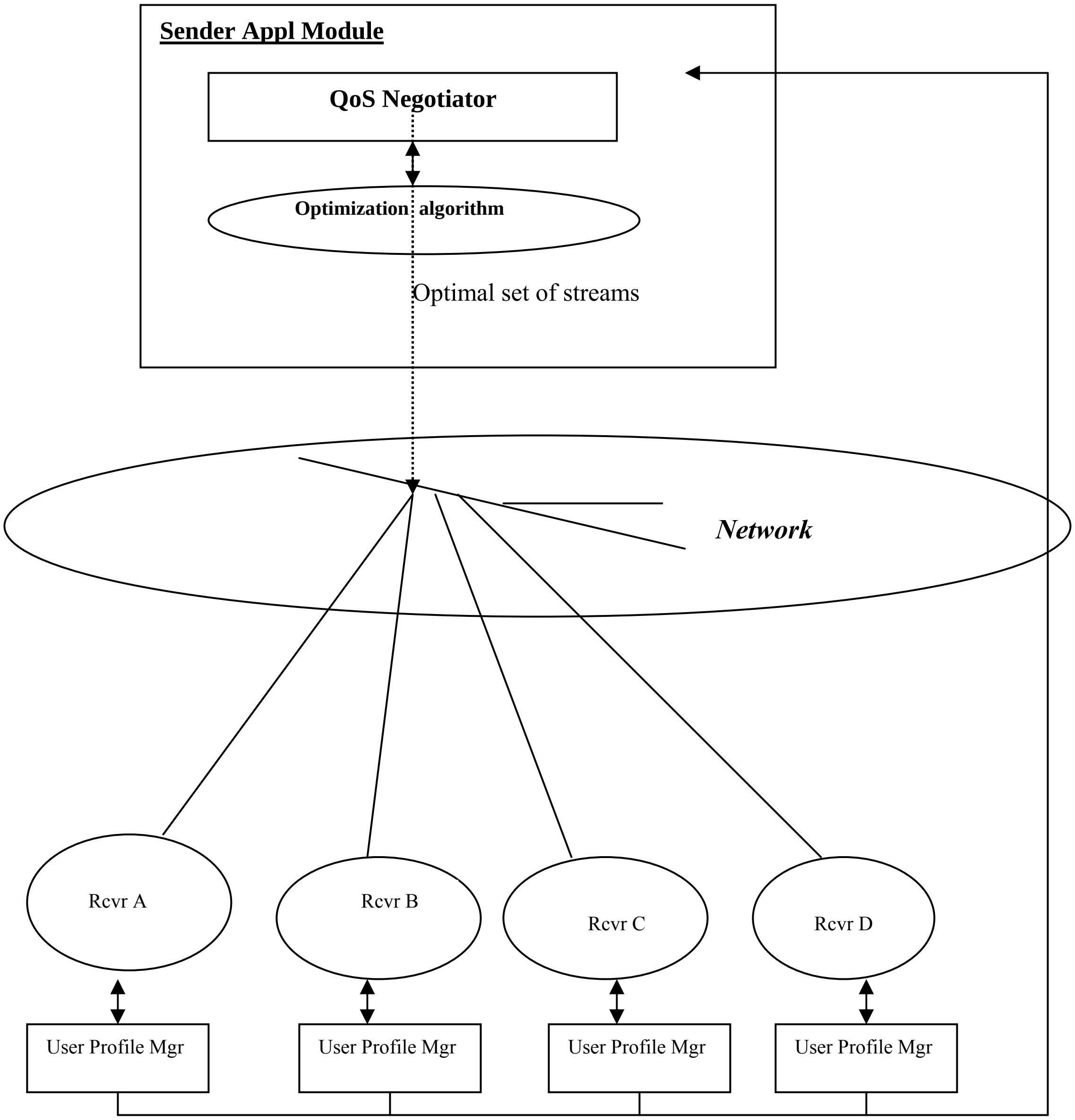


Figure 3. Basic System Architecure of Receiver-Satisfaction Driven (RSD) solution:

This architecture is characterized by the following points:

- Each receiver has its own QoS preference file.
- Based on the user profiles, the centralized QoS negotiator at the sender side decides whether to send out one channel, two channels, or more channels, as well as the values of the QoS parameters for each of those channels
- The user profile managers at the receiver side receive the information about the available stream variants from the sender, and then apply the satisfaction algorithm to select the stream variant (channel) that best suits the receiver.

4.3 Optimizing User Satisfaction for Small Number of Users

Section 4.2 presented the system architecture and how the system works. In this section, we are going to introduce an optimization algorithm derived from Richards' paper[22] introduced in Section 3.3.2. An example will also be given on how the optimal QoS parameters are calculated.

4.3.1 Optimization Algorithm

For a fixed number of channels, the purpose of the optimization algorithm is to select the QoS parameters for each of the channels that maximize the overall user satisfaction for all users. The input of the optimization algorithm are the user profiles and the number of channels; the output are the QoS parameters matching the number of channels. The goal is achieved by optimizing a satisfaction function, which takes the user profiles as input, and calculates the overall satisfaction for all users.

The optimization algorithm can be broken down into the following steps:

(a) Define the satisfaction $\boldsymbol{S_i}$ for user *i* as a function of all QoS parameter vectors and data from the user profile

$$S_i = f(Fr, Res, Pro_i)$$

Where Fr is the vector of frame rates for the different streams $(Fr_1, Fr_2, \ldots Fr_k)$, and k is the number of channels. Similarly, Res is the vector of resolutions of the streams. Pro_i is the user profile for user i, which includes the minimum value *(Min)* and ideal value *(Ideal)* for each QoS parameter, for example, Min_i^{Fr} indicates the minimum frame rate for user i, $Ideal_i^{Fr}$ indicates his ideal frame rate. Pro_i also includes the bandwidth limit $(BwLmt_i)$ for user i. An example of user profile is shown Section 4.3.2.

As we discussed in Section 3.3.2, the satisfaction of user i, with regard to a single QoS parameter x, is a function of x, the user's minimum value of x (indicated as Min_i) and the user's ideal value of x (indicated as $Ideal_i$).

$$\begin{array}{ll} S_x\ (x, Min_i, Ideal_i) & \\ =0 & \text{if } x <= Min_i^x \\ = (x - Min_i) / (Ideal_i^x - Min_i^x) & \text{if } Min_i^x < x < Ideal_i^x \\ = 1 & \text{if } x >= Ideal_i^x \end{array}$$

In our case, we have two QoS parameters for a stream, say, namely frame rate and resolution. Using the formula in Section 3.3.2 for the joint satisfaction in respect to these two parameters, the satisfaction of user i with respect to the stream l is:

$$S_{i,l} = 2/(1/S_x(Fr_l, Min_i^{Fr}, Ideal_i^{Fr}) + 1/S_x(Res_l, , Min_i^{Res}, Ideal_i^{Res}))$$

However, when the bandwidth required by stream l is larger than the user's bandwidth limit $BwLmt_i$ then we have, $S_{i,l} = 0$.

Since the QoS agent of the receiver station selects the best stream among the available streams, the satisfaction of user i will be given by

$$S_i (Fr, Res, Pro_i) = \underset{i}{Max}\ (S_{i,l})$$

(b) Define the Overall Satisfaction Function for All Users

The overall satisfaction for all users is a function of the individual satisfaction of all receivers of the applications. It is obtained by some kind of averaging. We assume I general that each user may have a weight factor w_i. However, in most cases, the weight factors will all be equal.

We consider two methods of averaging:

(i) Arithmetic average

$$\boldsymbol{S} = \sum_{i} (S_i * w_i)/\sum_{i} w_i$$

(ii) Collaborative average

$$\boldsymbol{S} = \sum_{i} w_i /(\sum_{i} (w_i/S_i))$$

The collaborative average is the appropriate averaging formula to be used in the case that it is important that all participating receivers obtain a non-zero satisfaction, for instance in the case of collaborative work within a teleconference setting. The collaborative average becomes zero when the satisfaction of at least one receiver becomes zero. In the case of using the arithmetic average, the optimum overall user satisfaction may still become optimal when some users obtain completely unsatisfactory quality (satisfaction zero)

(c) Optimization: find the values for the frame rate (Fr) and resolution (Res) vectors that yield the maximum value for ***S***.

4.3.2 Selecting An Appropriate Number of Channels

When considering how many channels are good enough to generate a reasonable satisfaction level for all users, ideally, it would be one channel for one user. Due to network resource constraints, the bandwidth limits exist not only at the receiver node, but also at server access link to the network. Too many channels may cause congestion at the server access link and lead to packet loss and thus lower satisfaction level. If we apply the layered stream approaches discussed in Section 3.2.3, we may achieve to transmit a larger number of stream variants (i.e. channels) without laying too much burden on network traffic. Still, the total bandwidth needed for the channels should not exceed the source bandwidth limitation.

Our simulation results in Section 5 show that the average satisfaction rises when we increase the number of channels. When we partition the users into different classes according to their bandwidth limit, the simulation results show that the satisfaction is good when one channel is dedicated to one class, which means the number of channels is the same as the number of classes.

4.3.3. Example Configuration

In this section, we present a simplified example to show how the RSD approach picks up a pair of stream variants that gives the best overall user satisfaction. In this example, we assume no grouping, and no automatic optimization of QoS parameters by the sender. Those issues will be discussed further in our simulation study in Chapter 5. In this simplified example, we will consider only 4 users, who exchange their profile information directly with the QoS negotiator at the sender side, and the average of the satisfaction of the 4 users is the selection criteria.

<u>User population profile</u>

Suppose there are four receivers in the network, their user profiles are as follows:

Rec._ID	**BW_Limit**	**Fr_Min**	**Fr_Ideal**	**Res_Min**	**Res_Ideal**
A	2400	3	9	192 (160X120)	1200 (400X300)
B	4200	4	8	243(180X135)	2187 (540X405)
C	8000	11	16	432 (240X180)	2700 (600X450)
D	Unlimited	20	35	768 (320X240)	4800 (800X600)

Rec._ID	– Receiver ID
BW_Limit	-- Bandwidth limit in terms of kbps
Fr_Min	-- Acceptable frame rate level
Fr_Ideal	-- Ideal frame rate level
Res_Min	-- Acceptable resolution level, indicated by (horizontal_resolution x vertical_resolution)/100
Res_Ideal	-- Ideal resolution level, indicated by (horizontal_resolution x vertical_resolution)/100

We assume that the sender will send two stream variants and will select one of the following three pairs of streams:

Pair_ID	Fr_Ch1	Res_Ch1	Fr_Ch2	Res_Ch2	Bw_Ch1	Bw_Ch2
X0	11	308	15	1200	1478	7855
X1	7	1000	21	3600	3055	32989
X2	8	2000	14	2500	6892	15273

Stream_ID -- ID of the pair of streams
Fr_Ch1 -- Frame rate for channel 1
Res_Ch1 -- Resolution for channel 1
Fr_Ch2 -- Frame rate for channel 2
Res_Ch2 -- Resolution for channel 2
Bw_Ch1 -- Bandwidth needed for channel 1, calculated as Fr_Ch1 * Res_Ch1 * 0.4364 (the compression constant)
Bw_Ch1 -- Bandwidth needed for channel 1, calculated as Fr_Ch1 * Res_Ch1 * 0.4364

Determination of user satisfaction levels

We apply the satisfaction function introduced in Section 4.3.1 to calculate the user satisfaction. Here is an example of calculating the satisfaction level of user A with respect to X0 :

$$S_A(X_0) = Max\ (\ S_A(Chnl1),\ S_A(Chnl2)\ )$$

The user can choose among the two streams of X0, and since bandwidth limit for user A is 2400, which is lower than bandwidth requirement for the second channel, $S_A(Chnl2) = 0$.

$$S_A(Chnl1) = 2/[(9-3)/(11-3) + (1200-192)/(308-192)] = 2/[0.75+8.69]$$
$$= 0.21$$

Therefore, $\boldsymbol{S_A(X_0) = Max\ (\ 0.21,\ 0\ ) = 0.21}$

Using the same approach, we can calculate the user satisfaction for user A, B, C and D with respect to X0, X1 and X2, and obtain the following results:

Stream ID	Sat_A	Sat_B	Sat_C	Sat_D	Sat_Overall
X0	0.21	0.065	0.48	0	**0.19**
X1	0	0.51	0	0.12	0.16
X2	0	0	0	0	0

Sat_”X” -- The satisfaction level for receiver “X”

Sat_Overall -- The overall satisfaction level for all receivers, calculated by the average value of Sat_A, Sat_B and Sat_C

We see from the table above that the best overall satisfaction is 0.19, which is provided by the pair X0.

4.4 Characterizing User Classes

In a context of a large number of users receiving the application from one source, the source link bandwidth limitation may not allow every user to submit their user profiles. Receivers can be classified into different user classes based on factors such as the geographical region and bandwidth limit. Each class can summarize the user profiles into one or several representative group profiles, which contain the same criteria as that in the user profile. The source does the optimization based on the representative group profiles, and selects the QoS parameters that can maximize the satisfaction for all classes. A graph below shows the structure of user class characterization.

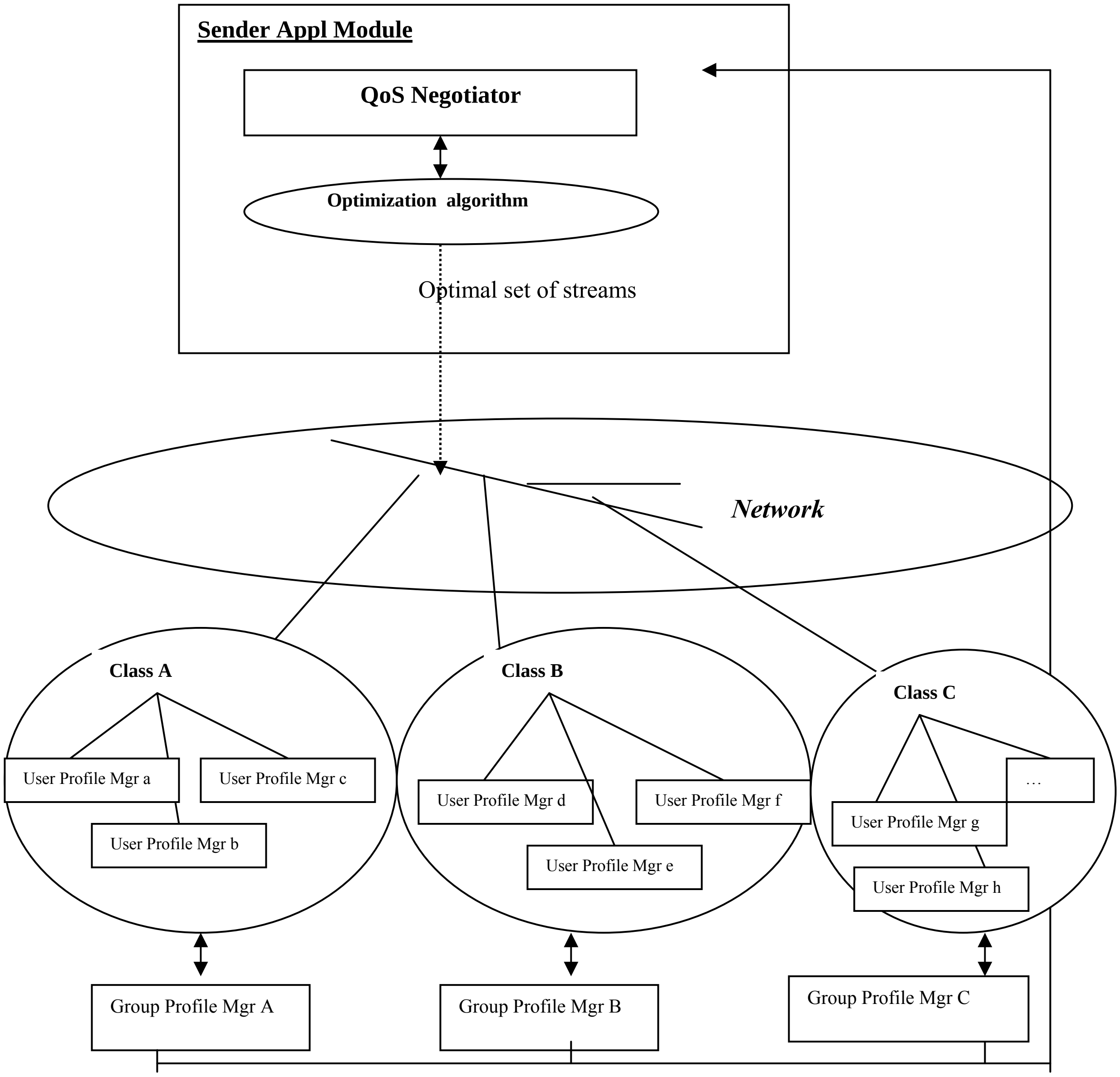


Figure 4. System Architecure of (RSD) Solution with Characterizing User Classes

When we compare Figure 4 with Figure 3, we see in Figure 4, the receivers are partitioned into different classes. There is a “Group Profile Manager” for each class. The

Group Profile Manager collects the data from the user profile managers in its group. It generates “representative group profiles” representing all users of that group, and then sends this to the source. Since there may not be the same number of users in each class, the large classes can be partitioned into small classes, and generate multiple representative group profiles. The number of group profiles for each class can be proportionate to the number of users in that class. Or, to make it simple, each class generate one representative group profile, and attach the number of users as a weight factor when sending the group profiles to the source. The source does optimization using the group representative profiles and the weight factor.

Characterizing user classes may encounter certain degree of information loss. It is very important that the representative group profiles reflect the user preferences of the whole group, so that the selected stream is not too different from the real optimal values. If the representative group profile only covers some extreme cases among the receivers, streams selected based on the representative group profile that satisfy that profile completely, may still get very poor satisfaction from many individuals in that group.

In the following context, we are going to introduce three variants for determining the group representative profiles. We call them mid-point value, average value and boundary value methods. We define the following variables to represent the parameters in user profiles:

Values in the user profile for user *i:*

$frMin_i$ – Minimum frame rate
$resMin_i$ -- Minimum resolution
$frIdeal_i$ –- Ideal frame rate
$resIdeal_i$ –- Ideal resolution

Values in the representative group profile for group:

gpFrMin – Representative minimum frame rate
gpResMin -- Representative minimum resolution
gpFrIdeal -- Representative ideal frame rate
gpResIdeal -- Representative ideal resolution

- **4.4.1 Mid-Point Method**

In this method, the preferences of the group are the mid-point (i.e. (max+min)/2) of the maximum and minimum values of the group.

$$gpFrMin = (\max_i(frMin_i)+\min_i(frMin_i))/2,$$

$$gpFrIdeal = (\max_i(frIdeal_i)+\min_i(frIdeal_i))/2$$

$$gpResMin = (\max_i(resMin_i)+\min_i(resMin_i))/2$$

$$gpResIdeal = (\max_i(resIdeal_i)+\min_i(Ideal_i))/2$$

* The max( ) and min( ) indicate the maximum and minimum value of the above symbol sets in the user class.

This method is applied based on the assumption that the user preferences in the group are normally distributed, and the variance should not be too big. Our simulation studies show that if there are some extreme values, like an extremely large ideal value or extremely small minimum value, the group profile generated by this method cannot be very representative for this group. In this case, streams obtained by this approach may not meet the requirements for a majority of the users in the group.

- **4.4.2 Average Value Method**

In this method, the parameters of the group profile are the mean value of these parameters over all users in the group. We write *n* for the number of receivers in this group

$$gpFrMin = (\sum_i(frMin_i))/n$$

$$gpFrIdeal = (\sum_i(frIdeal_i))/n$$

$$gpResMin = (\sum_i(resMin_i))/n$$

$$gpResIdeal = (\sum_i(resIdeal_i))/n$$

This approach is based on the assumption that receivers that have the same bandwidth limit are more likely to have close preferences, which is true in many cases. My simulation studies show that in a group that has a very small number of users, the user preferences could also be distributed divergently, which indicates a high variance. In such a case, this approach may not be very representative. In this case the streams obtained by this approach may not generate good satisfaction levels for a majority of users in the group.

- **4.4.3 Boundary Value Method**

In this method, the minimum parameters of the group profile is the maximum of this parameter over all individuals in this group, and the maximum parameters of the group profile is the maximum of this parameter over all individuals in this group

$$\mathrm{gpFrMin} = \max_i(\mathrm{frMin}_i)$$

$$\mathrm{gpFrIdeal} = \max_i(\mathrm{frIdeal}_i)$$

$$\mathrm{gpResMin} = \max_i(\mathrm{resMin}_i)$$

$$\mathrm{gpResIdeal} = \max_i(\mathrm{resIdeal}_i)$$

This method expands the preferences of the representative to cover every member's preferences in this group. My simulation studies show that streams obtained by this approach may not be able to generate the best satisfaction level for the majority of members, but at least the stream is acceptable by most users.

CHAPTER V SIMULATION STUDY AND RESULTS

5.1 Simulation Review

In the last chapter, we defined the Receiver Satisfaction Driven method including the optimization algorithm and three different approaches of user class characterization. We experimented with this RSD approach in a simulation environment including a population of simulated users. By comparing the results for different cases, we will be able to answer the following questions, as discussed further on in this chapter:

(1) Is it easy to find the optimal QoS combination?

(2) How does the user satisfaction vary with the number of channels and how many channels are required to obtain a "reasonable" user satisfaction?

(3) How effective are the three methods of characterizing user classes, and how do the results differ from the case that no grouping takes place?

We present in this Chapter the simulation results when the sender send out one, two three and four streams that have the same content, but different QoS parameters. The same sets of simulation are also done with partitioned user classes.

Our simulation study shows that there is limitations in MATLAB fminsearch ( ) functionality. Due to the bandwidth limit for each class, and the user's minimum and ideal values for each QoS parameter, our optimization algorithm is not linear nor quadrant, fminsearch () can only find the local optimum that is around the initial value. In our simulation, we resolve this issue by running multiple optimizatons with uniformly distributed initial value, then pick up the best value among the set the local optimum as our global optimum. An example of the multi-test results will be presented in Section 5.3. As for the user population, as we mentioned before, the users are so diversified due to various network conditions, end-system equipments, and even geographical regions, it is very difficult to simulate a user population that covers all realistic user preferences. In our simulation, we simulated user populations of type "A" that covers four classes of users with low, medium, high and unlimited bandwidth limitation (detail will be shown in following Section). Based on this simulated user population, our simulations results show that the overall user satisfaction level increases with increasing the number of channels. When the number of channels reaches the number of user classes, the overall user satisfaction level is the best, which is about 75% -- 85%. On the other hand, the average satisfaction for 1-channel is between 25%-30%; for 2-channel, between 50%-60%; and for 3-channel, between 65%-75%. Among the three ways of grouping, the boundary-value method produces the best results in terms of user satisfaction level. Since the user class characterization summarizes the preferences from all individual users, it suffers a certain degree of information loss. The satisfaction with user classification is slightly lower than the corresponding cases without user classification. But generally speaking, the results between the two are close enough to show that the representative group profiles of the classes represent well the user preferences of the individual users in its class.

In our simulation, we generate a user population of 20 users based on the criteria presented in Section 5.3, which categorize all users into four classes. The same simulations have been run 20 times with different randomly chosen user populations to show how stable the simulation results are.

5.2 User Sample Modeling

The teletaching application involves multicast to a large number of receivers that are of high heterogeneity. Due to divergent geographical distribution and available network resource, it is very hard to predict a general pattern of all receivers in the network. In our work, we simplified our users to 4 classes with the bandwidth limits of 2400, 4200, 8000 kbps and unlimited.

The following tables show the detailed information of the user population type “A” that was used for most of our simulations. The frame rate is measured in frames/second, and resolution is measured by the product of horizontal and vertical resolution divided by 100. For each class, the user preferences are uniformly distributed within a range indicated in the table. For example, the minimum frame rate for group 1 ranges from 1 to 3, means that the minimum frame rate of users in group 1 range from 1 frame/second to 3 frames/second, and they are uniformly distributed. The bandwidth is calculated as the product of frame rate and resolution multiplied by a constant 0.4364

Due to the heterogeneity of the users, the simplified user model may not be able to reflect a real situation. But it provides reasonable information needed for the simulation.

Table 1. Simulated user population profile type A:

	frMin	frIdeal	resMin	resIdeal	bwLmt
Class 1	1--3	4--9	27--243	243--2187	2400
Class 2	4--7	8--13	108--432	432--2187	4200
Class 3	8--11	12--17	261--1044	1044--2700	8000
Class 4	12--15	16--21	432--1044	1044--3888	unlimited

frMin - Minimum frame rate
frIdeal - Ideal frame rate
resMin -- Minimum resolution
resIdeal - Ideal resolution
bwLmt -- bandwidth limit

The frame rate is measured in frames/second. Resolution is measured by the product of horizontal and vertical resolution divided by 100.The relationship between resolution values in above table and the horizontal and vertical resolution is presented in the following table:

Indicator	27	243	2187	108	432	261	1044	2700	3888
Horizontal X vertical	60*45	135*180	540*405	90*120	180*240	140*186	280*373	450*600	540*720

User population profile type "A" is also presented in graphical format in Figure 5. The gray boxes indicate the range of the minimum values (frMin and reMin) for 4 classes of users. The black boxes indicate the range of ideal values (frMax, resMax) for 4 classes of users.

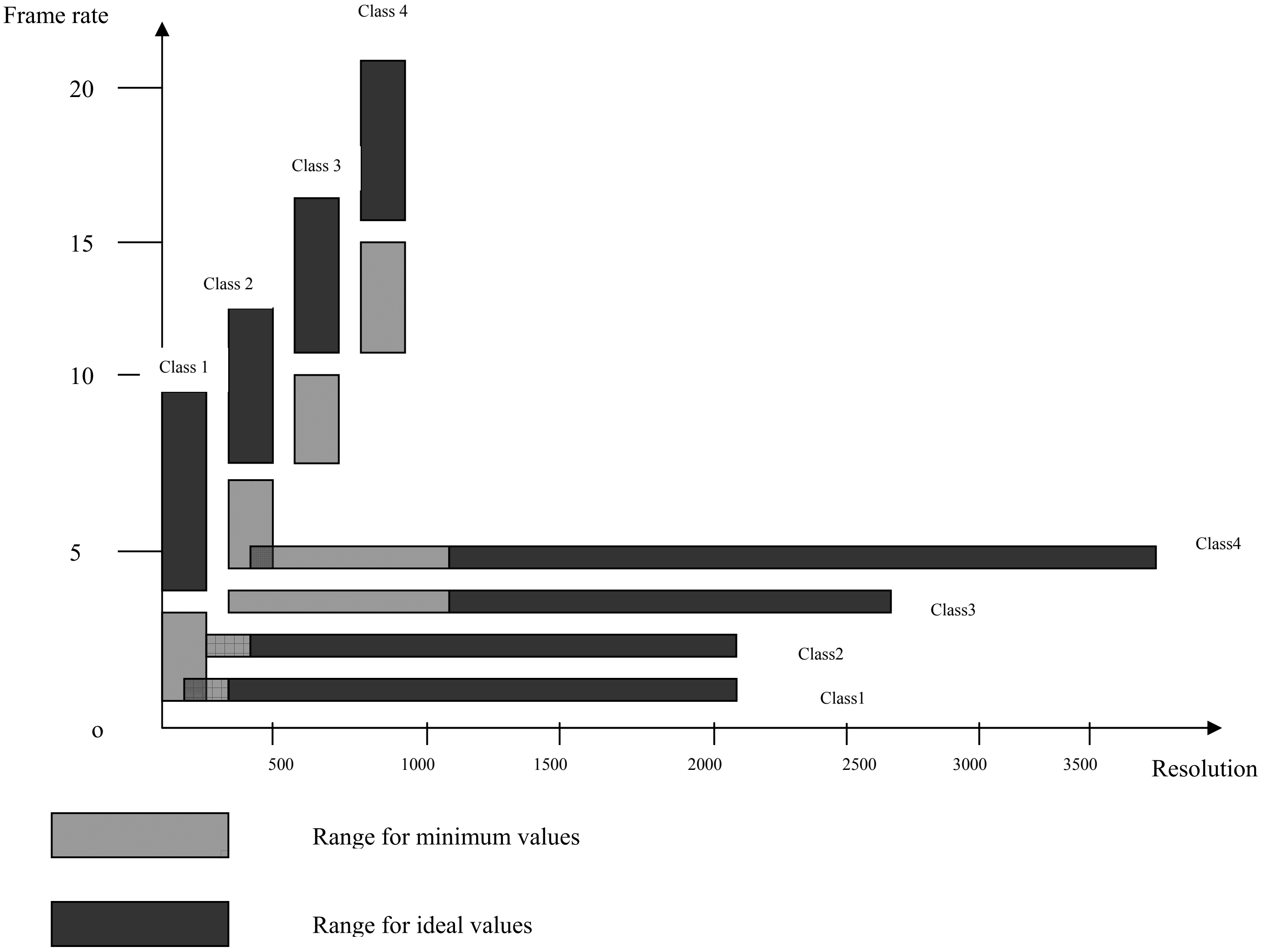
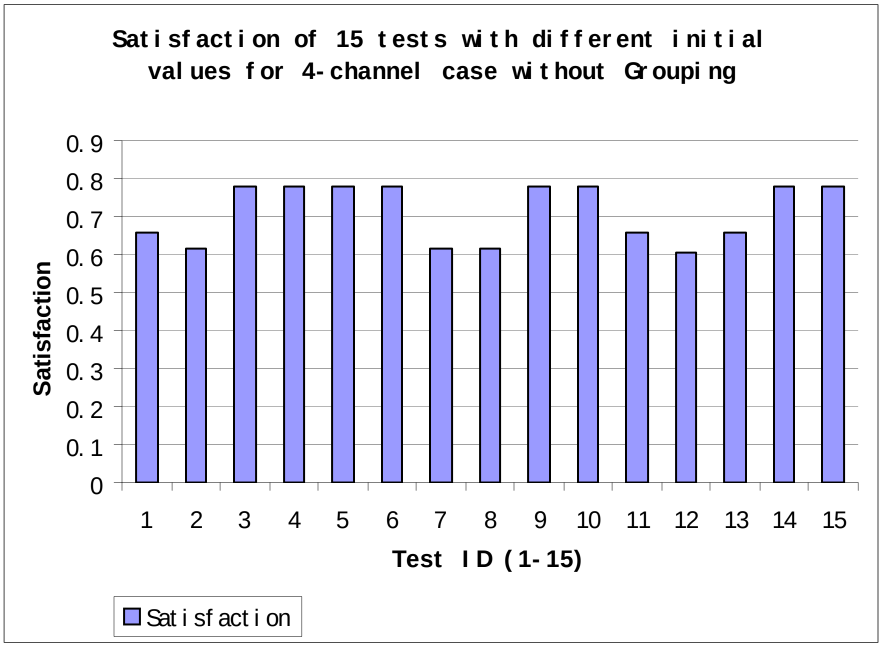


Figure 5 Graphical presentation of the simulated user population profile type A

5.3 Restriction of MATLAB fminsearch Method and Solution

Simulation in this project has been programmed with MATLAB. As to the optimization part, we use the fminsearch[23] function defined in the MATLAB Optimization Toolbox 2.1 to help find the QoS parameters that give the best satisfaction level for all users. `x = fminsearch(fun,x0)` starts at the point x0 and find x where the function `fun`( ) has a local minimum. x0 and x can be scalars, vectors, or matrices. In our case, x is a vector indicating the QoS parameters of the streams and fun ( ) is the negative satisfaction function which has x as argument and also depends on the user preferences and bandwidth limits. x0 is a vector of QoS parameters such as frame rate and resolution that are picked up randomly. x is the optimal QoS parameter set found by fminsearch( ). As shown in Figure 2 in Section 3.3.2, the user satisfaction is one when the QoS parameters reach its ideal values and is always 0 when the QoS parameters are below its minimum values or when the stream exceeds the user's bandwidth limit. As a result, the satisfaction function is not a linear function, nor is it smooth function. Using fminsearch ( ) with random x0 can only find its local optimum around the x0 area, which in most cases, is not necessarily the global optimum within the user accessible range.

In order not to be restricted by the limitation of the program and to reach the global optimum as closely as possible, we run the optimization function 15 times with uniformly distributed x0, and pick up the best satisfaction level found together with its associated frame rate and resolution. Using the 4-channel case, non-grouping as an example, Figure 6 presents the distribution of the satisfaction for the 15 tests. From Figure 6 and the table in Appendix 1, we see that the maximum value is found in 7 tests out of 15, and the QoS parameters in the 7 tests are the same, which means that by running more number of tests, the probability of finding the global optimum is increased.



Minimum satisfaction:0.6033 Maximum v: 0.7803

Average satisfaction: 0.7112 Standard Deviation: 0.078

Figure 6 Maximum satisfactions found in 15 tests with different initial values for the 4-channel case without grouping

5.4 Simulation Results without Grouping

1, 2, 3 or 4 channels of multimedia streams are sent out from the distribution center (the sender). The different streams provide users the same application with different QoS. In order to verify the credibility of our simulation model from a statistics point of view, 20 sets of tests have been done with different randomly chosen user populations, each including 100 users according to the user population profile type “A” shown in Figure 5 and Table 1. For each set of test, the 100 users consist of 25 users of class 1, 25 users of class 2, 25 users of class 3 and 25 users of class 4. The minimum and ideal appreciation values for frame rate and resolution in the profiles are chosen randomly from a uniformly distribution over the value ranges shown in Table 1.

As shown in Figure 7, even though there are small fluctuations, the satisfaction levels of the 20 tests for each number of channels are pretty much close to one another. The difference between the results for those different test runs are therefore due to the statistical variation of the user profiles of the test user population. The results of these simulations will be discussed further in Section 5.4.1.

Before we compare the simulation results between different numbers of channels, let us assume that there is only one class of receivers – class 1, and they are expecting a single multicast stream. We have plotted the relationship between frame rate, resolution and user satisfaction for class 1 for a single channel in 3-D graphic format in Figure 7, to see how the user satisfaction changes with the changes of the resolution and the frame rate values.

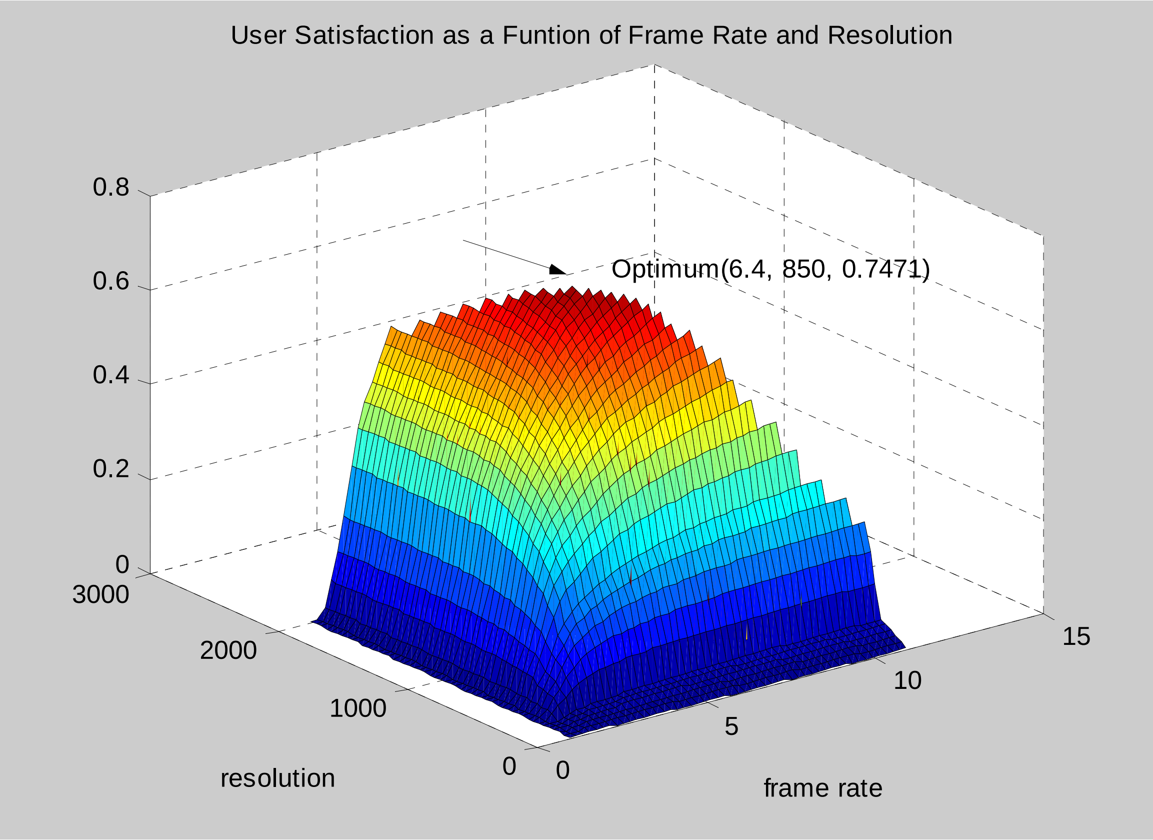


Figure 7 3-D relationship between frame rate, resolution and user satisfaction level for a single stream

From the figure, we can see a trade-off between frame rate and resolution. At the same satisfaction level, a higher frame rate is always associated with a lower resolution. In the following section, we are going to compare the satisfaction levels generated from different number of channels to see how many channels are good for generating a "reasonable" user satisfaction level before we characterize the user classes.

5.4.1 Comparing the Average Satisfaction for Different Number of Channels

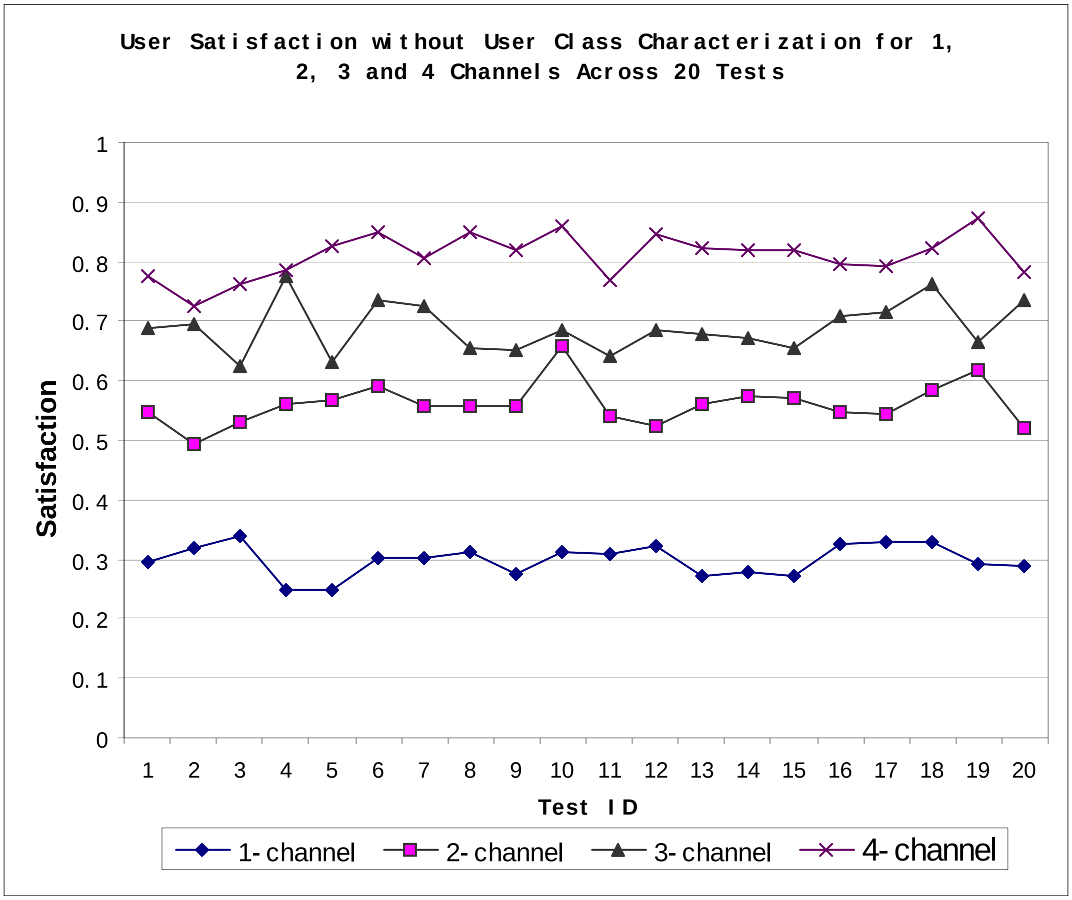


Figure 8. User satisfaction level for 1, 2, 3 and 4 channels across 20 tests

As mentioned above, the simulations were repeated for 20 different user populations, all statistically selected based on the user population profile "A" of Table 1. The detailed results for 1,2,3 and 4 channels are given in Appendices 2,3,4,5 respectively. The results are also shown in Figure 7. The result show that the larger the number of channels, the better the users get satisfied in general, which was to be expected. The Appendices also present, for each test simulations, the change of frame rate and resolution for 1, 2, 3 and 4 channels across 20 tests.

From the results in Appendices we see that the bandwidths of most "picked" channels lie on bandwidth limits of one of the user groups. For example, when we check the bandwidth (abbreviated as "BW") or "bwCh1", "bwCh2" columns, we can see that the bandwidths needed for optimal stream variants are very close to the group bandwidth limitations (bwLmt) shown in Table 1 in Section 5.2. This means that the best user satisfaction is obtained when we make full utilization of the provided bandwidth for each user class (which is to be expected).

For the one-channel case, from the table above, we see that in 90% out of the 20 test cases, the optimal streams can satisfy users in the first and second group, users in the third and fourth group are not satisfied at all for most cases.

For the two-channel case, the first channel concentrates on the first group, whose bandwidth limit is 2400, and the second channel jumps between the third and fourth group. In 7 out of 15, the second channel adapts to the preferences of the third user class.

For the three-channel case, the first channel still adapts to the first group, but with a lower frame rate and higher resolution combination in general. In comparison with the two-channel case, the three-channel case provides more options to the users, thus the first channel can better focus on serving the first group users and do not need to worry about how satisfied the users in other classes would be. In that way, it can pick up QoS parameters that better suit the first class of users. The general satisfaction level of the users of class 1 is higher than in the two-channel case. In the second channel in the three-channel cases, in 5 out of 20 cases adapts to the user class 2, in the other cases to the user class 3. As a result, the satisfaction level of class 2 is 16.7% higher than in two-channel case, and for class 3, it's 76.5% higher! For class 4, it's 11% higher.

For the four-channel case, since there are as many channels as user classes with their specific bandwidth limitations, each channel can adapt to a specific user class, therefore, the overall satisfaction levels of each group are improved greatly. The average satisfaction is between 75% and 85%. For user class 4, the class with unlimited bandwidth capacity, the satisfaction reaches 100%. We may observe from the results in the appendices that for the four-channel case, the first and second channel take lower frame rates but higher resolutions than those in the three-channel case.

5.5 Three Different Approaches to Characterize a User Group by a Representative Group Profile

The decision of characterizing receivers according to their bandwidth limits is based on the fact that the receivers that have the same bandwidth limit are more likely to have similar preferences, and the values of the preferences of their class representative are more likely to represent the preferences of individual receivers.

In Section 4.3, we introduced 3 different ways of summarizing user profiles into a reduced number of representative profiles. They are the mid-point, average value and boundary value methods. We also discussed their characteristics, pros and cons under different scenarios. In this section, using the 2-channel case as an example, we are going to show, in the case of grouping, how the user satisfaction level depends on the approaches taken for obtaining the representative group profiles.

The following graph presents the user satisfaction level for three different characterization approaches as well as for the case without grouping. It has been run for 20 tests on the same user population profiles as we did for non-grouping (see Figure 8). Again, for each test there are 100 sample users. The detailed results can be found in Appendix 6.

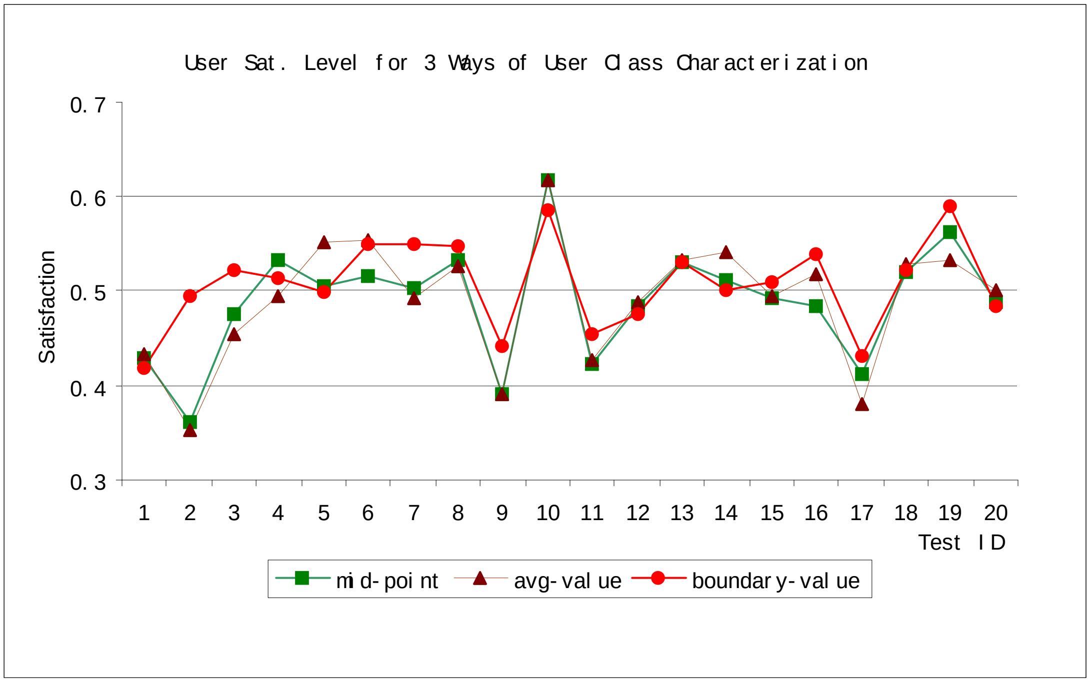


Figure 9 . User Satisfaction Level for 3 Ways of User Class Characterization

The results shown in Figure 9 show that the satisfaction level in the case of grouping is lower than in the case without grouping, due to the so-called "information loss" which is introduced when the sender processes the representative group profiles instead of the individual profile of all the users. Among the three characterizing approaches, the boundary-value approach gives the best satisfaction level, about only 0.05 lower than the non-grouping case. The other approaches are about 0.07 lower than without grouping. Also, if we look at the standard deviation, the boundary-value approach has the smallest standard deviation among the three user group characterization approaches. As a result, in following simulation, we are going to use only the boundary-value approach to determine the group profiles.

5.6. Simulation Results with Grouping

As discussed in the above sections, the receivers may be partitioned into several groups and each group is characterized by a given value of the bandwidth limits. In our simulation, each class sends one representative group profile to the sender. The data in the representative profile include the boundary values of the class and the number of users in the class. The sender uses the representative group profiles to do the optimization. The obtained QoS parameters of the streams are meant to satisfy the users at in the case of optimization without grouping. The goal can be verified by taking the QoS parameters back to the real users, and see how they get satisfied from the streams generated from the representative group profiles.

The detailed results for the 1,2,3 and 4 channel cases are presented in Appendices 7,8,9 and 10, respectively. The results are also shown in Figure 10.

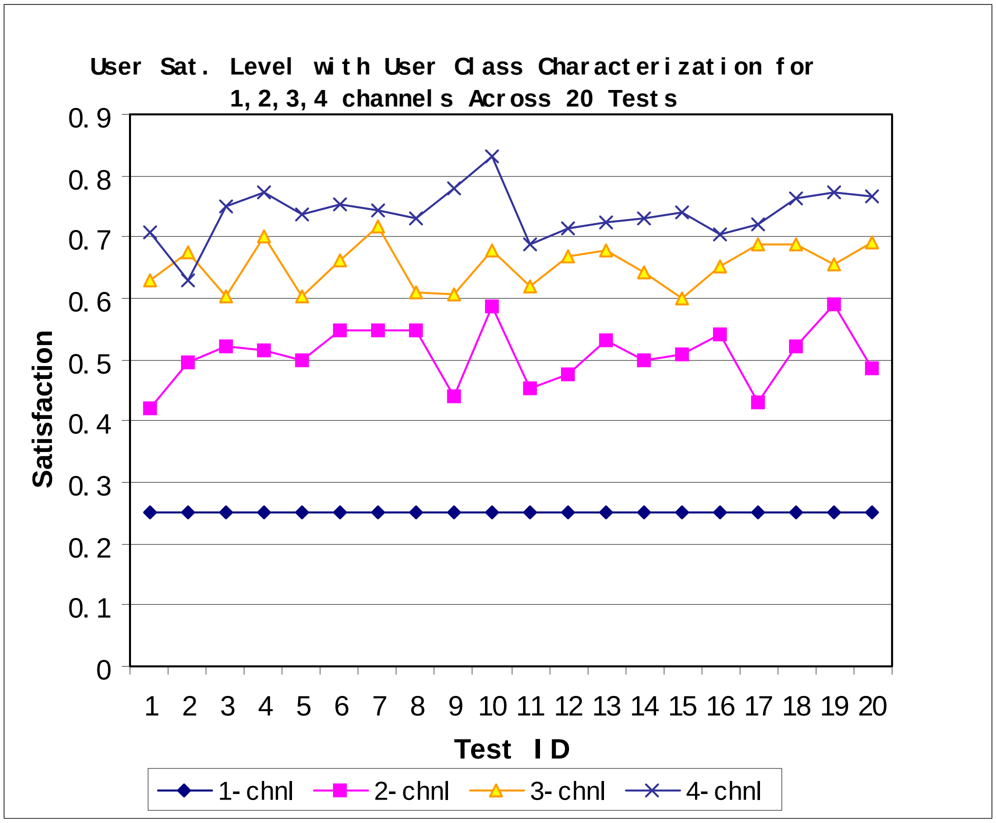


Figure 10 User Sat. Level with User Class Characterization for 1,2,3,4 Channels across 20 Tests

When we look at the graph above, we notice that the satisfaction level is a straight line at 0.25 for the 1 channel case. The user population is modeled in a way that if a high-quality stream can fully satisfy the 4th class users (100% satisfaction), it is not affordable to the other three classes due to their bandwidth limits. The satisfaction generated by the high quality stream is always (1+0+0+ 0)/4 = 0.25. On the other hand, a lower quality stream may satisfy more than 1 channel of users, but it may not satisfy some high-quality classes at all (like class 3 and class 4) because it does not meet their acceptable levels. Also a lower-quality stream can not give a very high satisfaction to its users due to its low quality. As a result, when there is only 1 channel allowed, the high quality stream is always picked up to make the average satisfaction maximum.

If we compare the average satisfaction of grouping with the non-grouping case, we see that the characterization approach gives slightly lower satisfaction level than without grouping. For the 1-channel case, by average, it is 0.25 vs. 0.2985, for the 2-channel case, it is 0.5080 vs. 0.5204, for the 3-channel case it is 0.6536 vs. 0.7355, and for the 4-channel case, it is 0.7651 vs. 0.8095. The difference is due to the information loss due to the processing of the representative group profiles instead of the real user profiles. One benefit of user class characterization is that by summarizing a large number of end-user profiles into a few copies of representative group profiles, the control traffic is effectively reduced to avoid network congestion.

From Appendices 2-10, we can compare the QoS parameters selected in grouping and non-grouping approaches to verify the accuracy of the characterization approach with respect to QoS parameter selection. For the 1-channel case, the streams generated without grouping meet in most cases the bandwidth limit of the first class, and serve the first group the best, also serve the second class, but give very poor user satisfaction level for third and fourth class. In contrary, the streams generated with grouping serve the fourth class only, since they are high quality streams and can not meet the bandwidth limits for the first, second and third class, leading to 0 satisfactions for those classes. Although the satisfaction levels are close between grouping and non-grouping, the selected QoS parameters are very different. As a result, for the 1-channel case, when there are a small number of users, the non-grouping approach is preferred.

For the 2, 3 and 4 channel cases, the selected QoS channels are similar between grouping and non-grouping, although grouping produces lower frames rate and higher resolutions than non-grouping. For the 2-channel case, the first channel in both approaches (grouping and non-grouping), meets the first class bandwidth. The second channel, while trying to meet the bandwidth limit of one class, jumps between satisfying the second and the third-class users (as shown in Appendix 3)for the non-grouping case, and for grouping it concentrates on serving the second-class users in case of grouping (see Appendix 8)

For the 3-channel case, in both grouping and non-grouping approaches, streams in the first channels are to meet the first class bandwidth limit, and the stream generated from the second channel jumps between satisfying second and third class users, while the third channel remains dedicated to the fourth class users.

For the 4-channel case, each channel is dedicated to one class of users in both grouping and non-grouping approaches. The simulation results show that for grouping, when increasing the number of channels, the selected QoS parameters become closer to the QoS parameters selected without grouping.

5.7.Validate Optimization Method by Varying User Sample Modeling

In order to validate our optimization method, we run through the simulation for 1,2,3,4 channels with populations generated according to a different user population profile (user population profile type B shown in Figure 10). Even though the profile type B is populated within the same range as the original profile (user population profile A), the distance between the minimum value and the ideal value have been redefined to differentiate from the original ones, so that we can compare the results between the two samples. In profile A, there is no overlap between the minimum and ideal values, but for profile type B, the minimum and ideal value range may overlap. For example, in profile type A, the frame rate profile for class 1 users are: minimum: 1- 3, ideal 4 -9. In profile type B, these ranges are, minimum 1-5, ideal 2-9. The users are more diversified within one class in profile type B.

Appendix 11 shows the user population profile of type “B”. The original profile – population profile type “A” -- is shown in Table 1 in Section 5.3. Figure 11 also presents the user population profile of type “B”.

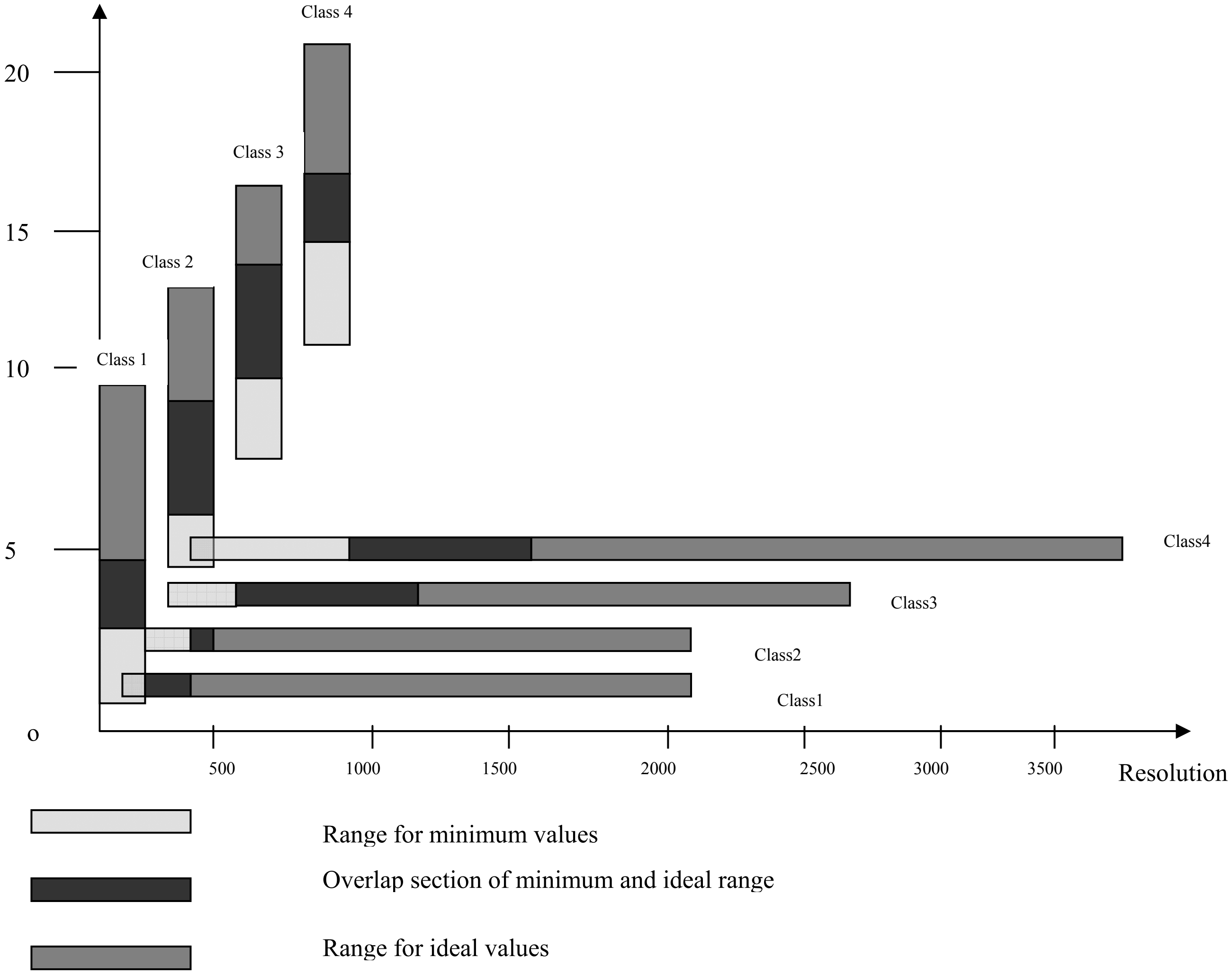


Figure 11 Graphical presentation of simulated user population profile B

Detailed simulation results with population profile B, including non-grouping and grouping, are shown in Appendices 12,13,14,15,16,17,18,19 Appendices 12 and 13 are for the 1-channel case, non-grouping and grouping, respectively. Appendices 14 and 15 are for the 2-channel case, Appendices 16 and 17 are for the 3-channel case, and Appendices 18 and 19 are for the 4-channel case, non-grouping and grouping, respectively.

For the 1-channel case, the results are very similar between for populations of type A and type B. Class 3 is better satisfied for type B populations, since users in this class have preferences closer to their peers in the neighboring classes. The compositions of the QoS parameters between the two different cases are very close, too. As for the average satisfaction level, it is 0.2985 vs. 0.3106 without grouping; and 0.25 vs 0.25 with grouping.

For the 2, 3 and 4 channel cases, the results are similar between the population type A and B. Especially for the non-grouping, not only the satisfaction are close to one another in both case, but also the selected QoS parameters are not far from one another. Also, since the distance between minimum and ideal are more flexible in population profile type B, class 3 is better satisfied with a cost of slightly reduced in satisfaction for its neighboring classes. In grouping, the selected QoS parameters show a bigger difference between the population types “B” and “A” than without grouping, but their overall user satisfaction levels are close to one another. We can draw the conclusion that the RSD approach works similarly with both types of user populations.

CHAPTER VI CONCLUSION

In a multicast network of a large number of receivers where the transmission resources, the end-systems and the receiver preferences are of high heterogeneity, multicast streams with the same content but different QoS parameters should be sent out to the target receivers in such a manner that each receiver finds a stream with appropriate quality. In this thesis, we make use of the concept of *user profiles* and a *user satisfaction function* to automatically select the stream variants for a multicast application that can maximize the average user satisfaction level for all receivers.

6.1 Contributions

By introducing the Receiver-Satisfaction Driven (RSD) method, this thesis makes the following contributions:

1. Given the user profiles and the number of streams to be transmitted, we introduce the RSD method that can automatically select the QoS parameters of the stream variants that maximize the average user satisfaction. We have done simulation studies to test the user satisfaction for different simulated user populations.
2. When there are a large number of receivers, we partition the receivers into different user classes and define a representative group profile for each group. We have shown that the automated selection of stream variants can be based on the representative group profiles.
3. We have compared three different user class characterization methods: the average value, the mid-point value and the boundary-value methods. Our simulation results show that the average user satisfaction obtained with the boundary-value method is the best among the three methods.

6.2 Future work:

This thesis leaves the automated selection of the number of channels for future study. For this purpose, one should take the bandwidth limitation at the server side into consideration and optimize the number of channels to be multicast by the source and consider at the same time the trade-off with the obtained user satisfaction.

Appendices

Results for User Population Profile Type A

No-Grouping (Related to Section 5.4)

Grouping (Related to Section 5.5 and 5.6)

Results For User Population Profile Type “B” (Relate to Section 5.7)

User Population Profile Type “A”

No-Grouping

Appendix 1: Maximum satisfactions and QoS parameters for 15 tests with different initial values for the 4-channel case

Test_ID	Sat	Fr_Ch1	Res_Ch1	BW_Ch1	Fr_Ch2	Res_Ch2	BW_Ch2	Fr_Ch3	Res_Ch3	BW_Ch3	Fr_Ch4	Res_Ch4	BW_Ch4
1	0.6587	5.354	1027	2399.75	11.2852	853	4200.91	3.4966	5243	8000.38	23.12	3568	36001.29
2	0.6152	1.754	3136	2400.03	9	1069	4198.6	14.7505	1243	8001.34	21.46	3844	36003.64
3	0.7803	5.355	1027	2399.79	9	1069	4198.6	14.7506	1243	8001.39	23.76	3472	36000.39
4	0.7803	5.354	1027	2399.75	9	1069	4198.6	14.7505	1243	8001.34	22.33	3695	36007.73
5	0.7803	5.354	1027	2399.71	9	1069	4198.6	14.7506	1243	8001.39	20.05	4114	36003.58
6	0.7803	5.355	1027	2399.79	9	1069	4198.6	14.7505	1243	8001.34	21.84	3778	36007.37
7	0.6152	0.813	6762	2400.29	9	1069	4198.6	14.7505	1243	8001.34	23.69	3482	36005.31
8	0.6152	1.504	3658	2400.43	9	1069	4198.6	14.7505	1243	8001.34	22.62	3647	36007.40
9	0.7803	5.355	1027	2399.79	9	1069	4198.6	14.7505	1243	8001.34	21.27	3879	36003.39
10	0.7803	5.354	1027	2399.75	9	1069	4198.6	14.7505	1243	8001.34	22.68	3637	36005.17
11	0.6587	5.354	1027	2399.75	11.2852	853	4200.91	5.6205	3262	8000.99	23.09	3572	35998.00
12	0.6033	5.354	1027	2399.75	9	1069	4198.6	15.9069	1153	8003.86	11.37	7259	36004.58
13	0.6587	5.354	1027	2399.75	11.2852	853	4200.91	8.9998	2037	8000.34	20.86	3955	36000.63
14	0.7803	5.354	1027	2399.75	9	1069	4198.6	14.7505	1243	8001.34	20.01	4124	36005.43
15	0.7803	5.354	1027	2399.75	9	1069	4198.6	14.7506	1243	8001.39	21.90	3768	36005.95
Avg	0.7112	4.555	1725.3	2399.86	9.5	1025.8	4199.1	13.1	1691.2	8001.4	21.34	3986.3	36003.99
Stdev	0.0783	1.665	1624.5	0.22	0.9	89.4	1.0	3.8	1124.5	0.8	3.00	928.0	2.86
Var	0.0061	2.773	2638944.7	0.05	0.9	7998.2	0.9	14.5	1264403.5	0.6	8.99	861218.1	8.17

Appendix 2: Optimal Stream Parameters for Single Stream and No Grouping

TestID	Frame Rate	Resolution	BW	Sat_Grp1	Sat_Grp2	Sat_Grp3	Sat_Grp4	Sat_All
1	8.28	664	2399.96	0.6677	0.5179	0	0	0.2964
2	8.00	687.5	2400.00	0.6932	0.5834	0	0	0.3191
3	9.36	587.6	2400.23	0.7726	0.5465	0.0303	0	0.3374
4	21.27	3869.7	35911.38	0	0	0	1	0.25
5	21.95	3607.7	34547.34	0	0	0	1	0.25
6	8.28	664.7	2400.17	0.739	0.4466	0.0241	0	0.3024
7	8.59	640.4	2400.17	0.731	0.4774	0	0	0.3021
8	7.15	769.2	2399.90	0.8893	0.3595	0	0	0.3122
9	9.15	601.4	2400.18	0.552	0.5033	0.0391	0	0.2736
10	9.00	611.1	2399.96	0.6792	0.5714	0	0	0.3126
11	8.00	687.5	2400.00	0.5773	0.656	0	0	0.3083
12	8.41	654.1	2399.86	0.817	0.4538	0.0154	0	0.3215
13	7.85	701.1	2400.06	0.7539	0.3316	0	0	0.2714
14	8.23	668.2	2399.98	0.6142	0.5004	0	0	0.2786
15	7.90	696	2399.91	0.6685	0.4141	0	0	0.2706
16	8.00	687.5	2400.00	0.7751	0.5231	0	0	0.3246
17	9.30	591.7	2399.94	0.8739	0.4155	0.0238	0	0.3283
18	7.14	770.4	2399.95	0.8077	0.5135	0	0	0.3303
19	8.66	635	2399.88	0.751	0.395	0.0214	0	0.2919
20	8.00	687.5	2400.00	0.6297	0.5244	0	0	0.2885
Avg	9.6249	974.1150	5682.94	0.6496	0.4367	0.0077	0.1000	0.2985
StdDev	4.143	947.705	10107.08	0.240	0.168	0.013	0.308	0.026
Var	17.2	898144.7	102153126.09	0.1	0.0	0.0	0.1	0.0

Appendix 3: Optimal Stream Parameters for Two Streams and No Grouping

Index	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	satGp1	satGp2	satGp3	satGp4	satAll
1	8.2832	663.9986	2400.214	24.6973	3301	35577.85	0.6677	0.5179	0	1	0.5464
2	7.5852	725.0947	2400.195	20.021	4048	35368.04	0.6254	0.354	0	1	0.4948
3	7.7903	706.01	2400.213	23.3827	3524	35959.64	0.6732	0.4482	0	1	0.5304
4	8.7361	629.01	2398.059	16.1537	1135	8001.154	0.6438	0.5504	0.6374	0.4144	0.5615
5	7.5024	733.0956	2400.19	19.1558	2898	24226.1	0.7402	0.5297	0	1	0.5675
6	9	611.1119	2400.203	16.4731	1113	8001.202	0.778	0.6216	0.6313	0.3372	0.592
7	8.1722	673.01	2400.188	21.7529	3379	32076.73	0.594	0.64	0	1	0.5585
8	7.5851	725.1054	2400.199	15.0452	1219	8003.619	0.7296	0.4752	0.7415	0.2761	0.5556
9	9	611.1111	2400.2	20.6902	3880	35033.31	0.6337	0.5992	0	1	0.5582
10	8.4253	652.799	2400.21	23.9616	3443	36002.9	0.7813	0.8382	0.0252	0.9864	0.6578
11	8.2211	669.01	2400.199	21.8313	3289	31334.89	0.5969	0.5629	0.0002	1	0.54
12	8.338	659.633	2400.209	16.2568	1128	8002.559	0.6373	0.4074	0.7129	0.338	0.5239
13	8.7485	628.6826	2400.213	15.9999	1146	8001.78	0.6365	0.5296	0.7297	0.3466	0.5606
14	8.2563	666.1602	2400.208	14.9742	1221	7978.919	0.6418	0.4839	0.8222	0.3526	0.5752
15	8.2367	667.7457	2400.209	25.1919	3010	33091.17	0.7928	0.4778	0.0117	1	0.5706
16	8.3253	660.6382	2400.205	20.0263	3854	33681.95	0.6537	0.5044	0.0279	1	0.5465
17	8	687.5	2400.2	21.0158	3926	36006.5	0.7589	0.4217	0	1	0.5452
18	7.7163	712.7799	2400.21	21.1659	3501	32338.03	0.852	0.4857	0	1	0.5844
19	9.1209	603.01	2400.197	21.4051	3606	33684.32	0.939	0.5309	0	1	0.6175
20	7.7133	713.0577	2400.212	15.981	1147	7999.302	0.6162	0.4406	0.7051	0.3196	0.5204
Avg	8.23781	669.9282	2400.097	19.75909	2688.4	24518.5	0.6996	0.520965	0.252255	0.768545	0.56035
Stdev	0.497748	40.20186	0.479658	3.326014	1185.128	12694.89	0.093119	0.103408	0.34773	0.323001	0.035467
Var	0.247753	1616.19	0.230072	11.06237	1404528	1.61E+08	0.008671	0.010693	0.120916	0.10433	0.001258

Appendix 4: Optimal Stream Parameters for Three Streams and No Grouping

Index	frCh1	resCh1	bwCh1	frCh2	reCh2	bwCh2	frCh3	resCh3	bwCh3	satGp1	satGp2	satGp3	satGp4	satAll
1	8.056	683	2401.1	13.814	1326	7993.8	21	3852	35301	0.6777	0.5065	0.5701	0.8579	0.6886
2	5.192	1059	2399.4	11.471	839	4200	21.74	3204	30404	0.7771	0.7375	0.2594	0.8333	0.6935
3	5.054	1088	2399.7	10.637	905	4201	19.03	3770	31305	0.7138	0.6557	0.1262	0.8333	0.6239
4	8.534	644	2398.5	14.565	1259	8002.2	21	2597	23800	0.8387	0.6168	0.7004	0.8639	0.7767
5	8.656	635	2398.6	15.588	1176	8000.1	21.48	3841	36001	0.6133	0.4792	0.4341	0.9127	0.6316
6	8	688	2401.9	15.509	1182	7999.7	23.54	2646	27187	0.7843	0.5422	0.6458	0.9006	0.7359
7	6.262	849	2320.2	12	802	4199.9	21.36	3862	36006	0.8673	0.8275	0.2108	0.8333	0.7264
8	8.134	676	2399.4	13.789	1329	7997.5	20.91	3530	32219	0.7352	0.4294	0.4485	0.8726	0.6533
9	7.49	734	2399.1	14.586	1257	8001.3	22.91	3096	30955	0.6259	0.4731	0.4998	0.8606	0.6497
10	5.097	1079	2400.2	10.375	928	4201.7	17	4837	35886	0.8324	0.8441	0.0863	0.8148	0.6851
11	7.72	712	2398.7	14.631	1247	7961.8	23.04	3276	32946	0.5833	0.4537	0.5315	0.8569	0.642
12	8.443	651	2398.6	14.199	1291	7999.6	24.2	3204	33843	0.5841	0.5424	0.616	0.8749	0.6856
13	8.892	619	2401.9	12.777	1435	8001.1	18.7	3618	29531	0.6691	0.5383	0.527	0.8186	0.6792
14	9.319	590	2399.3	13.936	1316	8003.6	20	2723	23766	0.7108	0.5567	0.4708	0.816	0.6701
15	5.499	1000	2399.9	11.681	824	4200.3	22.26	2881	27987	0.6533	0.7949	0.2001	0.8123	0.6558
16	9.037	608	2397.7	14.899	1230	7997.5	20	4125	36003	0.7356	0.4993	0.6017	0.8588	0.7091
17	9.11	604	2402	13.78	1330	7999	21	3892	35668	**0.715**	**0.59**	**0.5487**	**0.8333**	**0.7135**
18	9.259	590	2383.8	13.55	1353	8000.6	20.98	3828	35055	0.7592	0.6895	0.6008	0.901	0.7622
19	7.433	740	2400.2	11	875	4200.4	20	3966	34615	0.7607	0.7761	0.1217	0.8333	0.6646
20	8.634	637	2400.1	15.79	1161	8000.2	19.04	3674	30530	0.6248	0.6104	0.7067	0.9083	0.7355
Avg	7.691	744.3	2395	13.429	1153.3	6858.1	20.96	3521.1	31950	0.7131	0.60817	0.44532	0.85482	0.689115
Stdev	1.465	172	17.999	1.6928	207.12	1785	1.763	564.62	3938.2	0.084	0.13017	0.20269	0.03232	0.042657
Var	2.146	29584	323.96	2.8655	42899	3E+06	3.108	318798	2E+07	0.0071	0.01694	0.04108	0.00104	0.00182

Appendix 5: Optimal Stream Parameters for Four Streams and No Grouping

	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	frCh3	resCh3	bwCh3	frCh4	resCh4	bwCh4	satGp1	satGp2	satGp3	satGp4	satGpAll
1	6.838	804	2399.1	10	963	4202.5	13	1410	7999.2	23.637	3490	36001	0.6885	0.7519	0.6545	1	0.7737
2	6	917	2401.1	10	962	4198.2	12.74	1440	8003.1	20.287	4067	36005	0.7181	0.6502	0.5368	1	0.7263
3	5.447	1010	2400.7	10.128	950	4198.7	13.76	1332	7999.4	23.389	3527	36000	0.7133	0.6828	0.6529	1	0.7623
4	7	786	2401.1	9.9046	972	4201.3	14.74	1244	8001.4	24.295	3396	36006	0.6701	0.7948	0.6784	1	0.7858
5	6	917	2401.1	9.5606	1007	4201.5	13.42	1366	7999.5	23.099	3572	36008	0.8149	0.8001	0.6823	1	0.8243
6	6.872	800	2399.3	11	875	4200.4	14	1310	8003.6	20.352	4054	36006	0.8544	0.8628	0.68	1	0.8493
7	5.097	1079	2400.2	11	875	4200.4	12.48	1469	7999.4	21.889	3769	36003	0.684	0.8489	0.6952	1	0.807
8	5	1100	2400.2	9	1069	4198.6	14.1	1300	8000.6	22.578	3654	36003	0.8545	0.7857	0.7589	1	0.8498
9	5.621	978	2399	11	875	4200.4	13	1410	7999.2	20.946	3939	36006	0.7082	0.8296	0.7399	1	0.8194
10	7	786	2401.1	11	875	4200.4	13.31	1377	7999.7	20.766	3973	36005	0.7936	0.9762	0.6624	1	0.858
11	5.924	928	2399.3	11	875	4200.4	13.01	1409	8001.7	24.004	3437	36004	0.7	0.8073	0.5653	1	0.7681
12	5	1100	2400.2	10	962	4198.2	14	1310	8003.6	21.232	3886	36007	0.8263	0.7383	0.8183	1	0.8457
13	6	917	2401.1	10.526	914	4198.6	14.85	1235	8002.9	22.304	3699	36003	0.7396	0.7966	0.7508	1	0.8217
14	5.051	1089	2400.3	9.2771	1037	4198.3	14.55	1260	8000.6	20.77	3972	36002	0.7341	0.7127	0.8291	1	0.819
15	6.509	845	2400.2	11.519	836	4202.3	16	1146	8001.8	30.389	2715	36006	0.8343	0.7353	0.6993	1	0.8172
16	5.481	1003	2399.2	9.73	989	4199.5	14.53	1262	8002.2	20.601	4005	36006	0.7515	0.7631	0.6704	1	0.7963
17	7	786	2401.1	9.7565	987	4202.4	13	1410	7999.2	21.454	3845	35999	0.7688	0.6841	0.7122	1	0.7913
18	6.785	811	2401.3	11	875	4200.4	14.07	1303	7997.9	20.236	4077	36004	0.8794	0.7615	0.6512	1	0.823
19	8.346	659	2400.2	12.731	756	4200.3	14.43	1270	7999	23.49	3512	36001	0.9525	0.8342	0.7067	1	0.8733
20	5.354	1027	2399.7	9	1069	4198.6	14.75	1243	8001.3	22.33	3695	36008	0.6604	0.7277	0.733	1	0.7803
Avg	6.116	917.1	2400.3	10.357	936.15	4200.1	13.89	1325.3	8000.8	22.403	3714	36004	0.767325	0.7772	0.6939	1	0.80959
Stdev	0.901	128.91	0.7763	0.9283	81.05	1.4423	0.892	83.769	1.7197	2.3029	325.8	2.542	0.0803	0.0743	0.0703	0	0.036857
Var	0.812	16617	0.6027	0.8617	6569.1	2.0803	0.796	7017.3	2.9573	5.3035	1E+05	6.462	0.0064	0.0055	0.0049	0	0.001358

Grouping

Appendix 6: User Satisfaction Level for 3 Methods of User Class Characterization

testID	non-grping	mid-point	avg-value	boundary-value
1	0.5464	0.4298	0.4338	0.4193
2	0.4948	0.3614	0.3539	0.4947
3	0.5304	0.4749	0.4539	0.5229
4	0.5615	0.5324	0.4944	0.5142
5	0.5675	0.5058	0.551	0.4985
6	0.592	0.5158	0.5532	0.5487
7	0.5585	0.5023	0.4931	0.549
8	0.5556	0.5321	0.5271	0.5469
9	0.5582	0.3908	0.392	0.4415
10	0.6578	0.6174	0.6185	0.5865
11	0.54	0.4237	0.4267	0.4538
12	0.5239	0.485	0.4886	0.4765
13	0.5606	0.5314	0.533	0.5305
14	0.5752	0.5116	0.5403	0.5005
15	0.5706	0.492	0.4939	0.5093
16	0.5465	0.4833	0.5182	0.5397
17	0.5452	0.4112	0.3804	0.4316
18	0.5844	0.5201	0.5277	0.522
19	0.6175	0.5633	0.5322	0.5902
20	0.5204	0.4891	0.502	0.4846
Mean	0.5604	0.4887	0.4907	0.5080
StdDev	0.0355	0.0608	0.0660	0.0474

Appendix 7: Optimal Stream Parameters for Single Stream and Grouping

TestID	Frame Rate	Resolution	BW	Sat_Grp1	Sat_Grp2	Sat_Grp3	Sat_Grp4	Sat_All
1	23.5648	3126.5	32149.24	0	0	0	1	0.25
2	20.9047	3672.2	33498.00	0	0	0	1	0.25
3	21.0625	3849	35375.81	0	0	0	1	0.25
4	21.07	3896.8	35827.89	0	0	0	1	0.25
5	20.8789	3370.4	30707.02	0	0	0	1	0.25
6	20.6329	3920.5	35298.02	0	0	0	1	0.25
7	19.3259	3421.6	28854.76	0	0	0	1	0.25
8	21.1042	3883.4	35762.64	0	0	0	1	0.25
9	22.3418	3642	35506.40	0	0	0	1	0.25
10	20.7147	3169.2	28646.85	0	0	0	1	0.25
11	19.604	4099.4	35068.21	0	0	0	1	0.25
12	21.4028	3276.9	30604.29	0	0	0	1	0.25
13	21.9787	3489.5	33466.77	0	0	0	1	0.25
14	21.7935	3429.3	32612.27	0	0	0	1	0.25
15	19.1971	3264.1	27343.09	0	0	0	1	0.25
16	21.4548	3696.6	34607.92	0	0	0	1	0.25
17	21.0019	3852.9	35309.77	0	0	0	1	0.25
18	20.7993	3929.9	35668.00	0	0	0	1	0.25
19	22.2935	3546.1	34496.72	0	0	0	1	0.25
20	22.4659	2997.1	29381.48	0	0	0	1	0.25
Avg	21.18	3576.67	33009.26	0.00	0.00	0.00	1.00	0.25
StdDev	1.07	314.56	2790.54	0.00	0.00	0.00	0.00	0.00
Var	1.148	98949.518	7787133.243	0.000	0.000	0.000	0.000	0.000

Appendix 8: Optimal Stream Parameters for Two Streams and Grouping

	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	satGp1	satGp2	satGp3	satGp4	satGpAll
1	8.4733	649.092	2400.2	19.173	956	7998.94	0.6595	0.5253	0.3144	0.1779	0.4193
2	7.4779	735.5014	2400.2	20.586	3893	34973.3	0.6325	0.3464	0	1	0.4947
3	7.581	725.502	2400.2	28.15	2918	35846.8	0.679	0.4411	0	0.9716	0.5229
4	8.0777	680.8842	2400.2	18.472	992	7996.73	0.6538	0.527	0.4718	0.4042	0.5142
5	8.5953	632.1197	2371.1	17.948	1021	7996.9	0.6798	0.5164	0.4455	0.3521	0.4985
6	8.4416	651.5375	2400.2	35.385	2331	35995.3	0.7996	0.5825	0	0.8125	0.5487
7	8.4954	647.4096	2400.2	22.981	2693	27008.2	0.577	0.6512	0	0.9678	0.549
8	7.1996	763.9284	2400.2	21.784	3215	30563.7	0.7529	0.4445	0	0.99	0.5469
9	7.465	736.774	2400.2	17.561	1043	7993.25	0.6799	0.4951	0.2948	0.2964	0.4415
10	8.5453	643.6261	2400.2	17.721	1035	8003.98	0.7797	0.8388	0.4771	0.2506	0.5865
11	8.2251	668.6196	2400	17.824	1029	8003.83	0.5966	0.5631	0.3382	0.3171	0.4538
12	8.0655	681.9139	2400.2	18.256	979	7799.7	0.6554	0.3869	0.5668	0.2967	0.4765
13	8.0448	683.6692	2400.2	28.502	2890	35946	0.6609	0.4802	0	0.981	0.5305
14	7.4128	741.9641	2400.2	18.502	991	8001.73	0.6801	0.4523	0.5713	0.2982	0.5005
15	7.4352	739.7285	2400.2	17.73	1034	8000.49	0.8181	0.4416	0.6318	0.1457	0.5093
16	7.504	732.9445	2400.2	20.025	3740	32683.5	0.6989	0.4613	0	0.9985	0.5397
17	8.8043	624.6746	2400.1	17.662	1038	8000.68	0.7287	0.4159	0.3773	0.2046	0.4316
18	7.2245	761.2968	2400.2	17.314	996	7525.48	0.8706	0.4616	0.5184	0.2374	0.522
19	8.2839	663.9412	2400.2	16.782	1092	7997.3	0.9524	0.4825	0.5805	0.3455	0.5902
20	8.1662	673.5081	2400.2	18.299	1002	8001.7	0.6026	0.446	0.6035	0.2862	0.4846
Avg	7.9759	691.9318	2398.7	20.533	1744.4	16816.9	0.7079	0.49799	0.30957	0.5167	0.50805
Stdev	0.5134	45.50395	6.5094	4.8211	1066.4	12550.8	0.09575	0.1057	0.25039	0.34094	0.04739
Var	0.2636	2070.609	42.372	23.243	1E+06	1.6E+08	0.00917	0.01117	0.0627	0.11624	0.00225

Appendix 9: Optimal Stream Parameters for Three Streams and Grouping

TestId	frCh1	resCh1	bwCh1	frCh2	reCh2	bwCh2	frCh3	resCh3	bwCh3	satGp1	satGp2	satGp3	satGp4	satGpAll
1	6.76	814	2401.2	12.259	785	4199.4	21.01	3927	36003	0.6884	0.7044	0.1305	0.8333	0.6308
2	5.907	931	2399.9	10.736	897	4202.4	23.78	3103	32203	0.7313	0.7655	0.2081	0.8295	0.6751
3	5.465	1005	2397	11.581	831	4199.9	24.87	3076	33389	0.7	0.6144	0.1547	0.784	0.6025
4	6.927	793	2397.2	11	875	4200.4	21.29	3534	32831	0.8582	0.8904	0.0682	0.8266	0.7022
5	6.388	861	2400.4	11.411	844	4202.7	20.4	4030	35870	0.6697	0.7027	0.0453	0.8333	0.6044
6	6.099	902	2400.6	10.14	949	4199.4	22.85	2775	27677	0.8193	0.7754	0.0809	0.8141	0.6631
7	6.169	892	2401.4	11.204	859	4200.2	20.88	3079	28054	0.876	0.8109	0.1989	0.815	0.716
8	8	687	2398.5	16.98	1078	7988.2	23.55	3213	33022	0.7422	0.4225	0.2814	0.8779	0.6087
9	6.353	865	2398	11.114	866	4200.4	24.3	2875	30483	0.6569	0.7025	0.0732	0.826	0.606
10	5.493	980	2349.2	11.734	820	4199	18	3533	27752	0.8051	0.8471	0.0727	0.8256	0.6789
11	5.844	941	2399.7	12.43	774	4198.5	25.85	3191	36003	0.6814	0.6984	0.1047	0.8302	0.6202
12	5.779	952	2400.9	10.979	877	4201.9	21.65	3318	31347	0.6537	0.8352	0.1815	0.8333	0.6676
13	6.647	827	2399	11.991	803	4202.1	24.43	3317	35366	0.72	0.8317	0.1785	0.8235	0.6796
14	6.179	890	2399.7	10.983	876	4198.8	23.4	2915	29764	0.7903	0.7556	0.046	0.8118	0.6415
15	7.509	691	2264.2	17.001	885	6566.1	21	2749	25193	0.521	0.5037	0.406	0.8541	0.5997
16	5.276	1042	2399.1	10.101	953	4201	20.31	3038	26923	0.8428	0.7568	0.0738	0.7742	0.6506
17	6.2	887	2400	12	802	4201	25.5	3222	35921	**0.806**	**0.8034**	**0.187**	**0.8011**	**0.6895**
18	6.173	887	2389.4	12.993	741	4201.7	21	3005	27539	0.8153	0.8883	0.1071	0.8083	0.6879
19	5.847	941	2401	11	875	4200.4	20.38	3705	32944	0.726	0.7762	0.1218	0.8333	0.656
20	5.241	1049	2399.2	13	740	4198.2	22.8	3421	34044	0.7567	0.7968	0.2334	0.819	0.6924
Avg	6.213	891.85	2389.7	12.032	856.5	4508.1	22.36	3251.3	31616	0.743	0.7441	0.14769	0.82271	0.653635
Stdev	0.703	97.863	31.677	1.8748	78.729	974.68	2.078	354.97	3491.3	0.0864	0.11864	0.09021	0.02225	0.03721
Var	0.494	9577.1	1003.4	3.5149	6198.3	950009	4.319	126001	1E+07	0.0075	0.01408	0.00814	0.0005	0.001385

Appendix 10: Optimal Stream Parameters for Three Streams and Grouping:

Index	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	frCh3	resCh3	bwCh3	frCh4	resCh4	bwCh4	satGp1	satGp2	satGp3	satGp4	satGpAll
1	3.817	1441	2400.4	6.6938	1438	4200.6	8.469	2165	8001.9	20.686	3988	36001	satGp1	satGp2	satGp3	satGp4	satGpAll
2	2.674	2057	2400.6	8.0958	1189	4200.7	8.73	2100	8000.2	20.643	3996	35999	0.6587	0.5259	0.6418	1	0.7066
3	8.438	652	2400.8	8.9023	1081	4199.6	12.51	1466	8000.5	21.753	3793	36007	0.6325	0.3464	0.5362	1	0.6288
4	1.933	2846	2400.2	6.4541	1491	4199.5	4.761	3851	8000.9	23.423	3522	36001	0.679	0.6671	0.6514	1	0.7494
5	8.908	617	2398.6	10.224	941	4198.5	10.17	1802	8001.5	19.288	4277	36001	0.6547	0.7937	0.6432	1	0.7729
6	4.504	1221	2399.9	8.0383	1197	4199	6.226	2945	8001	22.01	3748	35999	0.7407	0.5292	0.6821	1	0.738
7	1.921	2864	2400.3	6.6095	1456	4199.7	6.587	2783	7999.3	22.442	3676	36002	0.8072	0.5614	0.6497	1	0.7546
8	7.191	765	2400.7	10.221	942	4201.8	12.3	1490	8000.1	20.556	4013	36000	0.577	0.7094	0.6846	1	0.7427
9	5.01	1098	2400.7	7.7576	1241	4201.3	15.2	1207	8003.8	20.016	4122	36005	0.7529	0.4445	0.7261	1	0.7309
10	2.359	2332	2400.7	10.955	879	4202.4	5.501	3333	8001.3	20.162	4092	36004	0.6801	0.6956	0.7397	1	0.7789
11	2.107	2610	2399.9	6.7334	1429	4199.1	13.07	1403	7999.4	21.167	3898	36007	0.7826	0.885	0.6614	1	0.8323
12	6.459	852	2401.4	6.2675	1536	4201.2	11.54	1589	8000.8	21.381	3859	36008	0.5939	0.5936	0.5618	1	0.6873
13	1.986	2769	2399.9	9.2842	1037	4201.5	9.316	1968	8001	21.575	3824	36004	0.6572	0.3843	0.818	1	0.7149
14	0.817	6733	2400.3	10.798	891	4198.4	8.12	2258	8001.7	20.543	4016	36004	0.6609	0.4802	0.7499	1	0.7227
15	6.58	836	2400.4	5.8777	1638	4201.5	9.213	1990	8001	21.768	3790	36002	0.6802	0.4521	0.7861	1	0.7296
16	4.618	1191	2400.4	7.5737	1271	4200.9	13.4	1368	8001	20.696	3986	36000	0.8204	0.4681	0.6691	1	0.7394
17	3.316	1658	2399.5	12.234	787	4201.8	5.538	3311	8001.8	21.453	3846	36007	0.6989	0.4613	0.6524	1	0.7031
18	1.888	2913	2400.1	5.3927	1785	4200.8	10.48	1750	8002.6	20.243	4075	35999	0.7634	0.4082	0.7074	1	0.7197
19	1.194	4606	2400.4	11.748	819	4198.8	12.01	1527	8001.7	20.554	4014	36005	0.8708	0.5336	0.648	1	0.7631
20	3.824	1438	2399.9	7.0217	1371	4201.1	6.073	3019	8000.5	20.497	4025	36004	0.9501	0.4988	0.6382	1	0.7718
Avg	3.977	2075	2400.2	8.3441	1221	4200.4	9.46	2166.3	8001.1	21.043	3928	36003	0.6025	0.7265	0.7315	1	0.7651
Stdev	2.428	1507.9	0.5703	2.0731	292.2	1.2233	3.06	776.05	1.0347	0.95	173.6	2.885	0.713185	0.5582	0.6789	1	0.73759
Var	5.895	2E+06	0.3252	4.2976	85379	1.4964	9.362	602248	1.0707	0.9025	30132	8.324	0.09687985	0.1445	0.0679	0	0.041447

Results For User Population Profile Type “B”

Appendix 11: Simulated User Population Profile Type “B”

	frMin	frMax	resMin	resMax
gp1	1--5	2--10	27--267	127--2187
gp2	4--8	5--13	108--450	308--2187
gp3	8--12	9--17	261--1200	561--2700
gp4	12--16	13--21	432--1500	1044--3888

Appendix 12: Optimal Stream Parameters for Single Stream and No Grouping:

TestID	Frame Rate	Resolution	BW	Sat_Grp1	Sat_Grp2	Sat_Grp3	Sat_Grp4	Sat_All
1	9.29	592	2399.63	0.6407	0.5991	0.0521	0	0.323
2	7.96	691	2399.70	0.6998	0.63	0	0	0.3324
3	9.34	589	2401.53	0.7947	0.4948	0.0346	0	0.331
4	8.12	677	2400.24	0.816	0.633	0	0	0.3623
5	6.67	824	2399.40	0.6948	0.3492	0	0	0.261
6	8.61	639	2400.22	0.7256	0.4564	0.0705	0	0.3131
7	21.18	3258	30116.86	0	0	0	1	0.25
8	8.12	678	2401.04	0.8637	0.4117	0.0286	0	0.326
9	9.37	587	2399.11	0.5548	0.5323	0.0257	0	0.2782
10	9.00	611	2399.56	0.6815	0.4922	0	0	0.2934
11	7.61	723	2401.42	0.6221	0.5738	0	0	0.299
12	9.20	598	2400.83	0.7787	0.4553	0.0308	0	0.3162
13	8.05	683	2400.12	0.7453	0.3591	0	0	0.2761
14	8.30	662	2398.65	0.6309	0.5175	0	0	0.2871
15	7.31	752	2398.55	0.7419	0.3618	0	0	0.2759
16	16.00	1146	8001.16	0	0	0.7611	0.5805	0.3354
17	8.00	687	2398.25	0.9011	0.4809	0	0	0.3455
18	7.00	786	2400.87	0.7798	0.6444	0	0	0.3561
19	8.36	658	2400.04	0.7854	0.4706	0.0266	0	0.3206
20	15.47	1185	8000.02	0	0	0.6429	0.6718	0.3287
Avg	9.6486	851.3000	4345.8607	0.6228	0.4231	0.0836	0.1126	0.3106
StdDev	3.635	589.792	6304.676	0.281	0.202	0.213	0.284	0.032
Var	13.2	347854.5	39748944.1	0.1	0.0	0.0	0.1	0.0

Appendix 13: Optimal Stream Parameters for Single Stream and Grouping:

TestID	Frame Rate	Resolution	BW	Sat_Grp1	Sat_Grp2	Sat_Grp3	Sat_Grp4	Sat_All
1	19.5632	2425	20701.42	0	0	0	1	0.25
2	19.9093	3672	31901.21	0	0	0	1	0.25
3	19.5144	3261	27768.64	0	0	0	1	0.25
4	25.6902	2115	23709.72	0	0	0	1	0.25
5	19.8847	3370	29241.36	0	0	0	1	0.25
6	20.1463	2391	21019.55	0	0	0	1	0.25
7	20.5549	3273	29356.88	0	0	0	1	0.25
8	23.6637	2525	26073.09	0	0	0	1	0.25
9	22.5915	2409	23748.18	0	0	0	1	0.25
10	20.7147	2665	24089.31	0	0	0	1	0.25
11	6.3003	873	2400.07	0.6668	0.4826	0	0	0.28735
12	20.6384	2228	20065.03	0	0	0	1	0.25
13	20.9583	2286	20906.48	0	0	0	1	0.25
14	20.7557	3266	29580.27	0	0	0	1	0.25
15	19.9082	2749	23881.15	0	0	0	1	0.25
16	18.9656	2857	23644.24	0	0	0	1	0.25
17	19.041	2143	17805.76	0	0	0	1	0.25
18	20.1385	2474	21740.79	0	0	0	1	0.25
19	20.1079	2573	22576.42	0	0	0	1	0.25
20	18.0731	2151	16963.74	0	0	0	1	0.25
Avg	19.86	2585.30	22858.67	0.03	0.02	0.00	0.95	0.25
StdDev	3.62	613.95	6278.04	0.15	0.11	0.00	0.22	0.01
Var	13.090	376939.168	39413848.134	0.022	0.012	0.000	0.050	0.000

Appendix 14: Optimal Stream Parameters for Two Streams and No Grouping:

Index	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	satGrp1	satGrp2	satGrp3	satGrp4	satRepAll
1	9.2891	592	2399.828	19.6131	2567	21971.36	0.6407	0.5991	0.0521	1	0.573
2	8.0918	680	2401.258	20.085	4098	35919.36	0.5924	0.3529	0	1	0.4863
3	6.5749	837	2401.593	15.8592	1155	7993.703	0.771	0.4803	0.5497	0.5449	0.5865
4	7.5476	729	2401.16	17.4318	1052	8002.814	0.6866	0.5127	0.5778	0.5856	0.5906
5	7.488	735	2401.806	27.5394	2149	25827.1	0.7393	0.5842	0	1	0.5809
6	8.4952	647	2398.627	19.0123	3955	32814.51	0.8028	0.6266	0	1	0.6074
7	9	611	2399.764	19.4559	2176	18475.45	0.5552	0.7234	0	1	0.5696
8	7.7546	709	2399.332	16.3385	1114	7942.955	0.7234	0.5201	0.6966	0.3988	0.5847
9	8	688	2401.946	29.5485	2455	31657.14	0.6772	0.5817	0	1	0.5647
10	8	687	2398.454	15.1874	1207	7999.732	0.7685	0.895	0.6971	0.4276	0.697
11	8.211	670	2400.798	27.3223	2263	26982.77	0.62	0.5988	0	1	0.5547
12	9.023	610	2401.959	16.6402	1102	8002.485	0.5979	0.3897	0.744	0.2572	0.4972
13	8.8632	621	2401.966	17.3081	1059	7998.897	0.6495	0.4506	0.7949	0.4164	0.5779
14	7.6693	717	2399.715	15.9497	1149	7997.556	0.6719	0.5612	0.7109	0.4343	0.5946
15	8.1615	674	2400.571	32.8135	2273	32548.93	0.8208	0.4827	0.0197	1	0.5808
16	8.1782	673	2401.914	24.0621	2257	23700.08	0.6782	0.5314	0.0164	1	0.5565
17	8.4307	652	2398.811	22.8601	2340	23344.19	0.7112	0.4758	0	1	0.5467
18	7	786	2401.073	21.3236	2372	22072.93	0.8972	0.6547	0	1	0.638
19	8.995	611	2398.43	14.589	1257	8002.87	0.9398	0.5603	0.7005	0.2999	0.6251
20	7.4646	737	2400.815	15.9162	1152	8001.596	0.6351	0.4856	0.8877	0.4353	0.6109
Avg	8.111885	683.3	2400.491	20.4428	1957.6	18362.82	0.708935	0.55334	0.32237	0.74	0.581155
Stdev	0.714089	62.56374	1.269135	5.286346	916.1392	10445.02	0.100922	0.118986	0.362857	0.302386	0.045589
Var	0.509924	3914.221	1.610705	27.94545	839311	1.09E+08	0.010185	0.014158	0.131665	0.091437	0.002078

Appendix 15: Optimal Stream Parameters for Two Streams and Grouping:

Index	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	satGp1	satGp2	satGp3	satGp4	satGpAll
1	8.59	640	2398.99	14.19	1257	7785.254	0.6501	0.6125	0.6483	0.0396	0.4876
2	7.88	698	2401.77	21.89	1466	14003.7	0.6074	0.3352	0	0.4674	0.3525
3	8.12	677	2399.76	16.79	1092	8002.398	0.6951	0.5076	0.4776	0.5912	0.5679
4	10.76	895	4202.42	22.15	1526	14751.91	0	0.7895	0.1401	0.9145	0.461
5	7.54	730	2401.36	21.04	1667	15306.88	0.7388	0.5846	0	0.8068	0.5326
6	10.82	890	4202.64	19.32	1696	14298.14	0	0.8957	0.1425	0.8197	0.4645
7	9.00	1069	4198.60	20.39	1628	14483.85	0	0.9295	0.0247	0.8817	0.459
8	7.85	700	2398.72	19.23	1573	13203.32	0.7176	0.5259	0	0.7538	0.4993
9	8.09	680	2401.73	20.18	1502	13226.65	0.673	0.583	0	0.4608	0.4292
10	8.60	640	2401.83	15.72	1133	7771.429	0.7681	0.871	0.6318	0.4113	0.6706
11	10.00	963	4202.53	21.14	1706	15740.04	0	0.8707	0.0683	0.8678	0.4517
12	5.98	920	2401.26	49.95	1603	34940.66	0.8141	0	0	0.8061	0.4051
13	8.72	631	2401.54	16.00	1146	8001.83	0.6553	0.4415	0.8363	0.3271	0.5651
14	8.01	687	2400.10	22.98	1364	13677.35	0.6624	0.5637	0	0.6631	0.4723
15	5.37	1025	2400.89	12.26	785	4200.334	0.8815	0.7088	0.2826	0	0.4682
16	8.00	688	2401.95	17.18	1006	7543.214	0.6912	0.5224	0.5635	0.2256	0.5007
17	8.32	661	2399.73	19.09	1699	14157.72	0.7124	0.4742	0	0.7124	0.4747
18	6.88	799	2400.58	20.69	1657	14962.04	0.9004	0.6472	0	0.8256	0.5933
19	8.54	644	2400.57	18.01	1389	10917.96	0.9443	0.5382	0	0.6532	0.5339
20	8.17	673	2399.62	16.50	1108	7978.023	0.6181	0.4781	0.8623	0.4194	0.5945
Avg	8.26	765.5	2760.83	20.24	1400.15	12747.63	0.58649	0.593965	0.2339	0.582355	0.499185
Stdev	1.33	140.856	739.07	7.54	274.6298	6274.042	0.314286	0.218098	0.311105	0.277699	0.073603
Var	1.76	19840.5	546230.69	56.85	75421.5	39363603	0.098776	0.047567	0.096787	0.077117	0.005417

Appendix 16: Optimal Stream Parameters for Three Streams and No Grouping

TestId	frCh1	resCh1	bwCh1	frCh2	reCh2	bwCh2	frCh3	resCh3	bwCh3	satGp1	satGp2	satGp3	satGp4	satGpAll
1	8.679	634	2401.285	12.2707	1494	8000.271	29.0645	2421	30707.35	0.6507	0.6121	0.6709	0.8333	0.7334
2	7.2222	761	2398.496	14.0809	1302	8000.666	19	2156	17876.69	0.642	0.5786	0.7665	0.832	0.7463
3	8.0093	687	2401.243	14.458	1268	8000.409	18.0349	2320	18259.4	0.5904	0.3855	0.5026	0.9086	0.6196
4	8.3127	662	2401.512	11.3644	1613	7999.551	19	2058	17064.11	0.8602	0.5839	0.4908	0.8333	0.7337
5	9.0791	606	2401.044	12.6005	1455	8000.839	19.7414	2367	20392.05	0.6183	0.5237	0.449	0.8332	0.6477
6	8	687	2398.454	13	1410	7999.212	24.0022	2500	26186.4	0.8016	0.5673	0.5265	0.8333	0.7238
7	6	917	2401.073	11.1867	860	4198.413	17	3532	26203.2	0.8486	0.8391	0.3189	0.8333	0.7516
8	6.9976	786	2400.25	10.9251	881	4200.356	26.8574	2797	32782.43	0.8092	0.7625	0.1118	0.8333	0.6709
9	7	786	2401.073	13.866	1322	7999.584	20.2891	2270	20098.95	0.6611	0.5504	0.5277	0.8333	0.6848
10	5	1100	2400.2	10.5681	889	4099.996	24.4585	2820	30099.8	0.8904	0.863	0.0329	0.8333	0.6966
11	7.8177	704	2401.798	13	1410	7999.212	22.837	2833	28233.87	0.6117	0.5732	0.5662	0.8333	0.6878
12	7.8924	697	2400.638	13.7263	1336	8002.85	18.111	2270	17941.26	0.6216	0.6443	0.7159	0.8964	0.7454
13	10.2483	536	2397.184	16.4657	1113	7997.608	25.5258	2858	31836.57	0.6201	0.5934	0.5996	0.9156	0.7033
14	8.5989	640	2401.638	13.3971	1368	7998.004	24.1915	3010	31777.08	0.7512	0.5386	0.4638	0.8563	0.6884
15	7.6636	718	2401.276	14.1878	1292	7999.49	25.8487	2201	24828.1	0.5496	0.5956	0.8005	0.9299	0.7364
16	8.189	672	2401.513	14.1989	1291	7999.552	35.0015	2357	36002.36	0.7637	0.5644	0.6089	0.8333	0.7343
17	6.6296	830	2401.321	12.5696	766	4201.796	28.9732	2004	25338.38	0.803	0.7819	0.3502	0.8333	0.7337
18	9.4508	582	2400.36	13.3414	1374	7999.685	23.2522	2007	20365.55	0.7773	0.7026	0.6012	0.9642	0.7703
19	8.1865	672	2400.78	13.5988	1348	7999.728	26.5002	2438	28194.71	0.7733	0.501	0.5181	0.9008	0.6981
20	8.9471	615	2401.276	14	1310	8003.576	19.3226	3228	27219.73	0.6372	0.682	0.7733	0.8716	0.7731
Avg	7.89619	714.6	2400.621	13.1403	1255.1	7235.04	23.35059	2522.35	25570.4	0.71406	0.622155	0.519765	0.86208	0.71396
Stdev	1.206113	127.0671	1.218399	1.409168	231.3773	1569.822	4.635944	420.1108	5761.323	0.103572	0.11856	0.201356	0.041599	0.039932
Var	1.454707	16146.04	1.484496	1.985754	53535.46	2464342	21.49197	176493.1	33192842	0.010727	0.014056	0.040544	0.00173	0.001595

Appendix 17: Optimal Stream Parameters for Three Streams and Grouping

Index	frCh1	resCh1	bwCh1	frCh2	reCh2	bwCh2	frCh3	resCh3	bwCh3	satGp1	satGp2	satGp3	satGp4	satGpAll
1	8.445	651	2399.194	11	875	4200.35	20.9876	1309	11989.12	0.6487	0.8097	0.2137	0.446	0.5518
2	6.8334	805	2400.587	9	1068	4194.677	20	1342	11712.98	0.6722	0.8934	0	0.3976	0.5107
3	8	687	2398.454	15.3576	1193	7995.554	35.9443	1293	20282.12	0.5908	0.3851	0.484	0.6458	0.527
4	6.9998	786	2401.004	11.9998	802	4199.844	19.1332	1560	13025.58	0.8958	0.8679	0.0655	0.6634	0.6563
5	6.1136	173	461.5597	9	611	2399.764	19.5084	1604	13655.6	0.6231	0.5223	0	0.6519	0.4819
6	7.0026	785	2398.909	9	1069	4198.604	20.5388	1497	13417.81	0.8011	0.8491	0	0.5752	0.5851
7	7.4587	737	2398.918	11	875	4200.35	19.5738	1650	14094.31	0.793	0.8442	0.3155	0.6951	0.6967
8	7.5961	724	2400.015	12	802	4199.914	21	1805	16541.74	0.7892	0.7257	0.12	0.7309	0.628
9	6.8803	799	2399.048	8.9998	1069	4198.511	20.4339	1629	14526.37	0.6665	0.8551	0	0.7635	0.6094
10	5.974	921	2401.096	11.996	795	4161.868	17	1766	13101.6	0.8511	0.8472	0.0424	0.7863	0.671
11	6.9045	797	2401.46	10	962	4198.168	26.3532	1252	14398.67	0.6724	0.8672	0.1692	0.4688	0.5678
12	8.2104	670	2400.622	16	1146	8001.83	18	1563	12277.68	0.6079	0.6455	0.5885	0.5321	0.617
13	6.5812	836	2401.022	12	802	4199.914	19.2764	1546	13005.29	0.7607	0.8506	0.1592	0.6493	0.6374
14	6.6198	831	2400.66	11	875	4200.35	19.1323	1377	11497.04	0.7887	0.7903	0.022	0.5747	0.5727
15	8.0502	682	2395.939	16.2628	1126	7991.319	21.0005	1390	12738.82	0.5285	0.6073	0.7152	0.4999	0.5882
16	5.9903	918	2399.805	9.6928	993	4200.328	26.4469	1575	18177.75	0.8757	0.8386	0.0612	0.6562	0.6407
17	6.4563	852	2400.535	11.0003	875	4200.465	23.6944	2755	28487.35	0.8004	0.8359	0.2785	0.8333	0.7287
18	7.6001	724	2401.279	11.0342	827	3982.274	26.366	1317	15153.56	0.8661	0.9355	0.0656	0.4768	0.6098
19	7.9874	689	2401.648	16	1146	8001.83	23.1796	1949	19715.26	0.7813	0.4912	0.3827	0.8682	0.6638
20	5	1100	2400.2	10	962	4198.168	18.1404	1506	11922.2	0.8744	0.9148	0.0697	0.6258	0.6525
Avg	7.035185	758.35	2303.098	11.61717	943.65	4856.204	21.78549	1584.25	14986.04	0.74438	0.76883	0.187645	0.62704	0.609825
Stdev	0.890201	174.1222	433.4552	2.436242	154.3613	1660.149	4.347025	331.6842	4077.366	0.109242	0.154962	0.210884	0.131265	0.06285
Var	0.792459	30318.56	187883.4	5.935276	23827.4	2756096	18.89662	110014.4	16624915	0.011934	0.024013	0.044472	0.017231	0.00395

Appendix 18: Optimal Stream Parameters for Four Streams and No Grouping

	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	frCh3	resCh3	bwCh3	frCh4	resCh4	bwCh4	satGrp1	satGrp2	satGrp3	satGr4	SatAll
1	9.00	611.00	2399.76	9.80	983	4202.13	12.31	1489	7998.64	25.36	3253	36003.56	0.65	0.84	0.7683	1	0.8144
2	5.72	962.00	2401.14	10.00	962	4198.17	12.00	1528	8001.83	21.95	3758	36001.74	0.74	0.69	0.6108	1	0.7615
3	6.00	917.00	2401.07	8.77	1098	4201.00	13.00	1410	7999.21	21.61	3818	36002.38	0.78	0.77	0.6886	1	0.8108
4	6.08	905.00	2399.75	9.66	996	4198.80	14.09	1301	7998.50	23.06	3578	36006.01	0.71	0.81	0.6713	1	0.7971
5	7.00	785.00	2399.29	9.00	1069	4198.60	13.00	1410	7999.21	25.64	3218	36004.49	0.74	0.89	0.7128	1	0.8357
6	6.50	846.00	2400.21	10.00	962	4198.17	13.57	1351	8000.26	23.05	3579	35998.73	0.88	0.90	0.7196	1	0.8743
7	5.00	1100.00	2400.20	9.00	1069	4198.60	12.79	1434	8000.82	19.37	4260	36001.89	0.68	0.93	0.6699	1	0.8186
8	6.59	834.00	2400.19	9.00	1069	4198.60	13.90	1319	7999.16	21.36	3862	36006.90	0.78	0.83	0.8248	1	0.8568
9	7.13	771.00	2400.30	10.45	921	4198.70	12.39	1480	8002.61	23.78	3470	36007.97	0.68	0.84	0.6847	1	0.8015
10	8.24	668.00	2401.27	9.45	1019	4202.07	12.98	1412	7998.42	30.05	2745	36002.47	0.77	1.00	0.7974	1	0.8916
11	5.00	1100.00	2400.20	10.00	963	4202.53	12.77	1436	8002.27	24.87	3318	36005.93	0.78	0.87	0.6566	1	0.8274
12	5.47	1005.00	2400.27	10.60	908	4199.59	13.89	1319	7998.07	33.47	2465	36005.62	0.82	0.71	0.8337	1	0.8401
13	6.21	886.00	2399.51	10.44	922	4199.72	14.89	1231	7999.52	28.30	2915	36006.20	0.73	0.80	0.8429	1	0.8432
14	6.48	849.00	2399.86	8.91	1080	4200.33	13.23	1386	8000.84	22.68	3637	36001.52	0.69	0.82	0.8691	1	0.8436
15	6.00	917.00	2401.07	9.23	1043	4201.77	13.00	1410	7999.21	19.80	4167	36004.06	0.90	0.80	0.8465	1	0.8865
16	5.94	926.00	2400.51	9.00	1069	4198.60	13.39	1369	8000.15	20.98	3932	36005.08	0.77	0.82	0.7594	1	0.8375
17	7.95	691.00	2398.76	9.53	1010	4198.41	13.01	1409	7999.07	19.24	4288	36005.17	0.72	0.76	0.6244	1	0.7742
18	6.00	917.00	2401.07	8.17	1178	4201.36	15.30	1198	7999.21	20.82	3963	36000.30	0.92	0.87	0.7737	1	0.8928
19	7.93	694.00	2400.21	11.00	875	4200.35	13.98	1311	7999.79	18.30	4508	35999.86	0.95	0.87	0.7435	1	0.8898
20	7.00	786.00	2401.07	9.00	1069	4198.60	14.19	1292	8000.67	20.78	3970	35999.61	0.64	0.77	0.9135	1	0.8312
Avg	6.56	858.50	2400.29	9.55	1013	4199.81	13.38	1374.75	7999.87	23.22	3635.20	36003.47	0.76689	0.82823	0.750575	1	0.83643
Stdev	1.07	133.45	0.69	0.73	76.53	1.50	0.85	84.72	1.29	3.86	533.29	2.67	0.77264	0.827767	0.086315	0	0.037948
Var	1.14	17810	0.47	0.53	5856	2.26	0.71	7177	1.66	14.87	284404	7.13	0.774176	0.8345	0.00745	0	0.00144

Appendix 19: Optimal Stream Parameters for Four Streams and Grouping

Index	frCh1	resCh1	bwCh1	frCh2	resCh2	bwCh2	frCh3	resCh3	bwCh3	frCh4	resCh4	bwCh4	satGrp1	satGrp2	satGrp3	satGrp4	SatAll
1	8.59	640.00	2398.99	7.29	1321	4200.78	12.87	1424	7998.22	29.76	2772	35999.97	0.6501	0.7725	0.7587	1	0.7953
2	6.32	871.00	2401.08	5.86	1641	4200.04	12.55	1461	7999.66	21.72	3798	36006.88	0.7172	0.1861	0.5945	1	0.6244
3	7.85	701.00	2400.98	4.81	1999	4199.91	12.87	1425	8001.23	21.24	3884	36005.90	0.7098	0.5208	0.6841	1	0.7287
4	6.75	814.00	2398.90	12.20	789	4200.87	13.40	1368	7998.12	20.80	3966	36003.15	0.705	0.7425	0.6599	1	0.7769
5	7.54	730.00	2401.36	10.43	923	4202.34	13.44	1364	7998.90	22.23	3712	36006.86	0.7388	0.8113	0.7069	1	0.8143
6	8.89	619.00	2400.71	4.25	2265	4200.40	12.84	1427	7998.01	20.17	4091	36005.64	0.7808	0.6342	0.6883	1	0.7758
7	8.77	627.00	2399.10	11.60	830	4200.21	13.13	1396	7997.83	20.81	3964	36004.88	0.5667	0.8637	0.6623	1	0.7732
8	7.85	700.00	2398.72	4.63	2081	4200.19	12.54	1462	8001.11	21.41	3854	36005.13	0.7176	0.5497	0.7859	1	0.7633
9	8.09	680.00	2401.73	6.63	1451	4201.14	13.17	1392	8001.88	20.07	4110	36006.42	0.673	0.6215	0.6394	1	0.7335
10	8.60	640.00	2401.83	0.77	12507	4200.52	12.75	1438	8002.12	27.82	2965	35997.80	0.7681	0.871	0.7965	1	0.8589
11	8.81	624.00	2398.81	12.29	783	4198.52	13.48	1360	8000.37	21.20	3892	36005.33	0.5829	0.8007	0.6358	1	0.7549
12	5.98	920.00	2401.26	5.81	1657	4200.07	14.00	1310	8001.40	22.36	3689	36000.45	0.8141	0.15	0.8336	1	0.6619
13	8.72	631.00	2401.71	3.44	2798	4200.89	15.18	1208	8001.83	31.07	2655	36002.25	0.6553	0.4416	0.8428	1	0.7349
14	8.01	687.00	2400.10	11.43	842	4199.06	12.84	1428	8003.49	17.55	4702	36003.77	0.6624	0.6976	0.8556	1	0.8039
15	7.88	698.00	2400.55	4.84	1987	4200.54	13.14	1395	8001.29	20.04	4118	36006.06	0.8363	0.4695	0.8457	1	0.7879
16	8.00	688.00	2401.95	4.48	2146	4200.17	13.67	1341	8000.67	19.71	4186	36005.28	0.6912	0.5224	0.7583	1	0.743
17	8.32	661.00	2399.73	5.12	1880	4200.28	12.91	1420	8003.21	20.06	4112	36000.08	0.7124	0.5809	0.6201	1	0.7283
18	6.88	799.00	2400.58	4.29	2244	4200.53	15.44	1187	7999.84	21.15	3901	36000.24	0.9004	0.6472	0.7735	1	0.8302
19	8.54	644.00	2400.57	9.76	987	4201.78	13.36	1373	8002.79	21.70	3803	36006.15	0.9443	0.821	0.7216	1	0.8717
20	8.20	671.00	2401.04	0.57	16741	4200.09	15.29	1199	7998.35	18.93	4357	36002.72	0.6172	0.4771	0.8993	1	0.7484
Avg	7.93	702.25	2400.49	6.53	2893.6	4200.42	13.44	1368.90	8000.52	21.99	3827	36003.75	0.72218	0.609065	0.73814	1	0.76547
Stdev	0.85	84.94	1.09	3.60	4112.787	0.83	0.88	83.27	1.82	3.49	503	2.75	0.097386	0.203234	0.089586	1	0.058932
Var	0.71	7215	1.19	12.95	16915014	0.68	0.78	6934	3	12.16	252958	7.59	0.009484	0.041304	0.008026	1	0.003473

References

---

[1] Gojun Lu "Communication And Computing For Distributes Multimedia Systems" 1996 Artech House, Inc. Ch.2.1

[2] Gojun Lu "Communication And Computing For Distributes Multimedia Systems" 1996 Artech House, Inc. Ch.2.1

[3] Gojun Lu "Communication And Computing For Distributes Multimedia Systems" 1996 Artech House, Inc. Ch.2.6

[4] D. Hehmann, M. Salmony and H. Stuffgen, "Transport Services for Multimedia Applications on Broadband Networks" Computer Communications, vol. 13, no.4, pp. 197-203, May 1990

[5] D.C.A. Bulterman and R.V. Liere, "Multimedia Synchronization and UNIX", Proceedings of Second International Workshop on Network and Operating System Support for Digital Audio and Video, Nov 18-19, pp. 108-119

[6] Bolot, Turletti and Wakeman "A rate control mechanism for packet video in the Internet" Proceeding of IEEE INFOCOM'94.

[7] M.H. Ammar "Probabilistic multicast: Generalizing the multicast paradigm to improve scalability," in Proceedings of IEEE INFOCAM'94 March

[8] J.C.Bolt,T. Turtelli and I. Wakeman "Scalable feedback control for multicast video distribution in the Internet", in Proceedings of ACM SIGCOMM'94 pp. 58-67 September 1994

[9] Shun Yan Cheung, Mostafa H. Ammar and Xue Li, "On the Use of Destination Set Grouping to Improve Fairness in Multicast Video Distribution", Proceedings of IEEE INFOCOM '96, pp553-560, March 1996

[10] X. Li and M.H.Ammar, "Bandwidth control for replicated-stream multicast video distribution," in Proceedings of HPDC'96, (Syracuse, NY), August 1996

[11] S.Y.Cheung, M.H.Ammar, and X.Li, "Using destination set grouping to improve fairness in multicast video distribution," in Proceedings of INFOCOM '96, IEEE, March 1996

[12] McCanne, Jacobson and Vetterli, "Receiver-driven layered multicast" in Proceedings of ACM SIGCOMM'96(Stanford, CA), September 1996

---

[13] E.Amir, S. McCanne, and R.Katz, “Receiver-driven bandwidth adaptation for light-weight session”, in Proceedings of ACM Multimedia’97, (Seattle, WA), Nov. 1997

[14] X.Li, S.Pual, P. Pancha and M. H. Ammar, “Layered Video Multicast with Retransmission (LVMR): Evaluation of error recovery,” in *Proceedings of NOSSDAV’97* (St. Loius, MI), May 1997.

[15] L. Wu, R. Sharma, and B. Smith, “Thin Streams: An architecture for multicasting layered video” in Proceedings of NOSSDAV’97, (St. Louis, MI), May 1997

[16] J. Robinet, Y. Au and A. Banerurjea, “Implementation of a gateway for hierarchically encoded video across ATM and IP networks” in Proceedings of High Performance Networking Workshop (HPN) ’98 (Vienna, Austria), Sept. 1998

[17] Working paper by Gregor V. Bochmann and Zhen Yang, School of Information Technology and Engineering (SITE), University of Ottawa

[18], [19], A. Richards, G. Rogers, V. Witana, and M. Antoniades. “Mapping User Level QoS from a Single Parameter. In Second IFIP/IEEE International Conference on Management of Multimedia Networks and Services, Versailles, November 1998

[20] Working paper by K. El-Katib, X. He and G.v. Bochmann “Quality of Service Negotiation Based on Device Capabilities and User Preferences” School of Information Technology & Engineering , University of Ottawa

[21] Details about CC/PP are available from W3C Forum. http://www.w3.org/

[22] See reference 18

[23] `x = fminsearch(fun,x0)` starts at the point x0 and finds a local minimum x of the function described in fun. x0 can be a scalar, vector, or matrix.